

*Análise semiótica do uso de Robótica
Pedagógica no ensino de Programação de
Computadores*

Humberto Augusto Piovesana Zanetti

Dezembro / 2014

Dissertação de Mestrado em Ciência da
Computação

Análise semiótica do uso de Robótica Pedagógica no ensino de Programação de Computadores

Esse documento corresponde à dissertação de mestrado apresentada à Banca Examinadora no curso de Mestrado em Ciência da Computação da Faculdade Campo Limpo Paulista.

Campo Limpo Paulista, 15 de Dezembro de 2014.

Humberto Augusto Piovesana Zanetti

Prof. Dr. Rodrigo Bonacin (Orientador)

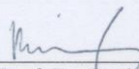
Faculdade Campo Limpo Paulista
Programa de Mestrado em Ciência da Computação

“Análise Semiótica do uso de Robótica Pedagógica no Ensino de Programação de Computadores”

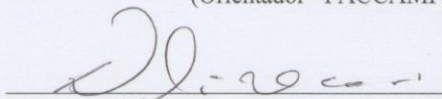
Humberto Augusto Piovesana Zanetti

Dissertação de Mestrado apresentado ao Programa de Mestrado em Ciência da Computação da Faculdade Campo Limpo Paulista, como parte dos requisitos para a obtenção do título de Mestre em Ciência da Computação.

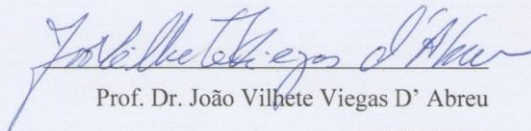
Membros da Banca:



Prof. Dr. Rodrigo Bonacin
(Orientador –FACCAMP)



Prof. Dr. Osvaldo Luiz de Oliveria
(FACCAMP)



Prof. Dr. João Vilhete Viegas D' Abreu
(UNICAMP)

Campo Limpo Paulista, 15 de dezembro de 2014.

FICHA CATALOGRÁFICA

Dados Internacionais de Catalogação na Publicação (CIP)

Câmara Brasileira do Livro, São Paulo, Brasil

Zanetti, Humberto Augusto Piovesana

Análise semiótica do uso de robótica pedagógica no ensino de programação de computadores / Humberto Augusto Piovesana Zanetti. Campo Limpo Paulista, SP: FACCAMP, 2014.

Orientador: Profº. Dr. Rodrigo Bonacin

Dissertação (mestrado) – Faculdade Campo Limpo Paulista – FACCAMP.

1. Robótica pedagógica. 2. Semiótica organizacional. 3. Programação de computadores. I. Zanetti, Humberto Augusto Piovesana. II. Faculdade Campo Limpo Paulista. III. Título.

CDD-005.1

Agradecimentos

Agradecimento especial ao meu orientador Prof. Dr. Rodrigo Bonacin, por ter acreditado em minha pesquisa e em minhas competências acadêmicas, e por toda sua orientação, dedicação e profissionalismo. Sem sua inestimável parceria este trabalho não teria sido possível.

A todo corpo docente do mestrado em Ciência da Computação Faccamp, por toda sua dedicação ao ensino, em especial ao Prof. Dr. Osvaldo Oliveira, pelo incentivo ao ingressar no curso e, conseqüentemente, o desenvolvimento desta pesquisa.

Aos meus colegas de mestrado da Faccamp, pela amizade e parceria no decorrer do curso.

A todos os alunos da escola técnica Etec Rosa Perrone Scavone, visto que sem eles essa pesquisa nunca teria se realizado.

A toda minha família, em especial à minha mãe, Celia Cristina Piovesana Zanetti, e ao meu pai, Alberto João Zanetti Neto, pela ajuda constante e por sempre acreditar em mim.

Agradecimento especial a minha amada esposa Flavia Fontana Rogerio Zanetti, por todo amor, carinho e paciência durante essa jornada.

Dedico este trabalho a todos educadores que, assim como eu, são entusiastas no uso de tecnologia e na inovação dentro da sala de aula.

“Ensinar não é transferir conhecimento, mas criar as possibilidades para a sua própria produção ou a sua construção”

Paulo Freire

Resumo

Alguns dos maiores desafios de ensino nos cursos de Computação estão relacionados com as disciplinas de Lógica de Programação, Construção de Algoritmo e Programação de Computadores. Estas disciplinas geralmente são ministradas logo no início do curso, causando grandes impactos devido ao seu nível complexidade e de abstração, trazendo dificuldades para alunos iniciantes e sendo motivo de evasão. A Robótica Pedagógica visa apoiar esse “árduo” processo de aprendizagem ao trazer algo mais receptivo e menos complexo para alunos ingressantes. A literatura apresenta vários estudos comprovando resultados positivos de sua aplicação em relação aos métodos tradicionais. Entretanto, a Robótica Pedagógica, como uma área em evolução, traz desafios de pesquisa tais como o estudo de um mecanismo formal e bem definido para o desenvolvimento de atividades e avaliações práticas de sua aplicação. Esta dissertação tem como objetivo investigar a utilização de ferramentas da Semiótica Organizacional como mecanismo para analisar e definir metas de aprendizagem, estruturar práticas e meios avaliativos. Como resultado, é apresentado um processo sistemático de desenvolvimento e avaliação de práticas com uso de recursos da Robótica Pedagógica para o ensino de Desenvolvimento de Algoritmos. Por meio de uma análise semiótica é possível eliciar e sistematizar os quais são os padrões comportamentais esperados dos alunos no desenvolvimento de práticas de programação no nível semântico, pragmático e social. Tal metodologia foi aplicada a três grupos de alunos de nível técnico para obtenção de resultados e análise da viabilidade do processo.

Palavras chaves: Robótica Pedagógica; Semiótica Organizacional; ensino de Programação de Computadores.

Abstract

Some of the major educational challenges in computer science courses are related to the disciplines of Logic Programming and Computer Programming. These disciplines are usually taught early in undergraduate courses, resulting big impacts due to their complexity and level of abstraction, bringing great difficulties for beginners and, consequently, being a cause of evasion in these courses. The Educational Robotics aims to support this "arduous" learning process, bringing something more receptive and less complex for students. The literature presents several studies pointing out positive results of its application compared to traditional methods. However, the Educational Robotics, as an evolving area, brings research challenges such as the study of formal and well defined mechanisms for the development of pedagogical activities and practical assessments. This proposal aims to investigate the use of methods from Organisational Semiotics as a means to analyze and define learning goals, structure and evaluation means. As a result, we present a systematic process to develop and evaluate practices with Educational Robotics for teaching algorithms. By using a semiotic analysis, it is possible elicit and systematize what are the expected students' patterns of behaviour in developing programming practices at semantic, pragmatic and social levels. This methodology will be applied to a three group of students with the objective of to obtain results and verify the feasibility of the methodology.

Keywords: *Educational robotics; Organizational Semiotics; teaching computer programming.*

Sumário

1. INTRODUÇÃO	1
1.1. CONTEXTO, MOTIVAÇÃO E JUSTIFICATIVA	1
1.2. OBJETIVOS, CONTRIBUIÇÕES E MÉTODOS.....	6
1.3. DESCRIÇÃO DO CENÁRIO E METODOLOGIA DE PESQUISA	9
1.4. ORGANIZAÇÃO E ESTRUTURA DA PROPOSTA	12
2. REFERENCIAL TEÓRICO METODOLÓGICO.....	14
2.1. A ROBÓTICA PEDAGÓGICA	14
2.1.1. O KIT DE ROBÓTICA PEDAGÓGICA	17
2.1.2. LIMITAÇÕES E DESAFIOS DA ROBÓTICA PEDAGÓGICA	21
2.2. O ESTUDO DAS DIFICULDADES NO ENSINO DE DESENVOLVIMENTO DE ALGORITMOS	22
2.2.1. METODOLOGIAS E PRÁTICAS DE ENSINO	25
2.3. A SEMIÓTICA ORGANIZACIONAL	28
2.3.1. A FERRAMENTA MEASUR.....	32
3. A SEMIÓTICA ORGANIZACIONAL NO USO DE ROBÓTICA PEDAGÓGICA	38
3.1. A ROBÓTICA PEDAGÓGICA NO ENSINO DE DESENVOLVIMENTO DE ALGORITMOS	39
3.2. ANÁLISE SEMIÓTICA NO CONTEXTO DO APRENDIZADO.....	41
3.2.1. FORMULAÇÃO DAS PRÁTICAS – MÉTODO PAM	42
3.2.2. DESCRIÇÃO DO DOMÍNIO – MÉTODO SAM	43
3.2.3. AVALIAÇÃO POR NORMAS – MÉTODO NAM	44
4. PROCESSO PARA DESENVOLVIMENTO E AVALIAÇÕES DE PRÁTICAS USANDO ROBÓTICA PEDAGÓGICA - PDAP-RP	46
4.1. FASE I – IDENTIFICAÇÃO DO CENÁRIO E PROBLEMAS	48

4.1.1 DELIMITAÇÃO DO CENÁRIO/PROBLEMA	49
4.1.2 ANÁLISE DE <i>STACKHOLDERS</i>	49
4.1.3 USO DE <i>FRAMEWORK</i> SEMIÓTICO	51
4.2. FASE II – DEFINIÇÃO E MODELAGEM DE AGENTES, COMPORTAMENTO E DEPENDÊNCIAS ONTOLÓGICAS	52
4.2.1. DEFINIÇÃO DE AGENTES E <i>AFFORDANCES</i>	53
4.2.2. AGRUPAMENTO DE CANDIDATOS E MODELAGEM DE ONTOLOGIAS	54
4.3. FASE III – ANÁLISE E MODELAGEM DE NORMAS	58
4.3.1 ANÁLISE DE NORMAS	59
4.4. FASE IV – APLICAÇÃO DE PRÁTICAS	60
4.4.1 DEFINIÇÕES DE PRÁTICAS USANDO RP	61
4.4.2 AVALIAÇÃO DE RESULTADOS	62
5. APLICAÇÃO DE PRÁTICAS COM MÉTODO PDAP-RP	63
5.1. ESTUDOS PRELIMINARES	63
5.1.1 ANÁLISE DE NORMAS E USO DE <i>FRAMEWORK</i> SEMIÓTICO	63
5.1.2 CRIAÇÃO E APLICAÇÃO EXPERIMENTAL DO PDAP-RP	65
5.2. DESCRIÇÃO DOS PARTICIPANTES E CONTEXTO DA APLICAÇÃO	66
5.3. ETAPAS DE APLICAÇÃO DO PDAP-RP	68
5.3.1 IDENTIFICAÇÃO DO CENÁRIO E PROBLEMAS NO ESTUDO DE CASO	68
5.3.2 ANÁLISE SEMÂNTICA NO DOMÍNIO	73
5.3.3 DELIMITAÇÃO DAS AÇÕES E COMPORTAMENTOS	74
5.3.4 APLICAÇÃO DA PRÁTICA E AVALIAÇÃO	79
6. ANÁLISE DOS RESULTADOS E DISCUSSÕES	85
6.1. ANÁLISE QUALITATIVA	86
6.2. ANÁLISE QUANTITATIVA	92

6.3. AVALIAÇÃO SOBRE AS NORMAS COMPORTAMENTAIS	98
6.4. DISCUSSÃO E TRABALHOS RELACIONADOS	104
7. CONCLUSÃO.....	108
7.1. CONTRIBUIÇÕES DA PESQUISA	108
7.2. TRABALHOS FUTUROS E LIMITAÇÕES DA PESQUISA	109
7.3. CONSIDERAÇÕES FINAIS	110
8. REFERÊNCIAS BIBLIOGRÁFICAS	113
ANEXO I – INFORMAÇÕES SOBRE A COMPETIÇÃO R.A.F.A.E.L.A.....	121
ANEXO II – ARTIGO PUBLICADO NA RETC.....	126
ANEXO III – ARTIGO PUBLICADO NO SBIE	139
ANEXO IV – IMAGENS DAS OFICINAS	150
ANEXO V – QUESTIONÁRIOS APLICADOS AOS ALUNOS	151
ANEXO VI – QUADROS DE RESPOSTAS	155
ANEXO VII – QUADROS DE AVALIAÇÃO DAS NORMAS	161

Glossário

AS – Análise Semiótica

CBIE – Congresso Brasileiro de Informática na Educação

CNE – Conselho Nacional de Educação

CNPq – Conselho Nacional de Desenvolvimento Científico e Tecnológico

IDE – *Integrated Development Enviroment*

IHC – Interface Humano-Computador

INEP – Instituto Nacional de Estudos e Pesquisas Educacionais

MEASUR - *Method for Eliciting, Analysing and Specifying User Requirements*

MEC – Ministério da Educação

NAM – *Norm Analysis Method*

OBR – Olimpíada Brasileira de Robótica

PAM – *Problem Articulation Methods*

PBL – *Problem Based Learning*

PDAP-RP – Processo de Desenvolvimento e Avaliações de Práticas usando Robótica Pedagógica

PV – Programação Visual

R.A.F.A.E.L.A. – Robótica Aplicada a Futuros Alunos de Engenharia e Lógica Aplicada

RP – Robótica Pedagógica

SAM – *Semantic Analysis Method*

SBIE – Simpósio Brasileiro de Informática na Educação

SI – Sistema de Informação

SO – Semiótica Organizacional

WIE – Workshop de Informática na Escola

Lista de Tabelas

Tabela 1 – Exemplo de Quadro de Normas Comportamentais	59
Tabela 2 – Layout do Quadro de Resposta.....	60
Tabela 3 – descrição dos níveis da cebola semiótica.....	70
Tabela 4 – Quadro criado a partir da análise do framework semiótico	72
Tabela 5 - Quadro da norma “Desviar de obstáculo”	76
Tabela 6 – Quadro da norma “Seguir traçado”	76
Tabela 7 – Quadro da norma “Encontrar traçado”	77
Tabela 8 – Quadro da norma “Identificar faixa”	77
Tabela 9 – Quadro da norma “Identificar falha no traçado”	78
Tabela 10 – Tabela síntese das respostas do questionário.....	91
Tabela 11 – Incidências dos cumprimentos das normas nos grupos	101
Tabela 12 – Análise dos grupos.....	103
Tabela 13 – Resultados do Teste t.....	103

Lista de Figuras

Figura 1 - Visão geral do <i>brick</i> e componentes do <i>kit</i> NXT 2.0. Adaptado de Lego (2013).....	19
Figura 2 - Visão geral do <i>brick</i> e componentes do <i>kit</i> EV3. Adaptado de Lego (2014). 19	
Figura 3 - Aspecto geral da IDE do LEGO NXT	20
Figura 4 – Blocos de programação da IDE Lego EV3.	20
Figura 5 – Exemplo do signo como uma relação triádica	29
Figura 6 – O Framework Semiótico (adaptado de Stamper, 1973).....	31
Figura 7 - Cebola semiótica (Bonacin, 2004, adaptado de Liu, 2000)	34
Figura 8 - Relação entre as bases teóricas/metodológicas.....	39
Figura 9 – As relações das áreas abordadas na pesquisa.....	46
Figura 10 – Visão geral do PDAP-RP	48
Figura 11 – Cebola Semiótica	50
Figura 12 – Framework Semiótico	51
Figura 13 – Fases da Análise Semântica (adaptado de Liu, 2000).	54
Figura 14 – Diagrama de ontologia parcial	56
Figura 15 – Diagrama de Ontologia com visão parcial do domínio.....	58
Figura 16 - Exemplo de prática em arena.....	64
Figura 17 – Exemplo de programação em blocos	75
Figura 18 – Desenho conceitual da arena.....	80
Figura 19 – Visão geral da arena	81
Figura 20 – Detalhamento da arena.....	82
Figura 21 – Detalhes da estrutura física do robô	83
Figura 22 – Gráfico com a porcentagem das respostas do Grupo de Controle	93
Figura 23 - Gráfico com a porcentagem das respostas do Grupo 1.....	94

Figura 24 - Gráfico com a porcentagem das respostas do Grupo 2.....	95
Figura 25 – Gráfico do número de ocorrências das respostas por questão do Grupo de Controle	96
Figura 26 - Gráfico do número de ocorrências das respostas por questão do Grupo 1 ...	96
Figura 27 - Gráfico do número de ocorrências das respostas por questão do Grupo 2 ...	98
Figura 28 – Desempenho geral no cumprimento das normas comportamentais do Grupo 1	99
Figura 29 - Número de ocorrências dos níveis de cumprimentos das normas do Grupo 1	100
Figura 30 - Desempenho geral no cumprimento das normas comportamentais do Grupo 2	100
Figura 31 - Número de ocorrências dos níveis de cumprimentos das normas do Grupo 2	101
Figura 32 - Desempenho geral no cumprimento das normas comportamentais do Grupo de Controle	102
Figura 33 - Número de ocorrências dos níveis de cumprimentos das normas do Grupo de Controle	102

1. INTRODUÇÃO

Um dos grandes desafios no ensino de Computação são as disciplinas que envolvem Lógica de Programação e Programação de Computadores, que muitas vezes representam o principal fator de insucesso e evasão de vários alunos. Tais disciplinas têm caráter introdutório e são apresentadas aos alunos logo no início do curso, e o processo de aprendizagem dos conceitos iniciais de Programação tem se mostrado complexo, abstrato e marcado pela presença constante de dificuldades para os iniciantes. A imperícia e o desinteresse por essas disciplinas são fatores marcantes que justificam os índices de evasão dos cursos de Computação e Informática.

Observou-se que o perfil mais comum entre os alunos que ingressam nessas disciplinas é de possuir pouca autonomia e baixa capacidade para resolver problemas. Geralmente, estes alunos assumem uma postura passiva, ou seja, têm dificuldade de buscar conhecimento de forma independente, devido à insegurança que o processo de programação de computadores proporciona e a falta de conhecimentos prévios em que possam se amparar.

Nesse cenário, torna-se relevante buscar meios para lidar com a complexidade que envolve o processo de ensino-aprendizagem e que ao mesmo tempo estimule o aluno. Como aponta Benitti (2012), entre as soluções possíveis para esse problema, a utilização da Robótica Pedagógica (RP) é umas das mais promissoras e mundialmente adotadas. Para teóricos na área de Educação, como Papert (1994), a utilização de robôs como artefatos educacionais, tem um grande potencial para prover um ambiente favorável ao aprendizado dentro da sala de aula. Benitti (2012) e Eguchi (2010) salientam ainda que a RP fornece uma vasta gama de possibilidades de práticas que podem atrair jovens estudantes, trazer uma abordagem pedagógica não tradicional e estimular a busca por soluções.

1.1. CONTEXTO, MOTIVAÇÃO E JUSTIFICATIVA

Atualmente há uma grande demanda por profissionais que atuem em áreas relacionadas à engenharia e à computação, não sendo possível completar as vagas devido ao baixo número de profissionais que se formam e também à qualidade da formação daqueles que concluem o curso.

Além disto, nota-se que existe uma diminuição significativa no interesse em cursar áreas relacionadas à computação (Benitti et al., 2010), motivada principalmente pelas dificuldades que envolvem o processo de aprendizagem, que impulsionam um alto índice de evasão discente, sendo um dos grandes problemas das instituições de ensino brasileiras. Segundo Silva Filho et al. (2007) cursos de graduação das áreas de Ciências, Matemática e Computação apresentaram um índice médio de 28% de evasão de alunos entre os anos de 2001 e 2005, destacando-se os cursos de Ciência da Computação, que atingiram um índice de 32%, em dados analisados do Instituto Nacional de Estudos e Pesquisas Educacionais Anísio Teixeira – INEP.

Um estudo do Observatório SOFTEX aponta que a média anual de evasão nos cursos superiores classificados como de Tecnologia da Informação (Bacharelado e Tecnologia) é de 21%, entre os anos de 2004 e 2008 e há uma taxa média de evasão acima de 20% desde 2005 (Softex, 2012).

Os altos índices de reprovação nas disciplinas iniciais de desenvolvimento de algoritmos ocorrem em parte devido a necessidade de uma nova forma de abstração com que o aluno se depare na análise e na solução de problemas propostos. Segundo Campos (2010), o nível de abstração necessário nessas matérias é muito alto e a formalização do português estruturado - utilizado para a formação de palavras utilizadas para comandar as ações a serem executadas pelo computador - pode não ser o melhor recurso didático.

O processo didático envolve também a prática da utilização de uma linguagem de programação, que serve como instrumento de aplicação dos conhecimentos adquiridos em um ambiente com recursos computacionais complexos, o que aumenta ainda mais as possibilidades de frustração ou desmotivação dos alunos. Uma abordagem que visa a diminuição da dificuldade no desenvolvimento de algoritmos e o aumento da facilidade na compreensão da lógica envolvida é a Programação Visual (PV), que tem como objetivo trazer ao ambiente de programação elementos visuais análogos, contextualizando os processos envolvidos no desenvolvimento de um software, metodologia que minimiza as dificuldades de compreensão.

Segundo Ferreira et al. (2012), a programação de computadores tem se tornado um importante instrumento no século XXI, mas estudantes de programação aprendem apenas a escrever e ler códigos em uma determinada linguagem de programação, muitas vezes sem saber raciocinar computacionalmente e sem entender a razão de estar usando

alguns dos elementos que são apresentados. Nesse sentido, facilitar a compreensão dos elementos usados na programação estimula e motiva pessoas que estejam no início do aprendizado em programação de computadores.

Há várias linhas de pesquisa que estudam metodologias de ensino que possam motivar e facilitar a aprendizagem de conceitos complexos, como forma de minimizar as dificuldades de aprendizagem e melhorar o panorama de evasão nos cursos citados. Seymour Papert foi um dos precursores nessa área com a teoria do Construcionismo. Segundo Valente (1993), o Construcionismo defende que o aprendizado deve ser construído a partir de um processo cíclico de formulação de hipóteses, teste e avaliação dos resultados. Essa teoria baseia-se na criação de significados através de experiências e ações de autoaprendizagem, sendo elemento ativo na construção do conhecimento por parte do sujeito (Papert, 1994).

Complementar à linha de pesquisa do Construcionismo, temos a metodologia *Problem Based Learning* (PBL), que propõe o aprender a aprender (Soares e Borges, 2011). Nessa metodologia, o próprio aluno é responsável pela criação do aprendizado e o papel do professor é agir como um facilitador, selecionando os problemas a serem resolvidos. O objetivo da metodologia é proporcionar um ambiente motivador no qual o aluno gere soluções a partir dos recursos disponíveis.

Segundo Huet et al. (2004), as atividades propostas em salas de aulas e sessões em laboratórios são os métodos mais comuns de ensino em disciplinas introdutórias à Programação de Computadores, mas têm se mostrado pouco efetivas. Alguns dos problemas são: aulas muito extensas, com muito conteúdo para assimilar; grandes grupos de alunos, o que inibe a interação com o professor; e pouco tempo para aplicação prática. Huet et al. (2004) nota que tais pontos negativos nas práticas mais comuns de ensino tornam o aluno menos motivado e menos produtivo, ocasionando uma construção de conhecimento deficitária.

Nesse contexto, o uso da RP apresenta-se como uma alternativa para agregar as premissas propostas pelo aprendizado Construcionista e pela metodologia PBL, além de possibilitar a utilização de PV. Essa metodologia consiste em um processo baseado na interação com dispositivos robóticos para motivar o aprendizado e desenvolvimento de processos cognitivos (d'Abreu e Garcia, 2010). Além disso, agrega um conjunto de recursos que visa o aprendizado científico e tecnológico integrado às demais áreas do

conhecimento, utilizando-se de atividades como design, construção e programação de robôs (Lopes, 2008).

Com a RP é possível propor atividades que desenvolvam a criatividade e o conhecimento, devido à necessidade imposta pelo problema definido. Para que essas tarefas sejam efetivas no processo de aprendizagem, devem haver métodos de coordenação por parte do facilitador e delineamento desses projetos com a participação de alunos. O desenvolvimento de tais práticas educacionais deve ser acompanhado, analisado e ao final é necessário verificar os possíveis problemas e a efetividade do processo de aprendizagem (Zanetti et al., 2012).

Mundialmente, a utilização de RP como instrumento didático vem crescendo cada vez mais. Atualmente existe um número significativo de pesquisas na área e também vêm sendo adotado como prática por escolas, colégios e instituições de ensino superior, dentro de suas grades curriculares ou como suporte pedagógico (d'Abreu, 2002; Benitti, 2012; NIED, 2014). No entanto, ainda são escassas pesquisas que sejam baseadas na realidade educacional brasileira, particularmente na formação técnica na área da Computação.

Há também aspectos da utilização da RP que carecem de aprofundamento e maiores resultados. Como aponta Benitti (2012), ainda se faz necessário estudar vários tópicos relacionados com a avaliação da aprendizagem do aluno, devendo ser investigado dois aspectos principais: (1) os contextos em que as avaliações são conduzidas; e (2) o projeto do experimento e a obtenção dos resultados.

Nesta dissertação a “avaliação da aprendizagem” é um termo que engloba maneiras de avaliar o desempenho do aluno diante de uma prática e se o mesmo consegue identificar e utilizar os recursos corretos e esperados de programação. Aprendizagem nesse contexto indica se o aluno teve a competência, habilidade e conhecimento necessários para efetivar uma prática, como resultado de uma experimentação e processo de aquisição de conhecimento. Está fora do escopo desta avaliação o desenvolvimento de outras habilidades (i.e., não menos importantes em outros contextos) tais como: de montagem do robô, de trabalhar em grupo, conhecimento sobre sensores e dispositivos de hardware, entre outras.

Outro tópico de estudo apontado por Benitti (2012) diz respeito a efetividade da robótica como ferramenta educacional no processo de ensino-aprendizagem, que surge pela necessidade de entender exatamente o que essa ferramenta pode oferecer como recurso educacional e como verificar os resultados obtidos. Esse tópico tem dois aspectos a serem considerados, segundo Benitti (2012): (1) contribuições para o aprendizado de conceitos/disciplinas; e (2) desenvolvimento de habilidades.

Segundo Alimisis (2013), ainda faltam na literatura avaliações sistemáticas e projetos experimentais confiáveis em RP. Benitti (2012) corrobora, afirmando que a maior parte dos estudos sobre o uso da RP é de natureza descritiva e se baseia em relatos de professores que conseguem resultados positivos com pequenas iniciativas, não apresentando um detalhamento sistemático.

Bredenfeld, Hofmann e Steinbauer (2010), por sua vez, apontam a falta de um exame sistemático dos projetos envolvendo RP e de uma avaliação significativa do impacto das abordagens ou do cumprimento dos objetivos dentro do processo de ensino-aprendizagem.

Outras pesquisas se mostraram pouco conclusivas, uma vez que os benefícios esperados não foram claramente medidos e definidos, porque não há um sistema de indicadores e de uma metodologia de avaliação padronizada para eles (Ortiz, Bustos e Rio, 2011). Por fim, apesar dos benefícios educacionais e motivacionais, geralmente positivos, sendo comprovados em diversos estudos, a literatura aponta para a necessidade de pesquisas que consigam trazer uma maneira mais sistemática e eficiente a especificação de atividades e a avaliação da aprendizagem através de práticas de RP.

No restante da dissertação será utilizada constantemente a expressão “Desenvolvimento de Algoritmos”, para englobar os termos relacionados, que em muitos cursos e currículos da área, são definidos como Lógica de Programação, Algoritmos e Programação ou Programação de Computadores. A expressão “Desenvolvimento de Algoritmos” terá como objetivo relacionar conhecimentos fundamentais necessários para que alunos possam desenvolver habilidades práticas e teóricas. Este termo envolve a concepção da solução algorítmica (desenvolver o algoritmo ou a lógica construtiva relacionada ao algoritmo) e capacidade do aluno em expressar através de uma linguagem, formar a rotina de comandos que representem o

algoritmo (diretamente relacionado às Linguagens de Programação ou Programação de Computadores).

A expressão Desenvolvimento de Algoritmos não incluirá aspectos relacionados a disciplinas comumente conhecidas como Projeto de Algoritmos. Tal aspecto teórico aborda um conhecimento mais aprofundado de teoria da Computação, como Matemática e Complexidade de Algoritmos, com um foco mais voltado à otimização e desempenho de soluções algorítmicas.

1.2. OBJETIVOS, CONTRIBUIÇÕES E MÉTODOS

O objetivo da pesquisa apresentada nesta dissertação é analisar e definir práticas inovadoras para o ensino-aprendizagem de conceitos relacionados ao Desenvolvimento de Algoritmos utilizando RP, seguindo as propostas definidas pelo aprendizado Construcionista e pela metodologia PBL na formação de alunos em cursos de nível técnico.

Serão avaliados casos reais de aplicação da robótica no aprendizado e formação de profissionais. Como resultado da dissertação, é proposta uma metodologia, que através de meios mais sistemáticos, seja eficaz para o desenvolvimento de práticas e avaliação da aprendizagem de Desenvolvimento de Algoritmos. Para atingir este resultado, é proposto o uso de RP via métodos provenientes da Semiótica Organizacional (SO).

A partir da avaliação adquirida pela metodologia, serão analisados os resultados, criando uma base teórica e prática para propor soluções que possam ser efetivas no processo de aprendizagem, e também um diagnóstico das atividades propostas para esse nível de formação técnica. Pretende-se, com este estudo, dar sequência ao trabalho iniciado em pesquisas anteriores, como em Zanetti et al. (2012), apresentado no Congresso Brasileiro de Informática na Educação (CBIE) em 2012, cujo um dos objetivos é abordar a aplicação da robótica em práticas de ensino.

Para alcançar os objetivos definidos, a questão principal dessa pesquisa é avançar na compreensão, através de uma análise semiótica de “Como a Robótica Pedagógica contribui como ferramenta de aprendizado e apoio no ensino de Desenvolvimento de

Algoritmos?”. Para responder tal questionamento, questões menores estão relacionadas. Entre elas:

- Como os robôs poderão se tornar objetos educativos e representativos para as práticas de Desenvolvimento de Algoritmos?
- Como a interface de programação desses robôs, compostos por elementos gráficos representativos (blocos) podem ajudar na abstração dos alunos?
- Como relacionar elementos abstratos de programação com objetos físicos, de forma a promover um processo de significação e compreensão dos alunos?
- A soma desses artefatos (robôs e interface de programação em blocos) terá efetividade como mediador do processo de ensino-aprendizagem e fator de motivação aos alunos na aprendizagem de aspectos complexos que envolvem o Desenvolvimento de Algoritmos?

Esta dissertação tem como principal contribuição apresentar uma análise das práticas desenvolvidas com alunos de ensino técnico da área de Informática em processo de formação profissional, utilizando-se robôs para a aprendizagem de Desenvolvimento de Algoritmos, desenvolvimento de projetos e solução de desafios. Dentre estas práticas, destacam-se:

- A realização de oficinas de robótica, para ambientação e estudo dos robôs e sua linguagem de programação;
- A elaboração de práticas pedagógicas com o intuito de comparar a curva de aprendizagem e desenvolvimento de soluções em métodos tradicionais e àqueles amparados pela Robótica Pedagógica;
- Proposição e aplicação de um método sistemático de desenvolvimento e avaliação de práticas utilizando RP;
- Avaliação individual, para identificação da efetividade ou não das atividades propostas, da melhora da aprendizagem decorrente da utilização de robôs e linguagens de programação visual.

Para a verificação e validação do alcance dos objetivos foi realizada uma avaliação adotando como referencial metodológico a SO, área da Semiótica, que interpreta todo conjunto ou organização como sendo um sistema de signos (símbolos) e

seus contextos sociais (Bonacin, 2004). Como aponta Gomes e Mendes (2007), as dificuldades no aprendizado de programação de computadores podem ter várias origens, incluindo aspectos ligados ao contexto social. Entre os aspectos destacados por Gomes, Henriques e Mendes (2008), é possível classificar como de ordem social os seguintes:

- A estratégia de ensino utilizada pelo professor não condiz com o estilo de aprendizagem do aluno;
- O ensino de conceitos dinâmicos através de materiais abstratos;
- A preocupação dos professores em ensinar a sintaxe da Linguagem de Programação, mais do que direcionar o aluno a resolver problemas;
- O alto nível de abstração;
- A falta de motivação dos estudantes para aprender algo abstrato, com sintaxe complexa.

Tendo em vista os aspectos supracitados, torna-se interessante uma análise semiótica do tema, na qual sejam explorados métodos que possam melhorar a aprendizagem da programação. Com a análise baseada em conceitos de semiótica, é possível observar como a comunicação e interpretação em um sistema de signos, diferente aos tradicionalmente apresentados nas Linguagens de Programação, poderá colaborar para a aquisição de conhecimentos por indivíduos ou grupos.

A SO apresenta métodos e artefatos que propiciam a análise organizacional para o desenvolvimento de sistemas computacionais (Liu, 2000). Dentre os sistemas computacionais, incluem-se a programação de robôs, devido aos seus aspectos de projeto. Com isso, podemos analisar quais são os aspectos internos desse sistema, assim como as interações com o ambiente provenientes dessa nova perspectiva de aprendizagem de programação, utilizando robôs como ferramenta pedagógica.

A partir dessa análise, haverá uma proposta de um método formal e sistemático de desenvolvimento e avaliação de práticas usando RP. Dentro do escopo da pesquisa, o método é proposto com o objetivo de responder à questão “Como um método formal e sistemático pode auxiliar no desenvolvimento e avaliação de práticas usando RP?”.

Abordagens cognitivas estão relacionadas ao entendimento da interação de pessoas com uma interface de programação, como por exemplo, sua percepção, execução de raciocínio e aprendizagem. Neste contexto uma análise semiótica pode

ampliar o leque de abordagens a serem estudadas, como os fatores interpessoais e sociais, levando em consideração a interpretação dos signos e a sua expressão. Na “análise semiótica” proposta, considera-se, entre outros aspectos, o cognitivo, já presente nas abordagens atuais, mas que adiciona novas questões comportamentais, perceptuais, avaliativas e denotativas através do estudo dos signos.

1.3. DESCRIÇÃO DO CENÁRIO E METODOLOGIA DE PESQUISA

O ponto de partida deste trabalho foi a partir da análise dos resultados de uma competição utilizando RP. Essa competição foi desenvolvida pelo projeto Forma-Engenharia¹ através de chamada pública do Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq), em parceria entre a Universidade São Francisco e a escola técnica ETEC Rosa Perrone Scavone, ambas as instituições localizadas na cidade de Itatiba, interior do estado de São Paulo.

A competição recebeu o nome de R.A.F.A.E.L.A. (Robótica Aplicada a Futuros Alunos de Engenharia e Lógica Aplicada)² e contou com a presença de vinte alunos de primeiro e segundo anos do curso técnico de Informática Integrado ao Ensino Médio, utilizando kits robóticos LEGO® Mindstorms em atividades relacionadas às práticas de engenharia e computação.

O evento foi realizado em novembro de 2013 e seguiu os moldes de grandes competições da área, tal como a realizada pela Olimpíada Brasileira de Robótica (OBR)³. Nesse formato são reunidas duplas de participantes que devem programar seu robô para executar tarefas em uma arena. Dentre essas tarefas estão desvios de obstáculos, reconhecimento de cores e áreas, seguimentos de linhas demarcadas no piso, entre outras.

Para o desenvolvimento dessas tarefas, foram priorizados os seguintes objetivos:

- Estimular o aluno a seguir seus estudos na área de Engenharia;
- Ter os objetivos bem definidos e possíveis de serem alcançados;

¹ Chamada pública do Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq), com a referência “CNPQ/VALE S.A. nº 05/2012 - FORMA-ENGENHARIA”, que teve como objetivo selecionar propostas para apoio financeiro a projetos para estimular a formação de engenheiros no Brasil, combater a evasão e despertar o interesse vocacional dos alunos de ensino médio pela profissão de engenheiro e pela pesquisa científica. Site do projeto: <http://roboticaeducacional.webs.com/>

² Informações sobre a competição estão disponíveis em: <http://roboticaeducacional.webs.com/>.

³ Olimpíada Brasileira de Robótica. Website: <http://www.obr.org.br/>.

- Ser desafiante e motivador;
- Contemplar noções básicas de programação;
- Analisar as formas como os alunos desenvolvem soluções através da robótica.

O Anexo I apresenta respectivamente: imagens da competição, exemplo completo de uma tarefa e questionário de avaliação para a finalização do projeto.

Para a análise e exploração do modelo proposto nessa dissertação, foram realizadas intervenções pedagógicas⁴ e oficinas com alunos das turmas de 1º, 2º e 3º anos do curso técnico de Informática Integrado ao Ensino Médio da Etec Rosa Perrone Scavone. Foram selecionados 16 alunos da turma de 1º ano, com idades entre 14 e 15 anos, 16 alunos da turma do 2º ano, com idades entre 15 e 16 anos e 8 alunos do 3º ano, com idades entre 16 e 17 anos.

O primeiro grupo é composto por alunos que estão cursando a disciplina de Lógica de Programação, que tem como objetivo ser a disciplina introdutória de aprendizagem de Desenvolvimento de Algoritmos e o primeiro contato com uma linguagem de programação formal. A estrutura programática da disciplina é composta essencialmente pelos seguintes itens:

- Aprendizagem de conceitos fundamentais de Desenvolvimento de Algoritmos;
- Desenvolvimento de algoritmos através da divisão de problemas;
- Introdução de uma linguagem de programação estruturada;
- Abordagem de conceitos de Desenvolvimento de Algoritmos na linguagem de programação estruturada;
- Solucionar problemas através de análise técnica.

O segundo grupo (com os alunos de 2º e 3º anos) é formado por alunos que já possuem experiência prévia com programação, pois já cursaram a disciplina introdutória de Lógica de Programação e em seu ano atual do curso têm contatos com disciplinas que aprimoram os conceitos de desenvolvimento de algoritmos. Entre esses alunos, também

⁴ A intervenção pedagógica nesse contexto pode ser interpretada como uma interferência feita pelo professor no desenvolvimento ou aprendizagem de um grupo de alunos. É um procedimento que pode ser adotado em um momento em que é apresentado algum problema no aprendizado, com o objetivo de compreendê-lo, explicitá-lo ou corrigi-lo. Também pode trazer novos elementos para que os alunos possam pensar e elaborar soluções de maneira diferenciada (Freire & Prado, 1996).

há alguns que possuem experiência em competição de robótica, como a R.A.F.A.E.L.A e a OBR.

Essas intervenções tiveram como objetivo trabalhar com três conceitos fundamentais para programação: estruturas de seleção, estruturas de controle e modularização. Foram preparadas duas práticas diferentes e aplicadas em quatro semanas no período de agosto a setembro de 2014, cada uma com foco em conceitos de programação, utilizando uma arena com obstáculos, traçados e faixas, utilizando os *kits* robóticos LEGO® Mindstorms. Durante a execução das tarefas os alunos foram orientados por um professor e após o término dessas práticas, foram realizadas análises visando uma avaliação dos conceitos aplicados e a eficiência do modelo proposto nessa dissertação.

A pesquisa teve tanto caráter qualitativo e exploratório, com foco em situações vivenciadas pelos alunos nas práticas utilizando RP, quanto quantitativo, com análises numéricas de aspectos-chaves. Quanto ao âmbito exploratório, foi utilizado ferramental estatístico-quantitativo com o objetivo de mensurar ou enumerar eventos em relação ao uso do método proposto. Também houve um recolhimento de dados quantitativos em questões de múltipla escolha e qualitativos através de observação direta e participante, além de entrevistas e questionários estruturados aplicados individualmente e respondidos pelos alunos.

A adoção do método qualitativo de pesquisa proporciona a oportunidade de observar e interpretar uma análise que extrapole dados racionais, como observação de comportamentos e linguagens “não-verbais” (motivação, satisfação, etc.).

Aplicada em um ambiente de aprendizagem e competição, a pesquisa qualitativa possibilitou desenvolver intuitivamente uma ideia sobre a forma que o modelo poderá ser aplicado e sua efetividade. Através das entrevistas e relatos de alunos é possível perceber se o indivíduo assimilou o que era esperado e seu comportamento perante as atividades propostas. Esses instrumentos podem ser organizados na seguinte categoria de análise:

- **Observação direta:** Motivação, empenho e autonomia dos alunos;

- **Relatos e artefatos de programação:** Capacidade de programação diante dos robôs, aplicação de conceitos de programação, facilidades/dificuldades em tarefas;
- **Entrevistas e questionários:** percepções dos alunos em relação a utilização, a aprendizagem e o ambiente fornecido pelo uso de RP.

Quanto a interatividade com alunos que já possuem certa experiência em programação, ou seja, vivenciam e possuem conhecimento sobre práticas para aprendizagem de programação, se mostra relevante para a validação do modelo, além de conduzir mudanças e acréscimos. O trabalho junto aos alunos que são iniciantes tem o objetivo de verificar indícios da eficácia do método para o desenvolvimento de práticas que incentivem a aprendizagem de Desenvolvimento de Algoritmos com a RP.

No que se refere aos dados com caráter quantitativo, foram feitos questionários utilizando a escala de resposta de Likert para especificar o nível de concordância com uma afirmação (questão) (Wainer, 2007). Através da adoção dessa escala e seu caráter bipolar (medindo respostas positivas e negativas), foi possível criar gráficos comparativos de respostas mais objetivas, intensificando a escala com fatores numéricos ordinais, sendo um (1) o item da escala com a resposta mais negativa e cinco (5) como sendo o item de resposta mais positiva.

1.4. ORGANIZAÇÃO E ESTRUTURA DA PROPOSTA

Textualmente, este trabalho está organizado da seguinte forma:

- O capítulo 2 apresenta todo o referencial teórico e metodológico utilizado na dissertação, que inclui a introdução de conceitos e práticas de RP, principais dificuldades no ensino de Desenvolvimento de Algoritmos e a teoria da SO.
- O capítulo 3 apresenta os principais métodos da SO e sua aplicação em cenário de ensino com o uso de RP.
- O capítulo 4 detalha as etapas do Processo para Desenvolvimento de Práticas, usando Robótica Pedagógica (PDAP-RP).
- O capítulo 5 mostra a aplicação experimental do PDAP-RP com os grupos de alunos.

- O capítulo 6 mostra e discute os resultados obtidos com o experimento.
- O capítulo 7 apresenta as considerações finais da proposta, contribuição da pesquisa e trabalhos futuros.
- Anexo I – Informações sobre a competição de robótica R.A.F.A.E.L.A realizada em novembro de 2013, contendo imagens da competição, um exemplo completo de uma tarefa e o questionário de avaliação para a geração de relatórios que foram submetidos ao CNPq, como exigência do projeto Forma-Engenharia.
- Anexo II – Artigo publicado detalhando estudo inicial na Revista Eletrônica de Tecnologia e Cultura (RETC), 14º edição, abril de 2014 (Zanetti e Bonacin, 2014A).
- Anexo III – Artigo publicado com resultados preliminares após o desenvolvimento do método no 25º Simpósio Brasileiro de Informática na Educação (SBIE 2014) (Zanetti e Bonacin, 2014B).
- Anexo IV - Imagens das oficinas realizadas com os alunos em setembro de 2014.
- Anexo V – Questionários aplicados aos alunos participantes do experimento.
- Anexo VI – Quadros de respostas dos questionários.
- Anexo VII – Quadros de avaliação sobre as normas comportamentais.

2. REFERENCIAL TEÓRICO METODOLÓGICO

A RP possui grande potencial como método de ensino, já que promove uma aprendizagem baseada na resolução de problemas, desenvolvimento do raciocínio e o pensamento crítico, em um ambiente motivador e interessante. Mas para sua efetiva aplicação e retorno de bons resultados, se faz necessário investigar quais são as melhores práticas e como elas podem ajudar no desenvolvimento cognitivo do aluno, e também como mensurar essa efetividade no processo educativo.

Na Seção 2.1 é apresentado o fundamental teórico e pesquisas na área de Robótica Pedagógica e suas aplicações; a Seção 2.2 mostra o referencial teórico no qual se baseiam estudos que analisam as dificuldades encontradas pelos alunos na aprendizagem de Desenvolvimento de Algoritmos; na Seção 2.3 é discutido o referencial teórico que envolve a Semiótica Organizacional e suas principais metodologias; e por fim a Seção 2.4 apresenta as considerações finais do capítulo.

2.1. A ROBÓTICA PEDAGÓGICA

Para a maior parte das pessoas o uso de robôs sempre está relacionado a algo novo, futurista, ou até mesmo ficção científica. Mas a robótica está cada vez mais inserida em nossa sociedade e presente em nosso cotidiano, seja em no trabalho, nas ruas ou até mesmo em casa. Esse aspecto futurista está embasado pelo fato de muitas pessoas associarem o termo “robô” a algum objeto mecânico que imite o ser humano, porém a robótica de maneira geral está em presente em indústrias, automação predial e entre outros ambientes há muito tempo e são equipamentos muitas vezes em nada se assemelham ao ser humano.

A robótica é uma ciência que cada vez mais ganha espaço na mídia e vem sendo aplicada em diversos ramos de atividades econômicas. Hoje é muito comum encontrar notícias de robôs auxiliando médicos em cirurgias delicadas ou na recuperação de pacientes; a exploração espacial só é viável devido ao uso de robôs; aplicação de monitoramento de clima, auxiliando na agricultura; dentre outras áreas as quais o uso da robótica se torna imprescindível.

Na área de educação, a robótica vem se mostrando forte aliada, podendo prover recursos tecnológicos eficazes e ambientes pedagógicos com grandes resultados.

Segundo Silva (2009) a utilização da robótica no processo educativo possui três aspectos positivos:

- O robô, como elemento tecnológico, traz uma série de conceitos científicos, cujos princípios básicos de diversas áreas, dificilmente são explorados com outros recursos;
- Robôs criam novas formas de interação e exigem uma nova maneira de gerar conhecimento, e;
- Envolve um processo motivação, colaboração, construção e reconstrução.

A adoção de ferramentas tecnológicas em práticas pedagógicas representa bem os avanços tecnológicos os quais alunos e professores vivenciam, potencialmente tornando o processo mais produtivo e estimulante, como afirma Papert (1994):

A mesma revolução tecnológica que foi responsável pela forte necessidade de aprender melhor, oferece também os meios para adotar ações eficazes. As tecnologias de informação, desde a televisão até os computadores e todas as suas combinações, abrem oportunidades sem precedentes para a ação a fim de melhorar a qualidade do ambiente de aprendizagem.

A utilização da RP não é algo novo, há décadas instituições de ensino e pesquisa a usam como ferramenta para criar uma atmosfera de ensino/aprendizagem mais atraente e completo. Segundo d'Abreu (2002), um ambiente no qual a RP é inserida propicia a montagem, automação e controle de dispositivos mecânicos. Ela também auxilia na interação entre aluno e professor junto a ferramentas tecnológicas, construindo conhecimento e caracterizando esse ambiente como pedagógico que não existia a princípio. Trata-se de um processo interativo, conciliatório, entre o concreto e o abstrato, na resolução de um problema que envolve etapas como: concepção, implementação, construção, automação e controle de um mecanismo.

Há diversas pesquisas sobre RP atualmente, com diferentes abordagens. Neste trabalho são exploradas as três principais abordagens encontradas na literatura (Eguchi, 2010):

- Abordagem curricular (*theme-based curriculum approach*): pesquisas que visam integrar diferentes áreas de conhecimento sobre um tópico de aprendizagem específico, sendo estudada principalmente através da investigação e comunicação;
- Abordagem baseada em projeto (*problem-based approach*): estudantes trabalhando em grupos e de maneira colaborativa para explorar problemas reais;
- Abordagem orientada a objetivos (*goal-oriented approach*): análise de resultados em competições com robôs, realizados fora das escolas ou universidades.

Um ambiente de RP pressupõe a existência de professores e alunos interagindo entre si em que eles produzem novos conhecimentos. É um ambiente de aprendizagem que não existe *a priori*. Fazem parte deste ambiente, componentes que permitem construir mecanismos que se movimentam e ferramentas de hardware e software para automação destes mecanismos. Além disso, existe a possibilidade de se desenvolver uma metodologia de ensino rica e diversificada, com base no currículo da disciplina. A RP também pode ser compreendida como um processo de interação com um dispositivo robótico (mecânico/eletromecânico), como forma de favorecer os processos cognitivos (d'Abreu e Garcia, 2010). Ou ainda, um conjunto de recursos que visa o aprendizado científico e tecnológico integrado às demais áreas do conhecimento, utilizando-se de atividades como design, construção e programação de robô (Lopes, 2008).

Com a utilização da RP, além das questões de motivação e aquisição de conhecimento, torna-se possível construir um ambiente de trabalho colaborativo e atividades interpessoais, o que muitas vezes é impossível com instrumentos educacionais convencionais. A prática pode explorar trabalho em equipe, a cooperação, a criatividade, a expressão escrita e oral, a organização e as habilidades relacionadas à execução de projetos. A RP trabalha com competências, além daquelas propostas pelos currículos escolares, oferecendo aos alunos experiências reais das áreas da informática, eletrônica, mecânica e design, através das possibilidades de conexão de peças mecânicas e de componentes eletrônicos, para realização de uma determinada tarefa, resolução de um determinado problema ou desafio (Papert, 1994).

A RP deve ser aplicada como atividade de experimentação e de construção de instrumentos, visando o aprendizado científico e tecnológico, integrando as demais áreas de conhecimentos, como define Lopes (2008). O principal objetivo não é priorizar o domínio técnico em robótica, mas utilizar ferramentas para construir e manipular artefatos que venham reproduzir, experimentar e analisar conceitos teóricos de maneira simples e objetiva. Como define d'Abreu (2002), atualmente temos iniciativas de pesquisadores em inovar e trazer para o contexto da disciplina ministrada experimentos que possam contribuir com o aprendizado, através de projetos isolados e com o objetivo de demonstração de fenômenos matemáticos ou físicos, nos diversos níveis escolares (d'Abreu e Chella, 1999).

Umas das características mais positivas encontradas na RP é a promoção da interdisciplinaridade. De acordo com Benitti (2012), a viabilização do conhecimento científico-tecnológico e o estímulo à criatividade de experimentação com forte apelo lúdico, podem ser proporcionados através do uso de RP, pois possibilita a integralização de áreas diversas, como robótica, mecânica, matemática, programação, entre outras. Benitti (2012) ainda salienta que a RP incentiva a descoberta e aplicação dos conceitos envolvidos, desenvolvendo a capacidade de elaborar hipóteses, investigar soluções, estabelecer relações e tirar conclusões. Segundo Zilli (2004), a RP além de permitir aos alunos um contato com tecnologia, contribui para o desenvolvimento de diversas competências, como resolução de problemas através da tentativa e erro, comunicação, aplicação de teoria em atividades concretas e análise crítica.

2.1.1. O KIT DE ROBÓTICA PEDAGÓGICA

Os *kits* LEGO® Mindstorms, tanto o modelo NXT 2.0 quanto o modelo EV3 (modelo mais recente, lançado em 2014), são formados por um conjunto de dispositivos robóticos com fins educacionais. Ele é composto por peças mecânicas e eletrônicas, que permite a criação de diversos modelos, e por um o ambiente de desenvolvimento integrado (*Integrated Development Environment* - IDE) (Lego, 2014). Dentre os diversos modelos possíveis que se podem construir, os mais adequados são os que formam um robô móvel. Um robô móvel permite o desenvolvimento de práticas mais diversificadas, utilizando quase sempre os recursos disponíveis na íntegra.

Entre as peças é possível encontrar encaixes de estruturas, sensores, motores e o “tijolo” (*brick*) programável. Esse *brick* atua como o cérebro do sistema, contendo um microprocessador e as conexões para comunicação com os sensores e motores (Lego, 2014). Para mover-se, o robô utiliza motores, que podem ser controlados através de rotações de 360° (por exemplo, em um movimento contínuo para frente) ou angulação em graus (por exemplo, girar 30° para a esquerda), além de definir a velocidade. O robô é capaz de interagir com o ambiente através de sensores, podendo reconhecer cores, intensidades luminosas, toques ou choques com objetos, sons e aferir distância.

Para a execução das tarefas propostas nas intervenções pedagógicas deste trabalho, a composição do robô móvel contava com os seguintes componentes, presentes em ambos os modelos (NXT 2.0 e EV3):

- Dois motores: responsáveis para movimentação do robô na arena;
- Sensor de luz: também chamado de *Light Sensor* ou sensor de luminosidade, serve para identificar variações entre traçado (faixa preta) no piso branco;
- Sensor de cor: também chamado de *Color Sensor*, responsável por identificar cores em contraste com o piso branco;
- Sensor ultrassônico: também chamado de *Ultrasonic Sensor* ou sonar, que verifica a distância em relação a obstáculos físicos.

Durante as práticas envolvendo alunos, foram utilizados diferentes de kits robóticos LEGO®, o NXT 2.0 e o EV3, que possuem todos os componentes supracitados. A Figura 1 mostra uma visão geral do *kit* NXT 2.0, com o *brick* e seus principais componentes:

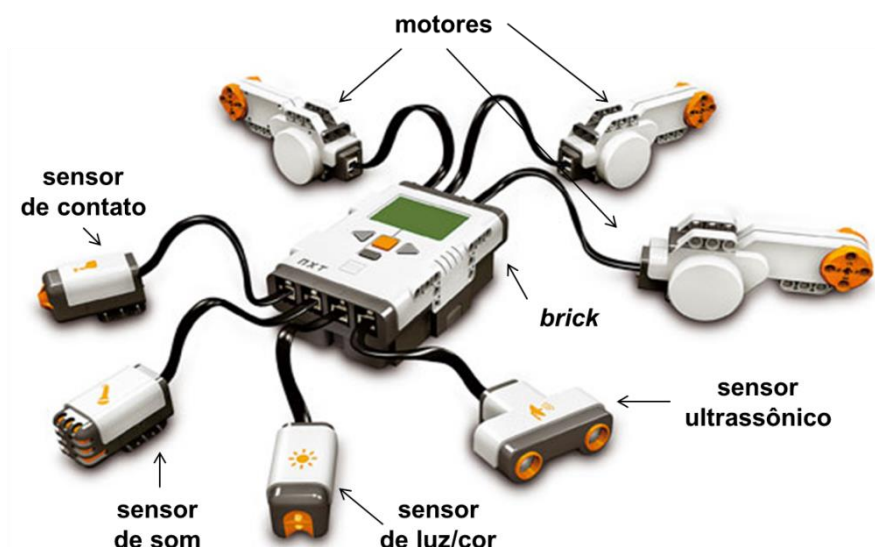


Figura 1 - Visão geral do *brick* e componentes do kit NXT 2.0. Adaptado de Lego (2013).

Ambos os modelos possuem aplicações e recursos similares, com diferenças significativas em relação ao aumento do poder de processamento do *brick*, e o acréscimo de um sensor giroscópio (para percepção de inclinação) no modelo EV3. A Figura 2 mostra a uma visão geral com componentes da versão EV3 *Educational*.

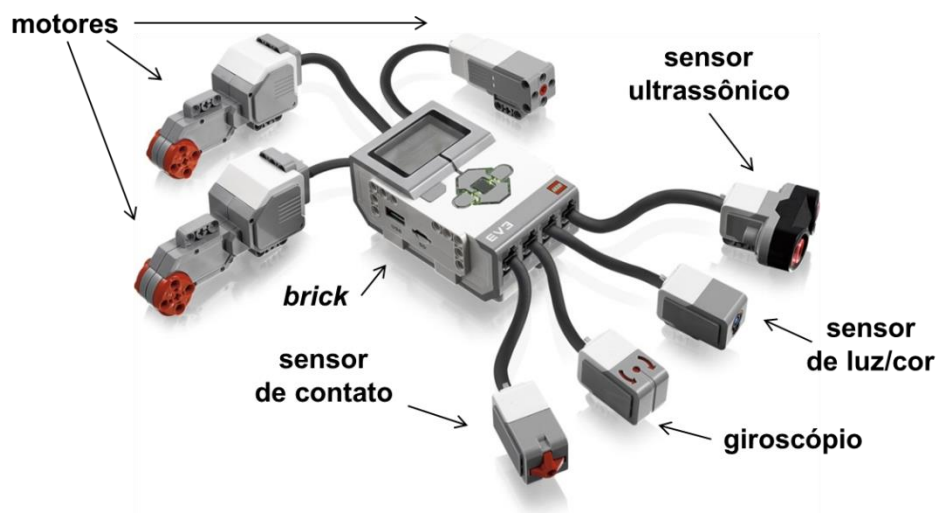


Figura 2 - Visão geral do *brick* e componentes do kit EV3. Adaptado de Lego (2014).

A IDE fornece um ambiente de programação visual baseado em blocos, de utilização intuitiva e pelo sistema *drag-and-drop*. No lado esquerdo da IDE, encontram-se os grupos de blocos, separados pelo grupo de funções. Cada um dos blocos pode ser arrastado ao plano de programação e configurado através das opções fornecidas,

dependendo de suas funções. A geração do programa se dá pela combinação desses blocos. A Figura 3 mostra o aspecto geral da interface da IDE.

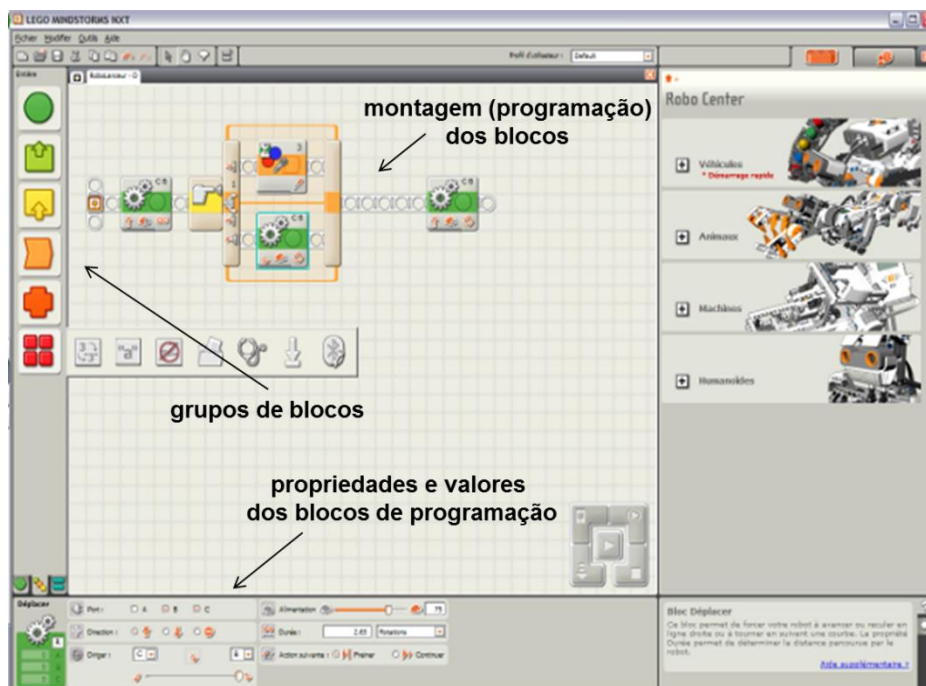


Figura 3 - Aspecto geral da IDE do LEGO NXT

Na IDE dos robôs LEGO®, os blocos representam seus sensores (como por exemplo, o sensor de luz) ou seus elementos atuadores (como os motores) e blocos com funções similares a estruturas de controle em programação. A Figura 4 apresenta no lado esquerdo um bloco para controle de um motor, ao centro há um bloco de comando de repetição similar à estrutura *while*, e no lado direito, um bloco de comando de decisão, similar ao *switch* ou *if-else*.

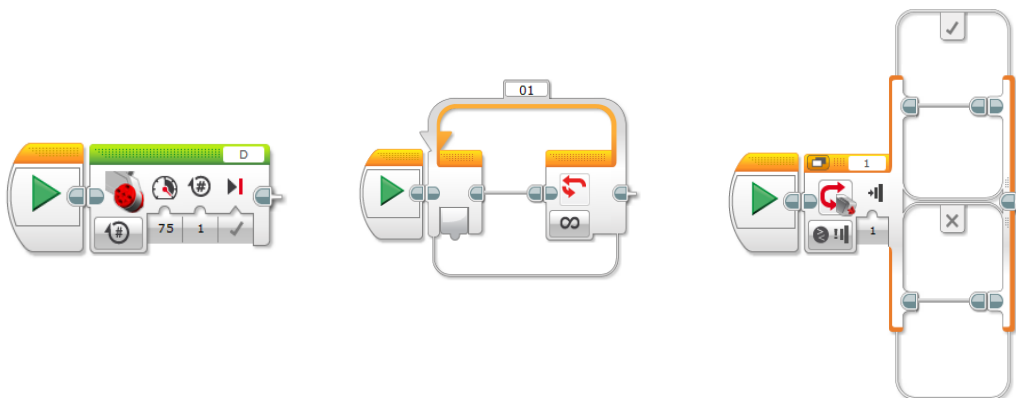


Figura 4 – Blocos de programação da IDE Lego EV3.

Ambientes de PV, segundo Kim e Jeon (2007), são ideais para programadores inexperientes ou em início de aprendizagem. Nesse tipo de ambiente encontram interfaces amigáveis e uma fácil identificação de componente e função. Gomes, Henriques e Mendes (2008) corroboram que a facilidade inerente a esses ambientes constituem uma vantagem no ensino de programação, uma vez que minimiza dificuldades associadas à aprendizagem de programação, como a complexidade da sintaxe e problemas de abstração por parte dos alunos.

2.1.2. LIMITAÇÕES E DESAFIOS DA ROBÓTICA PEDAGÓGICA

O trabalho de Benitti (2012) objetiva uma revisão literária sobre as recentes pesquisas na área de RP, abordando três principais aspectos: (1) identificar a potencial contribuição da RP na escola; (2) apresentar uma síntese de avaliações empíricas na efetividade educacional do uso de RP e; (3) definir quais são as perspectivas futuras sobre RP. Sobre as perspectivas de pesquisas futuras, através de sua análise literária, são apresentados três questionamentos relacionados aos desafios da utilização da RP: (1) *Quais os temas são ensinados através da robótica nas escolas?*; (2) *Como a aprendizagem dos alunos é avaliada?* e; (3) *A RP é uma ferramenta de ensino? O que os estudos mostram?*.

Ainda há uma demanda de estudo que possam trazer métodos de avaliação rigorosos e consistentes o suficiente para demonstrar a efetividade da RP como ferramenta de ensino. Em relação aos métodos de avaliação, Benitti (2012) discute dois aspectos que podem ser verificados:

- O contexto em que as avaliações são conduzidas: este aspecto questiona quais são os tipos de robôs utilizados, quais são os participantes, tamanho da amostra e o contexto educacional. Essa discussão aponta que a avaliação quando a RP é utilizada deve levar em consideração qual é o público e os objetivos esperados. A aplicação da RP com crianças com faixa etária menor que 10 anos e em práticas complementares ao currículo escolar, deve ter níveis de avaliações e objetivos diferentes de uma aplicação em um curso superior, em disciplinas de Desenvolvimento de Algoritmos, por exemplo;

- O projeto do experimento utilizado e como os resultados foram obtidos: nesse aspecto, a discussão recorre sobre a ausência de maneiras metódicas de elaborar e conduzir atividades, as quais possam efetivamente utilizar a RP como uma ferramenta para auxiliar o ensino. Também ressalta que através de um método formal, é possível criar meios para obter resultados de maneira mais fidedigna e coerente à atividade proposta. Muitas vezes, práticas utilizando RP são eventos extracurriculares ou complementares, não sendo o evento didático principal.

Adicionalmente, ao analisar resultados de d'Abreu (2002), compreende-se a necessidade de recursos que possam tornar as práticas usando RP integradas ao currículo de disciplinas da área tecnológica e que explorem a interdisciplinaridade.

2.2. O ESTUDO DAS DIFICULDADES NO ENSINO DE DESENVOLVIMENTO DE ALGORITMOS

A aprendizagem de Desenvolvimento de Algoritmos é um processo difícil, sendo necessário um alto nível de abstração e compreensão de uma linguagem de programação. Como aponta Sleeman (1986), desde os anos 1980 há pesquisas que mostram que aprender a programar não é uma tarefa simples para novatos. Além disso, Gomes et al. (2008) acrescenta que falta aos alunos iniciantes competências necessárias para a resolução de problemas, assim como a inadequação dos métodos pedagógicos que são apresentados. O maior entrave se mostra no fato de um conteúdo com caráter dinâmico ser apresentado de forma estática, o que não favorece o desenvolvimento dos alunos, por falta de compreensão ou desestímulo.

Outro fato apontado por Gomes et al. (2008) é a pertinência da motivação em um ambiente de aprendizagem de programação de computadores. Muito do que é lecionado em sala de aula utiliza métodos de estudos baseados em leituras sucessivas, memorização e mecanização de procedimentos. A mecanização dos processos na aprendizagem não apresenta os porquês da utilização de certo conceito de programação. Segundo Benitti et al. (2010), estudantes na área de Computação, em especial aqueles que nasceram com a presença de computadores pessoais, não percebem a razão pela qual têm que escrever códigos, já que estão habituados a usar programas que necessitam simples interações com ambientes gráficos.

O desenvolvimento do raciocínio lógico também é um fator preocupante. As disciplinas de programação possuem um caráter introdutório, preocupando-se em fornecer aos alunos as bases necessárias para a lógica de programação e o desenvolvimento de algoritmos. O nível de dificuldade envolvido nesse processo faz com que haja baixa motivação dos alunos, apatia e baixa autoestima culminando em evasão de cursos e altos índices de reprovação (Delgado et al., 2004). Para Borges (2000), o modo tradicional não consegue motivar os alunos a se interessar pela disciplina, pois não é claro para eles a importância de certos conteúdos para sua formação.

O ensino de Desenvolvimento de Algoritmos tem como propósito fazer com que os alunos desenvolvam um conjunto de competências necessárias para conceber sistemas computacionais capazes de resolver problemas reais. Entretanto, existe em termos gerais, uma grande dificuldade em compreender e aplicar certos conceitos abstratos de programação, por parte de uma percentagem significativa dos alunos que ingressam disciplinas introdutórias nesta área. Uma das grandes dificuldades reside precisamente na compreensão e, em particular, na aplicação de noções básicas, como as estruturas de controle, à criação de algoritmos que resolvam problemas concretos (Gomes et al., 2008).

Em um processo tradicional de aprendizagem, uma vez que os alunos têm posse do conhecimento, o próximo passo é elaborar soluções de problemas. O processo resolutivo pode ser resumido nos seguintes passos (Ambrósio et al., 2011):

- Compreensão do problema: leitura e compreensão do enunciado;
- Representação lógica: elaboração de um algoritmo que descreva uma sequência de comandos;
- Verificação e validação: manipulação de entradas e saídas, simulação, identificação de erros e correções.

Nesse processo tradicional, os alunos podem até mesmo entender o problema e ser capaz de resolver uma instância deste, entretanto encontram enormes dificuldades em traduzir a solução em uma sequência de comandos que possam ser executadas pelo computador (Ambrósio et al., 2011). Em seu trabalho, Winslow (1996) identifica as dificuldades dos alunos iniciantes na utilização de Linguagens de Programação,

afirmando que alunos iniciantes conhecem a sintaxe e a semântica, mas não sabem combiná-los para obter soluções válidas, e mesmo quando sabem resolver o problema, têm dificuldades em traduzir a solução para um programa de computador equivalente.

A maioria dos laboratórios utilizados para o ensino de programação não possuem recursos suficientes para trabalhar habilidades requeridas de formação para o século XXI. E quando há esses recursos, muitas vezes são simplesmente aplicadas de maneira antiquada e inapropriada (Alimidis, 2013; Blikstein, 2013). Tais laboratórios não são ambientes ideais para trabalhar competências técnicas, tampouco para desenvolver o pensamento crítico, a resolução de problemas, a criatividade, o trabalho em equipe e habilidades de comunicação (Blikstein, 2013).

Segundo define Saeli et al. (2011), programar envolve a habilidade de gerar uma solução para um problema, o que envolve a identificação da sua dimensão e a capacidade de dividi-lo em subproblemas. Nessa mesma linha de raciocínio, Resnick et al. (2009) diz que para geração de soluções são mais produtivas quando os recursos disponíveis são acessíveis e estimulam a criação. Com isso a utilização de RP para ensino de programação se mostra muito interessante.

Uma maneira de entender os objetivos do aprendizado de programação é pela ótica da Engenharia de Software, especificamente pela arquitetura do *software*, como descrita por Sommerville (2011). Essa arquitetura, através de um alto nível de abstração, define o sistema em termos de componentes, a interação entre eles e os atributos e funcionalidades de cada um. Essa visão estrutural muito se assemelha ao que ocorre na programação de robôs móveis, e segundo Sommerville (2011), traz os seguintes benefícios em um projeto de software:

- Organização do sistema como uma composição de componentes lógicos;
- Antecipação da definição das estruturas de controle globais;
- Definição da forma de comunicação e composição dos elementos do projeto;
- Auxílio na definição das funcionalidades de cada componente projetado.

No trabalho de Gomes et al. (2008) é apresentado uma síntese dos aspectos gerais, divididos e organizados por abordagens acadêmicas, que podem interferir no ensino e aprendizagem de conceitos de Desenvolvimento de Algoritmos:

- **Métodos de ensino:** A falta de um ensino personalizado: as estratégias dos professores não contemplam, habitualmente, todos os estilos de aprendizagem dos alunos; o ensino de conceitos dinâmicos é, normalmente, realizado através de materiais de natureza estática, e; os professores estão, normalmente, mais concentrados em ensinar uma linguagem de programação e os seus detalhes sintáticos, do que em promover a resolução de problemas usando uma linguagem de programação;
- **Métodos de estudo:** A programação (prática em programar) exige um estudo muito prático e intensivo e os alunos não estudam o suficiente para adquirir competências de programação;
- **Habilidades e atitudes dos alunos:** A maioria dos alunos apresenta enormes dificuldades em resolver problemas, muitos apresentam déficits de conhecimentos matemáticos e lógicos e, aos alunos falta “perícia” (erros ou percepções erradas sobre determinados conceitos de programação) específica de programação;
- **Aspectos psicológicos:** A falta de motivação e o período de vida em que os alunos têm que aprender a programar (período de transição).

2.2.1. METODOLOGIAS E PRÁTICAS DE ENSINO

Ao analisar as publicações do SBIE (Simpósio Brasileiro de Informática na Educação) e WIE (Workshop de Informática na Escola), dois dos mais importantes eventos na área de Informática e Educação no Brasil - que concentram vários trabalhos em RP-, é possível identificar o aumento de trabalhos de pesquisa, cujo foco são metodologias de ensino de Programação de Computadores⁵. Percebe-se que a maioria das publicações desse tema, entre 2008 e 2013, possuem uma abordagem direcionada a utilização de Jogos Digitais e RP. E entre os aspectos mais prováveis para amenizar as dificuldades no aprendizado de programação, destacam-se:

⁵ A consulta utilizou o termo “Programação de Computadores”, por ser a mais usual dentre os trabalhos. A análise feita através da verificação e leitura dos resumos encontrados na página que mantém os Anais do Workshop de Informática na Escola (WIE), disponível em <http://www.br-ie.org/pub/index.php/wie/issue/archive> e acessado no dia 07 de janeiro de 2014.

- O favorecimento de aspectos pedagógicos: optar por linguagens e ambientes de programação com menor carga sintática, ao invés de usar em cursos introdutórios, linguagens de grande aceitação no mercado, como por exemplo, Java e C (Bini, 2010; Benitti et al., 2010);
- O estímulo à resolução de problemas: motivar os alunos a compreender o problema e planejar a sua solução, além de criá-la e testá-la. Os problemas apresentados devem ser alinhados ao interesse do aluno, pois agindo a partir do próprio interesse, promove-se a motivação dos discentes. (Maloney et al., 2008; Falkner e Palmer, 2009; Gomes, 2008; Resnick et al., 2009; Blikstein, 2013; Bini, 2010).
- A diminuição da abstração: utilização de softwares que forneçam um retorno visual e/ou real, que possam favorecer o entendimento das estruturas básicas da programação de computadores e raciocínio lógico (Benitti et al., 2010; Ambrósio et al., 2011).

Vários trabalhos apontam soluções que podem contribuir para a melhoria do ambiente de ensino e aprendizagem, iniciando com um nível de abstração mais baixo, utilizando artefatos mais concretos e dinâmicos. Gomes et al. (2008) discute uma linha de pesquisa, na qual os conceitos devem ser apresentados através de resoluções de problemas genéricos e evoluindo para ambientes computacionais mais sofisticados. Para Resnick et al. (2009), a utilização de meios mais concretos que possam representar os conceitos de programação são ideais para o primeiro contato dos alunos.

No trabalho de Jenkins (2002), são discutidos aspectos que deveriam ser mudados em disciplinas introdutórias, indicando que o ensino relacionado a Desenvolvimento de Algoritmos e linguagens de programação deve utilizar recursos menos tradicionais. Os aspectos relacionados são: (1) a programação nunca deveria ser difícil até o segundo ano de qualquer curso; (2) a linguagem adotada deveria ser escolhida pelo seu fator pedagógico e não por ser popular no mercado; (3) a programação deve ser ensinada por aqueles que “podem ensinar programação” e não por aqueles que “sabem programar”; (4) os cursos devem ser projetados para serem flexíveis, permitindo que diferentes alunos possam aprender de diferentes maneiras; (5)

não haver uma avaliação somativa⁶ que possa causar pressão nos alunos e; (6) as instituições de ensino devem reconhecer que a programação é difícil e fornecer apoio adequado para os alunos.

A prática de programar deve contemplar altos níveis de abstração - principalmente quando estão relacionados ao processo de programação de estruturas de dados avançados, orientada a objetos e desempenho de algoritmos -, mas apenas quando o programador já possui um domínio sobre as estruturas lógicas básicas, autonomia e maturidade para resolver problemas, utilizando os recursos necessários e que consiga compreender todas as técnicas e formalismos de uma linguagem de programação.

Em contrapartida, quando o cenário apresenta alunos em fase de aprendizagem e desenvolvimento de habilidades, um nível de abstração menor pode ser vantajoso e eficaz. Dos Santos e Costa (2006) afirmam que o professor deve ser capaz de fazer o aluno compreender a abstração envolvida com toda a simbologia utilizada. Para isto, deve utilizar sua criatividade e tentar resolver cada problema baseando-se em situações do cotidiano, fazendo com que assim, o aluno comece a ter raciocínio lógico e ordem de pensamento.

Apresentar recursos de aprendizagem com baixo nível de abstração pode favorecer a inserção do aluno na nova rotina acadêmica. Segundo Buzin (2001), o aluno iniciante em um curso relacionado em Ciência da Computação traz uma bagagem cultural e de vivência acadêmica que, muitas vezes, não serve para o apropriado desenvolvimento de um estudo acadêmico em Computação. A experiência de estudante muitas vezes é de disciplinas desenvolvidas em torno do paradigma de apresentação de respostas e soluções, através de uma didática diretiva, ao invés de focar no processo de questionamento e resoluções de problemas.

Diante disso, faz-se necessário explorar alternativas que possam trazer um meio introdutório mais produtivo e relevante ao contexto de áreas tecnológicas e da Computação.

⁶ Esse tipo de avaliação ocorre ao final da instrução com a finalidade de verificar o que o aluno efetivamente aprendeu. Inclui conteúdos mais relevantes e os objetivos mais amplos do período de instrução; visa à atribuição de notas; fornece *feedback* ao aluno (informa-o quanto ao nível de aprendizagem alcançado), se este for o objetivo central da avaliação formativa; e presta-se à comparação de resultados obtidos com diferentes alunos, métodos e materiais de ensino (Menezes e Santos, 2002).

2.3. A SEMIÓTICA ORGANIZACIONAL

A Semiótica tem sido explorada por diversas áreas do conhecimento, mais presente nas Ciências Humanas, tais como Antropologia, Linguística, Filosofia, Educação, tornando-se uma filosofia científica. Mas a Semiótica tem grande relevância também em Computação, como em Interface Humano-Computador (IHC) e Semiótica Computacional (Bonacin, 2004).

A Semiótica é compreendida como a ciência que trata dos estudos dos signos, sua representação, o seu uso na cognição e comunicação (Gudwin, 2002). Liu (2000) aplica a Semiótica em áreas relacionadas à Ciência da Computação e Sistemas de Informação e atribui como base da Semiótica os trabalhos de Saussure (2004), com a criação da Semiologia e fundação da escola europeia de Semiótica, e posteriormente aos trabalhos de Peirce (1931-1958) e Morris (1938), fundando a escola da semiótica conhecida como “triádica” adotada na SO.

SO é um dos ramos da Semiótica e compreende uma organização como sendo um sistema de signos, tem como objetivo estudar a interpretação e a comunicação através deles pelas pessoas, de maneira individual ou em grupo (Bonacin, 2004). A SO utiliza conceitos e técnicas baseados na Semiótica *Pierciana*⁷ (Pierce, 1931-1958), que é baseada na lógica, na filosofia e na ciência da linguagem.

Na teoria Peirceana (Peirce 1931-1958) *“um signo, ou representâmen, é aquilo que, sob certo aspecto ou modo, representa algo para alguém. Dirige-se a alguém, isto é, cria, na mente dessa pessoa, um signo equivalente, ou talvez um signo mais desenvolvido. Ao signo assim criado denomino interpretante do primeiro signo. O signo representa alguma coisa, seu objeto. Representa esse objeto não em todos os seus aspectos, mas com referência a um tipo de ideia que eu, por vezes denominei fundamento do representâmen (...)”*.

Com o *representâmen*, o interpretante e o objeto, a tríade de Pierce pode ser graficamente representada por meio do triângulo, como na Figura 5 a seguir, que exemplifica o signo como uma relação triádica. Se, na interpretação de alguém (interpretante), o desenho de uma árvore, em uma tela, quadro ou folha de papel

⁷ O termo “Semiótica Pierciana” é corriqueiramente utilizada por diversos autores em detrimento aos termos “Semiótica de Pierce” ou a “Semiótica desenvolvida por Pierce”.

representa o objeto árvore (planta), essa relação (a tríade) entre o *representâmen*, o interpretante e o objeto é um exemplo de signo.

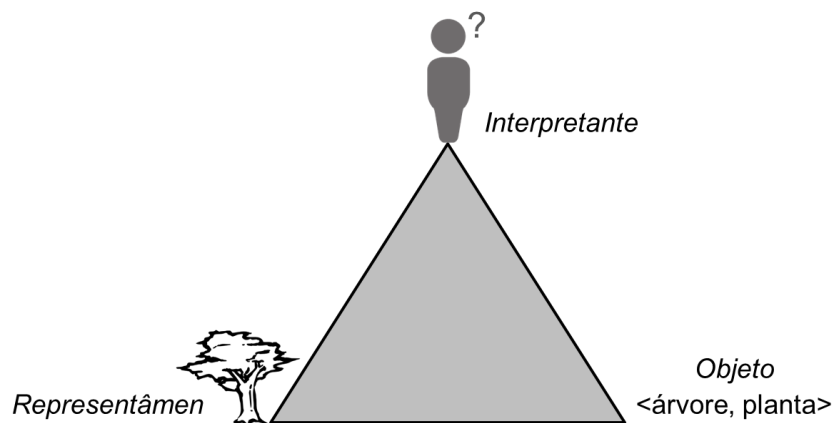


Figura 5 – Exemplo do signo como uma relação triádica

A partir do exposto, a Semiose pode ser compreendida como a produção de sentido, como elemento que compõe a tríade de Pierce. Segundo Gudwin (2002), a Semiose é um processo através do qual, alguma coisa (signo), representa outra (objeto), sob algum aspecto ou modo (interpretante), para um sujeito (intérprete). Pierce define que Semiose é “a ação do signo”, ou o processo pelo qual o signo tem efeito cognitivo. “[...]por semiose entendo uma ação, uma influência que seja ou coenvolva uma cooperação de três sujeitos, como, por exemplo um signo, o seu objeto e o interpretante, tal influência tri-relativa não sendo jamais passível de resolução em uma ação entre duplas” (Peirce, 1931-1958).

Conforme destacado por Eco (1976), na semiótica de Peirce, para estabelecer o significado de um significante é necessário nomear o primeiro significante por meio de outro significante, que a por sua vez conta com outro significante que pode ser interpretado por outro, e assim sucessivamente, constituindo um processo de Semiose Ilimitada. Segundo Eco (1976), esta é a única garantia de um sistema semiótico capaz de explicar-se a si próprio, em seus próprios termos.

Liu (2000) explica a fundamentação teórica da Semiótica Organizacional através de 4 características que descrevem a Semiose:

1. É universal e pode ser aplicada para todo tipo de propriedade de processamento de signos. Explica o mecanismo de criação e uso do signo;

2. É um processo capaz de identificar alguma coisa presente de acordo com critérios específicos ou normas;
3. O processo de representação pode ser recursivo: um *representâmen* pode ser visto como um objeto em outro processamento de signo, assim como um interpretante ou um objeto pode ser um *representâmen*;
4. É dependente de interpretação subjetiva. A Semiose está fortemente relacionada ao intérprete que pode ser um indivíduo, ou um grupo social que possui certo conhecimento e obedece a certas normas, sendo sempre subjetiva na interpretação, dependendo do ponto de vista do intérprete e o conhecimento e habilidade que ele possui.

A Semiótica Organizacional compreende uma área de pesquisa envolvida com o estudo de organizações, utilizando conceitos e métodos da Semiótica para modelar organizações do ponto de vista do processamento de signos (Liu, 2000). Para SO uma “organização” é qualquer sistema ou conjunto organizado ou que demande um comportamento organizado, no qual o processo de geração e interpretação de signos é um componente básico por trás do comportamento dessa organização. Vários estudos de Stamper (*e.g.*, Stamper, 1973) e Liu (*e.g.*, Liu, 2000) abordam principalmente os aspectos da SO aplicados a organizações humanas e a influência decisiva que as exercem sobre o comportamento organizacional, através do intercâmbio de signos de maneira individual ou em grupos.

Para Stamper (1973), a Semiótica se divide em seis níveis, sendo três níveis: sintático, semântico e pragmático, conforme apresentado no trabalho de Morris (Morris, 1938 apud Chandler, 2001) definidos como:

- **Nível sintático:** contendo as operações baseadas em regra entre signos de um sistema de signo (o que o signo implica);
- **Nível semântico:** mostrando a relação entre os signos e o mundo externo ao sistema de signos (o que o signo designa e denota);
- **Nível pragmático:** a avaliação do signo com relação aos propósitos de seus usuários (o que o signo expressa).

A esses níveis de Morris, Stamper (1973) adicionou mais três níveis: físico, empírico e social, formando assim um *framework* semiótico para analisar o uso de signos em uma organização. Esse três níveis são compreendidos como:

- **Nível físico:** contendo aspectos físicos relacionados à mídia em que os signos são representados;
- **Nível empírico:** contendo propriedades estáticas dos signos quando estes são utilizados em mídias diferentes;
- **Nível social:** onde os efeitos do uso dos signos no relacionamento humano em sociedade são estudados.

O *framework* da SO, composto por seis níveis, tem como objetivo identificar onde os efeitos do uso dos signos são percebidos. As camadas estão dispostas como mostra Figura 6 e são analisadas com técnicas específicas como a análise de informações, modelo de morfologia, análise semântica e análise de normas (Bonacin, 2004). Os três primeiros níveis dedicam-se ao estudo da função dos signos para comunicar significados e intenções e as consequências sociais do seu uso, abordando um aspecto mais humano. Os três níveis seguintes estão direcionados ao estudo de como os signos são utilizados na linguagem, como são organizados e transmitidos, e que propriedades físicas eles possuem ou são compostos.

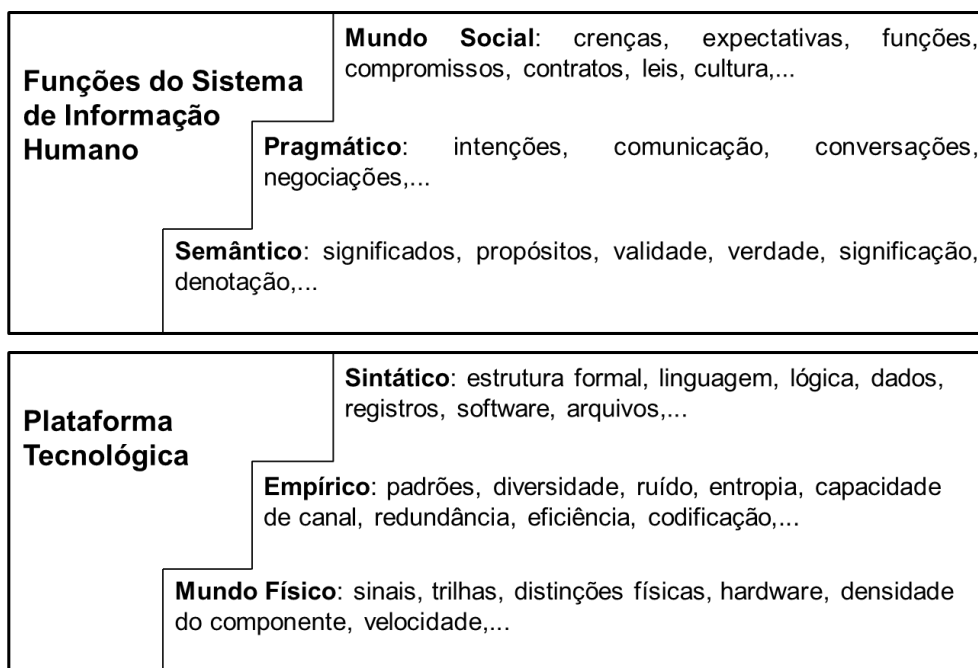


Figura 6 – O Framework Semiótico (adaptado de Stamper, 1973)

2.3.1. A FERRAMENTA MEASUR

Liu (2000) descreve que os diferentes níveis do *framework* são analisados por meio de métodos específicos, permitindo uma modelagem sob diversos pontos de vista e detalhando diferentes características semióticas em relação ao uso de signos na organização. Para isso, se faz necessária a utilização de uma ferramenta ou conjunto de métodos, como o MEASUR (*Method for Eliciting, Analysing and Specifying User Requirements*), orientados às normas e conceitos da SO para lidar com signos e seus significados (Semântica), intenções (Pragmática) e consequências sociais de seu uso (Social) (Stamper, 1993; Liu, 2000).

A ferramenta MEASUR é composta por cinco grandes métodos (Stamper, 1993; Liu, 2000):

- **Métodos para Articulação do Problema** (*Problem Articulation Methods* - PAM): podem ser utilizados nas fases iniciais de modelagem, que ajudam na identificação de elementos de maiores importâncias para o modelo. O PAM abrange quatro métodos: Definição da Unidade do Sistema (*Unit System Definition*), Quadro de Valoração (*Valuation Framing*), Análise Colateral (*Collateral Analysis*) e Morfologia do Sistema (*System Morphology*);
- **Método da Análise Semântica** (*Semantic Analysis Method* - SAM): tem como objetivo principal representar os requisitos do problema em um modelo formal. Normalmente, utilizam-se modelos de ontologias que descrevem uma visão de um domínio em foco;
- **Método da Análise de Normas** (*Norm Analysis Method* - NAM): Possibilitam a especificação de padrões gerais de comportamento dos agentes que participam do sistema organizacional nos níveis pragmático e social. Essa especificação é realizada por meio da análise dos comportamentos regulares e da sua descrição na forma de normas comportamentais;
- **Análise do Controle e Comunicação** (*Communication and Control Analysis*): analisa os diferentes tipos de comunicação entre os agentes

participantes da organização, por meio da categorização das mensagens trocadas entre eles;

- **Análise Meta-Sistêmica** (*Meta-Systems Analysis*): observa a organização de um ponto de vista externo, permitindo uma percepção de suas interações com outras organizações, em um nível hierárquico superior.

Nesse trabalho serão utilizados os métodos para Articulação do Problema (PAM), Análise Semântica (SAM) e Análise de Normas (NAM).

No processo de articulação de problemas (usando o PAM), o principal objetivo é identificar as partes interessadas (*stakeholders*) em um problema focal, com o intuito de discutir novas ideias e possíveis soluções (Bonacin et al, 2006). Para Liu (2000), essa fase inicial é importante para identificar problemas nos processos de negócio e técnicos, e também para entender contextos sociais, organizacionais, técnicos e culturais. Segundo Liu (2000), esse processo abrange quatro métodos:

- **Definição da Unidade do Sistema** (*Unit System Definition*): através desse método é possível listar e descrever os cursos e as interdependências dos agentes e seus interesses;
- **Quadro de Valoração** (*Valuation Framing*): método que possibilita mostrar quais são os comportamentos sociais em um determinado cenário, apontando quais são os benefícios e desvantagens;
- **Análise Colateral** (*Collateral Analysis*): método que auxilia estruturar um problema através das etapas mais fundamentais (núcleo do problema) e as que estão ao redor ou as atividades colaterais e subsequentes, e;
- **Morfologia do Sistema** (*System Morphology*): método para clarificar as três áreas funcionais de um sistema social: substantivo (*substantive*), comunicação (*communication*) e controle (*control*). O comportamento substantivo é composto por missões, tarefas, regras e normas; o comportamento relacionado à comunicação se preocupa com as “entradas” e “saídas” dos signos, mensagens que os agentes podem trocar, e; o comportamento de controle lida com aplicação e incentivo de uma norma.

Para fins de articulação de problemas, muitas vezes se faz necessária a utilização de artefatos como a Cebola Semiótica, que é capaz de representar um sistema de informação como um todo, a exemplo de uma organização social, em vez de abordagens estritamente técnicas.

A Cebola Semiótica é organizada em diferentes camadas. Na mais externa é apresentada a organização como um todo, definindo uma subcultura, cujos significados, intenções e crenças comuns caracterizam costumes e hábitos organizacionais e, no qual, compromissos são estabelecidos, alterados e descartados (Melo, 2007). Na camada intermediária, os significados e intenções são convertidos em um sistema formal de regras. Por fim, na camada mais interna, essas regras podem ser modeladas. A Figura 7 mostra as diferentes camadas da Cebola Semiótica.

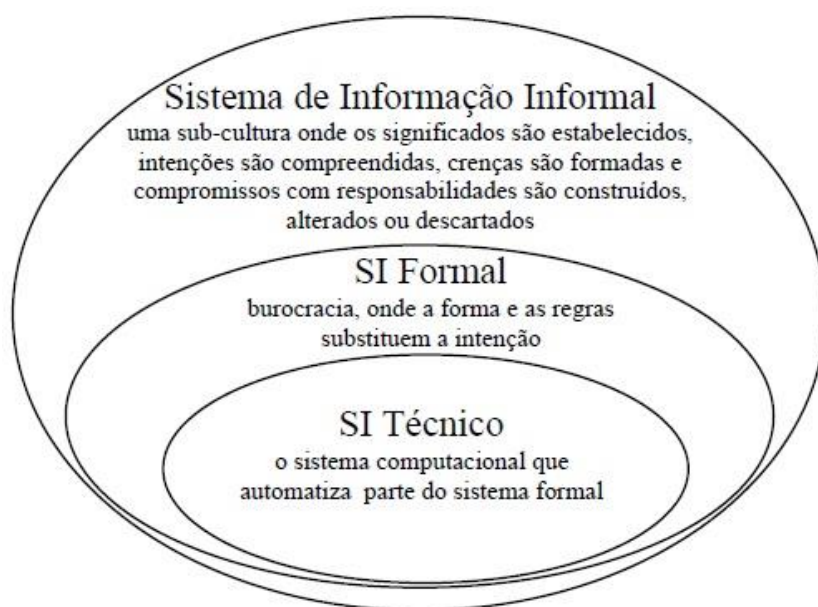


Figura 7 - Cebola semiótica (Bonacin, 2004, adaptado de Liu, 2000)

O uso do SAM tem como objetivo descrever uma visão dos agentes e seus padrões de comportamento em um domínio. Um agente pode ser definido como algo que é responsável por um comportamento. Nesse trabalho pode-se tomar como exemplo “aluno”, “professor” ou mesmo uma instituição. É importante notar que essa definição de agente da SO difere-se da definição de agente artificial comumente utilizado em inteligência artificial.

Os padrões de comportamento podem ser referidos como *affordances*. A análise semântica através do método SAM apoia-se nos conceitos de agente e *affordance* e tem

como objetivo produzir modelos semânticos e diagramas de ontologias para expressar o domínio do problema. Tais modelos elucidam a relação de agentes que possuem *affordances*, manifestados através de padrões de comportamentos, e que, ao mesmo tempo, estão sujeitos a *affordances* dos objetos e agentes com os quais interagem (Melo, 2003).

Affordance é um conceito introduzido por Gibson (1968), que pode ser utilizado para denominar a propriedade ou o comportamento de um objeto, elemento, sistema ou organismo combinada com as características de seu ambiente, e que permita a identificação de seu funcionamento ou ação dentro de um contexto. Em seu trabalho, Gibson pretendia descrever a relação do “animal”⁸ com seu ambiente. O conceito de *affordance* tem como função nomear alguma coisa que se refere tanto ao ambiente quanto ao animal, e a complementaridade entre eles. Gibson exemplifica essa complementaridade com o formato das superfícies terrestres. Dependendo de seu tipo, o animal ou o homem tem um tipo diferente de comportamento ou ação: andar em uma trilha na floresta, deitar em um gramado, nadar no mar, etc.

O conceito de *affordance* é um elemento presente na Análise Semântica definida no MEASUR. O conceito de Gibson foi expandido por Stamper para incluir invariantes que são percebidas no mundo social. No conceito original, o enfoque era a percepção do mundo físico; entretanto a noção de *affordance* adotada por Stamper foi expandida para incluir invariantes que nós percebemos no mundo social, incluindo conceitos abstratos como os relacionados com a programação. Liu (2000) reforça essa utilização na Análise Semântica, afirmando que *affordances* são construções sociais válidas em um certo contexto social.

Existem diversas dependências entre agentes e *affordances* em um determinado contexto. Um agente pode ser representado por uma pessoa, que por sua vez, depende de um representante mais complexo, por exemplo, sociedade. Essas dependências são mapeadas nesses modelos e diagramas. A análise semântica através do SAM apresenta outros conceitos importantes para a definição de modelos que representem o domínio mapeado (Liu, 2000; Melo, 2007):

⁸ Gibson usa o termo “animal” para descrever um organismo capaz de interagir com seu ambiente, sem distinção entre animal ou humano.

- **O mundo:** construção social através da interação dos agentes, em um determinado contexto;
- **Determinante** (*determiners*): invariante da quantidade e qualidade de agentes e *affordances*, que diferencia uma instância da outra;
- **Papel** (*role name*): determinante de individualidade, uma vez que um agente pode assumir vários papéis quando ele está envolvido em várias ações ou relações;
- **Relação genérico-específico** (*specifics*): ocorre com agente e *affordances* se eles possuírem propriedades comuns ou diferentes. Geralmente é determinada por normas que podem ser formadas socialmente ou culturalmente;
- **Ontologia:** agentes devem criar significados comuns para os signos, através de um conjunto de representações. Por exemplo, a comunicação humana depende de um vocabulário comum. Esse conjunto de signos que formam o vocabulário é chamado ontologia;
- **Dependência ontológica:** restrição de existência, formada quando um *affordance* só é possível quando existirem outros.

Os *affordances*, quando combinados com outros conceitos, formam um diagrama de ontologia, produzindo um modelo estável da organização, pouco suscetível a mudanças no contexto organizacional (Liu, 2000). Em um diagrama de ontologias, os agentes são representados por círculos, *affordances* por retângulos, dependências ontológicas por linhas da esquerda para a direita, papel (*role-name*) por semicírculos, parte todo por linhas com pontos pretos e os determinantes com “#” e linhas ligadas aos *affordances*.

O método NAM tem o foco nas normas sociais, culturais e organizacionais, que governam as ações dos agentes, e é utilizado para discutir aspectos dos níveis pragmático e social da organização. No nível pragmático, a Análise de Normas descreve o relacionamento entre o uso intencional dos signos para comunicação entre agentes e o comportamento resultante dos responsáveis no contexto, e no nível social, as normas expressam crenças, expectativas, compromissos, contratos, leis, culturas e também o negócio.

O conceito de normas define as responsabilidades dos agentes envolvidos em alguma tarefa, ou as condições sob as quais certas ações podem ou não ser realizadas. Normas correspondem no nível social à ideia de *affordance* no nível individual (Liu, 2000). Esse conceito baseia-se na teoria de normas criada por Stamper et al. (2000), que inclui: normas perceptuais, cognitivas, avaliativas, denotativas e comportamentais. De acordo com Stamper et al. (2000), normas podem ser representadas por todos os tipos de signos, sejam eles documentos, comunicação oral ou comportamento, com o intuito de preservá-las, difundi-las e segui-las.

3. A SEMIÓTICA ORGANIZACIONAL NO USO DE ROBÓTICA PEDAGÓGICA

Esta proposta de pesquisa objetiva a construção de uma metodologia para avaliação e mensuração da aprendizagem de conceitos de programação com o uso de RP via semiótica. Essa metodologia tem aplicação em um cenário no qual carece de métodos que possam apresentar e constatar o diferencial na percepção, processo cognitivo e avaliação (valores e estímulos) por parte dos alunos. A proposta se organiza em três bases teóricas e metodológicas para a construção da solução:

- **Robótica Pedagógica no ensino de Desenvolvimento de Algoritmos:** estudo da aplicação da RP como instrumento didático. Verificar o enquadramento dos recursos do robô para o ensino de conceitos fundamentais de programação;
- **Análise semiótica no contexto do aprendizado:** adequação dos métodos de análise semiótica em um ambiente de aprendizagem. Verificar, através da semiótica, a efetividade no aprendizado.
- **Modelo conceitual para desenvolvimento de práticas e avaliações:** especificação de um modelo que, através de meios sistemáticos, possa direcionar atividades práticas e uma estrutura de avaliação.

A Figura 8 apresenta através de uma visão macro, a relação entre as bases teóricas/metodológicas e como os resultados da proposta relacionam-se. Nesse diagrama, é possível identificar o que as diferentes bases provêm e como o modelo final produzirá um framework com o intuito de direcionar práticas pedagógicas e métodos de avaliação. Também nota-se como o uso de RP pode ser aplicado aliado aos métodos de avaliação de SO.

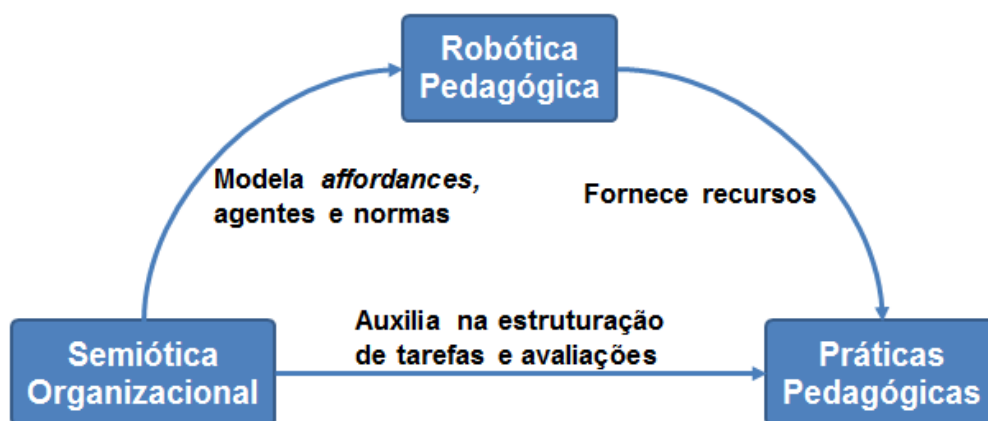


Figura 8 - Relação entre as bases teóricas/metodológicas

A Seção 3.1. destaca a aplicação da RP no ensino de programação. Ela tem como foco principal ilustrar como se pretende usar nesse trabalho a RP para criar um ambiente propício para práticas pedagógicas que possam auxiliar os alunos iniciantes na assimilação de conceitos fundamentais de Desenvolvimento de Algoritmos. A Seção 3.2 apresenta como os métodos de PAM, SAM e NAM do MEASUR serão utilizados para auxiliar na avaliação de um processo de aprendizagem e um exemplo de avaliação de uma prática através de Análise de Normas.

A Seção 3.3 apresenta o modelo conceitual proposto, com base nos conceitos discutidos nas Seções 3.1 e 3.2. O modelo tem como objetivo apresentar um guia para orientar práticas com objetivos de ensinar princípios fundamentais de Desenvolvimento de Algoritmos utilizando RP e também um *framework* de avaliação. Esse *framework* de avaliação será composto por uma análise do problema a ser respondido e avaliado; os *affordances* que o aluno tem que adquirir no processo de aprendizagem, e; a avaliação do processo cognitivo e do diferencial da percepção do aprendizado, através de normas.

3.1. A ROBÓTICA PEDAGÓGICA NO ENSINO DE DESENVOLVIMENTO DE ALGORITMOS

Em um ambiente tradicional de ensino de programação é muito comum utilizar fluxogramas e pseudocódigos para demonstração de conceitos abstratos, mas muitas vezes essa metodologia apresenta-se como uma abordagem ineficaz para sanar as principais dificuldades e dúvidas que os alunos possuem, *e.g.*, abstração, aplicação de conceitos em uma solução estruturada e resolução prática de problemas. A RP

proporciona melhores condições de ensino quando analisamos conceitos abstratos e aplicação de teoria. Com ela é possível observar o impacto real de instruções em um objeto concreto e, dessa maneira, associar os comandos abstratos aos movimentos e suas ações (Gomes, Henriques e Mendes, 2008).

A presença de um ambiente para PV baseado em uma interface de programação em blocos também pode ser destacada como uma vantagem para o ensino na área. A PV baseada em blocos, segundo Gomes, Henriques e Mendes (2008), aproxima os conceitos abstratos que envolvem a programação aos conceitos concretos de blocos, minimizando assim as dificuldades associadas a aprendizagem, como a complexidade da sintaxe e conceitos abstratos.

Segundo Utting et al (2010) na fase inicial de aprendizagem do aluno, o uso da PV é uma boa solução, por permitir que ele consiga ver o resultado das instruções que organizou de forma mais instantânea e clara. O uso da PV nesse trabalho tem como objetivo trazer um ambiente de programação, no qual o foco é uma sintaxe visual (não textual, que utiliza elementos gráficos) que possa, além de cativar o aluno, ser mais representativa, intuitiva e natural.

Os conceitos que apoiam o trabalho e o desenvolvimento de alunos durante as aulas podem ser potencializados com o uso da RP. Conforme Zili (2004), a utilização da RP contribui para o desenvolvimento de competências necessárias para alunos iniciantes em programação, como: desenvolvimento do raciocínio lógico; representação e comunicação; resolução de problemas por meio de erros e acertos; aplicação das teorias formuladas em atividades práticas, e capacidade crítica.

Para aplicação da RP como instrumento didático faz-se necessário associar os conceitos mais significativos e fundamentais no ensino de programação com os recursos de programação oferecidos pelo robô e sua linguagem de programação. Segundo Brennan e Resnick (2012), há sete conceitos computacionais comuns entre as linguagens de programação do paradigma procedimental: sequências, loops, eventos, paralelismo, condicionais, operadores e dados. Para Forbellone e Eberspächer (2005), fundamentalmente, a lógica de programação e a composição de um algoritmo se agrupam em: (1) tópicos preliminares ou fundamentais (*e.g.*, tipos primitivos, variável, expressões aritméticas, expressões lógicas, comando de atribuição e comandos de

entrada e saída); (2) estruturas de controle (*e.g.*, estruturas de seleção e estruturas de repetição) e; (3) modularização de algoritmos (módulos).

Para esse trabalho, serão adotados três grupos de conceitos fundamentais para o ensino introdutório de Desenvolvimento de Algoritmos, em disciplinas comumente denominadas como “Lógica de Programação”, “Algoritmos e Programação”, “Introdução à Programação”, entre outras, abrangendo as diretrizes curriculares vigentes pelo Ministério da Educação – Conselho Nacional de Educação (MEC/CNE), que estabelecem os currículos dos cursos de graduação na área de Computação (MEC/CNE, 2014A)⁹ e dos cursos técnicos nas áreas de Informática (MEC/CNE, 2014B)¹⁰. Esses três conceitos são:

- **Estruturas de seleção:** servem para agrupamento de comandos e sua execução, dependente de um termo condicional. Em pseudocódigo, um exemplo seria o termo *se-senão* (Forbellone e Eberspächer, 2005).
- **Estruturas de repetição:** agrupamento de comandos, com objetivo de gerar repetição ou ciclos, tendo a interrupção determinada por um termo condicional. Em pseudocódigo, temos como exemplos os termos *enquanto* ou *repita-até* (Forbellone e Eberspächer, 2005).
- **Módularização:** unidade criada para resolver a parte específica de um problema, composto por um grupo de comandos. Módulos podem ser requisitados através de chamadas por outras partes da solução. A técnica de modularização tem como objetivo de dividir o problema em partes menores e coerentes (Forbellone e Eberspächer, 2005).

3.2. ANÁLISE SEMIÓTICA NO CONTEXTO DO APRENDIZADO

Compreendendo o processo de ensino-aprendizagem como uma organização (*i.e.*, organização em seu sentido amplo como comportamento organizado), a SO pode auxiliar na avaliação criteriosa através de seus métodos. Para uma modelagem inicial na formulação de práticas que atendam aos critérios de avaliação e premissas do ensino, o método PAM se mostra adequado. Com o SAM é possível trabalhar com aspectos de

⁹ Relatório disponível no portal oficial do Ministério da Educação (<http://portal.mec.gov.br/>) com aprovação no dia 09/03/2012.

¹⁰ Documento disponível no portal oficial do Ministério da Educação (<http://portal.mec.gov.br/>) com aprovação no dia 05/10/1999.

linguagem (semântica), padrões comportamentais e definição de *affordances* (ontologia). Para o nível pragmático e social de avaliação, o método NAM possibilita a especificação de padrões comportamentais em relação a aprendizagem, provendo resultados que descrevam a efetividade do processo e suas falhas.

3.2.1. FORMULAÇÃO DAS PRÁTICAS – MÉTODO PAM

Na fase inicial da elaboração de um conjunto de práticas pedagógicas deve ser feito um levantamento de quais conceitos são abordados e de que maneira serão trabalhados. Essa definição depende de uma análise do que é esperado como resultado dessa prática e o que os alunos deverão executar. O PAM fornece um conjunto de métodos que podem ser aplicados nessa fase inicial de planejamento e desenvolvimento, quando a definição de problema ainda é muito vaga e de natureza complexa, auxiliado na definição pelas partes interessadas (alunos e professor), utilizando o *framework* semiótico.

O PAM inclui a análise e validação dos *stackholders* (partes interessadas do domínio). No contexto em estudo, podemos colocar o aluno, como um desses *stackholders*, definindo quais devem ser os elementos necessários no início da prática, qual é o objetivo ao final de sua conclusão e os possíveis *affordances*. Os métodos que compõem o PAM fornecem recursos para uma articulação inicial do problema de maneira formal. Para esse trabalho, tais métodos contribuirão da seguinte maneira:

- **Definição da Unidade do Sistema (*Unit System Definition*):** auxilia na definição principalmente dos interesses dos alunos (e outras partes interessadas) e todo o trajeto de resolução esperado;
- **Quadro de Valoração (*Valuation Framing*):** torna possível identificar a trajetória esperada na resolução de tarefas;
- **Análise Colateral (*Collateral Analysis*):** possibilita definir as etapas para resolução de problemas e suas variações;
- **Morfologia do Sistema (*System Morphology*):** os três diferentes tipos de comportamento - substantivo, comunicação e controle - são trabalhados de maneira unificada, auxiliam a definição de *affordances* e a sua apropriação pelos agentes.

3.2.2. DESCRIÇÃO DO DOMÍNIO – MÉTODO SAM

Para a análise de domínio, é necessário levantar requisitos para a formação das práticas pedagógicas de maneira sistemática e verificar aspectos relacionados à semântica da linguagem utilizada para comunicação entre os principais envolvidos e seus padrões comportamentais. A utilização do método SAM neste trabalho tem como objetivo identificar os agentes e os *affordances* do domínio, assim como as dependências no contexto de práticas, utilizando RP para ensino de programação de computadores.

O SAM tem como foco os agentes e seus padrões de comportamento para descrever uma organização. Através dessa análise será possível demonstrar e avaliar a interação e os padrões de comportamento de alunos, professores e robôs em um ambiente de ensino-aprendizagem. Esses padrões serão especificados através de um diagrama de ontologia, que representa uma visão dos agentes responsáveis no domínio em questão e seus comportamentos e ações (*affordances*). Em um modelo semântico, o significado das palavras no ambiente é visto como o relacionamento entre os signos e decorrência das ações apropriadas.

O modelo semântico e os conceitos da AS adotados neste trabalho são baseados no trabalho de Liu (2000) e apresentados a seguir:

- **Mundo:** construção social feita através das ações dos agentes, tendo como base o que é oferecido pelo mundo físico para ele próprio;
- **Affordance:** conceito introduzido por Gibson (1968), que pode ser utilizado para expressar invariantes de repertórios de comportamento de um organismo possíveis pela estrutura do organismo combinada com a de seu ambiente.
- **Agente:** tipo especial de *affordance*, que pode ser definido como alguma coisa que tem um comportamento responsável. Um agente pode ser um indivíduo, um grupo cultural, uma comunidade, sociedade, etc. (uma pessoa, um departamento, uma organização, etc. podem ser agentes);
- **Determinante:** invariante de quantidade e qualidade de agentes e *affordances*, pelos quais podemos diferenciar uma instância da outra;
- **Ontologia:** vocabulário com significados comuns para os signos;

- **Dependência ontológica:** dependência que ocorre quando um *affordance* só é possível com a existência de outros *affordances*.
- **Especialização:** estruturas genérico-específicas entre agentes e *affordances*, criadas de acordo com as propriedades que eles compartilham ou não;
- **Parte-todo:** quando um agente ou *affordance* pode ser parte de outro agente ou *affordance*;
- **Papel:** papel específico que um agente pode desempenhar em função de um *affordance* que ele possui;
- **Responsabilidade:** autoridade dada a um agente responsável pelas ações e trocas de estados em relação aos *affordances*.

3.2.3. AVALIAÇÃO POR NORMAS – MÉTODO NAM

Como referência para avaliação de resultados será adotada uma perspectiva orientada a normas de aspectos pragmáticos, sociais e comportamentais, formando um modelo sistemático de avaliação. Esse modelo incorpora diferentes tipos de normas em diferentes tipos de camadas semióticas. Neste trabalho cada um dos tipos de normas terá um objetivo específico no processo:

- **Normas perceptuais:** ajudam a compreender como a pessoa (aluno) consegue identificar as diferentes ações executadas pelo robô e aos processos de programação envolvidos. O robô proporciona uma gama de reações que podem representar diferentes fenômenos envolvidos na programação.
- **Normas cognitivas:** a análise sob a ótica das normas cognitivas possibilita verificar aspectos de compreensão do aluno das ações programadas e as respostas dadas pelo robô. A tarefa de programar exige um alto nível de cognição, envolvendo fatores como atenção, percepção, raciocínio e linguagem para desenvolver soluções. Com uso de robôs, as respostas dadas podem ser mais representativas, auxiliando na interpretação e compreensão de conceitos. Por exemplo, o ato de encontrar um obstáculo e o mesmo ser evitado pelo robô, demonstra a

“realidade” por trás de uma estrutura condicional, atuando, assim, como uma ação repetitiva que pode representar a ação de um laço de repetição;

- **Normas avaliativas:** demonstram as expectativas e as respostas dadas pelos robôs em sua programação diante do contexto. Essas normas estão relacionadas a escolha das funções corretas em determinada situação para atingir seu objetivo, desenvolvendo uma coerência pragmática na construção da solução;
- **Normas denotativas:** serão importantes para a verificação da compreensão e escolha correta dos elementos sintáticos (regras dos componentes e funções de programação) e sua aplicação semântica (o uso correto da programação);
- **Normas comportamentais:** toda prática de programação exige uma sequência de ações para atingir um objetivo. Nesse trabalho, as normas comportamentais definem os comportamentos dos alunos diante das adversidades encontradas no momento da resolução de um problema.

Estas normas podem delinear o comportamento do aluno e demais agentes envolvidos na pesquisa e deverão resultar em requisitos-chaves de aprendizagem a serem explorados nas práticas durante o desenvolvimento do projeto, assim como elementos a serem analisados posteriormente.

As normas comportamentais descrevem como um agente pode julgar e se comportar perante uma situação, podendo expor como as decisões foram tomadas para ações, incluindo aspectos ligados à autoridade e à delegação de responsabilidades, que podem ser descritas através da lógica Deontica. Tal formato se apresenta como:

<Norma> ::= sempre <condição> se <estado> então <agente> é <D> fazer <ação>

, onde <Norma> corresponde ao nome da norma. O campo <condição> especifica uma certa condição em que a norma é aplicada em um determinado estado <estado>. O campo <agente> especifica qual agente tem a obrigação, permissão ou proibição, de acordo com o operador deontico <D>, de realizar uma determinada ação <ação> (Bonacin, 2004).

4. PROCESSO PARA DESENVOLVIMENTO E AVALIAÇÕES DE PRÁTICAS USANDO ROBÓTICA PEDAGÓGICA - PDAP-RP

Primeiramente, com o objetivo de unificar as ferramentas com as bases teóricas e metodológicas, foi desenvolvido um modelo conceitual (*framework*). Este modelo serviu como base para definição de uma metodologia eficaz de desenvolvimento de práticas, avaliação e mensuração da aprendizagem de conceitos de programação, utilizando a RP, fundamentado em métodos da SO (*i.e.*, PAM, SAM e NAM).

Como apresentado anteriormente, cada método tem foco em um aspecto específico relacionado ao problema da pesquisa. Nesse capítulo será apresentada a metodologia proposta, com destaque ao processo que estrutura as atividades envolvidas, “Processo de Desenvolvimento e Avaliações de Práticas usando Robótica Pedagógica – PDAP-RP.

A Figura 9 ilustra a relação das áreas abordadas na pesquisa e suas contribuições para elaboração e análise de práticas pedagógicas. Em uma visão macro do modelo, é possível compreender que os métodos da SO têm como objetivo fornecer os recursos necessários para identificar aspectos comportamentais, pragmáticos e semânticos envolvidos em um cenário de ensino/aprendizagem. A RP contribui com ferramenta que possibilita a construção e aplicação das práticas pedagógicas.

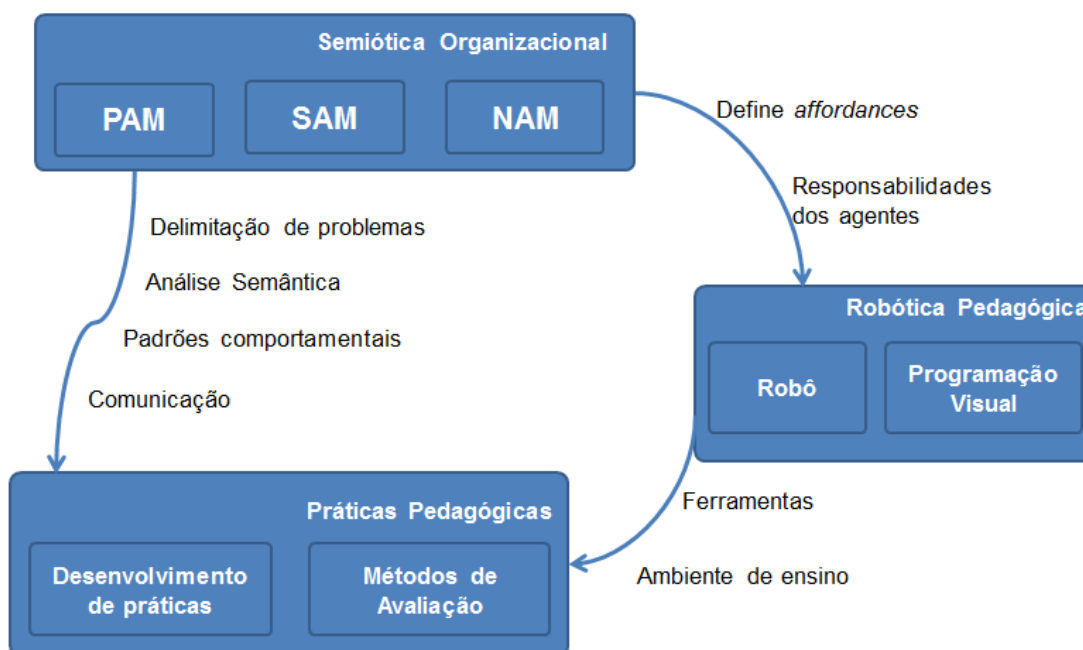


Figura 9 – As relações das áreas abordadas na pesquisa

O método PAM está relacionado diretamente à fase inicial de delimitação de problemas referente à especificação de uma prática. Com isso, é possível utilizar meios para definir os conceitos a serem trabalhados em um cenário que ainda se encontra vago e por formalizar.

Com o método SAM, é possível analisar a semântica envolvida nas práticas. O SAM tem como objetivo principal identificar *affordances*, assim como as dependências ontológicas e outros aspectos importantes do contexto de ensino/aprendizagem.

Por fim, o método NAM é usado para a especificação de normas, as quais definem uma especificação formal do comportamento esperado do aluno, incluindo o que deve ser compreendido e realizado por ele. As normas são utilizadas neste trabalho para formalizar e especificar diversos cenários, os quais envolvem conceitos fundamentais de programação e sua aplicação com o uso da RP.

Para articular os métodos da SO e RP, com base nas relações apresentadas na Figura 9, foi criado o PDAP-RP. A elaboração do método partiu de estudos preliminares em Zanetti e Bonacin (2014A), que exploraram a aplicação do *framework* semiótico e a análise de normas para a descrição da atividade e seus elementos a serem acompanhados no desenvolvimento da prática de ensino com RP. Neste contexto, são entendidos “elementos” como os objetos necessários, podendo ser físicos (*e.g.*, arena, obstáculos, faixas) e conceituais (*e.g.*, conceitos de programação, objetivos, problemas, etc.).

Esse estudo preliminar foi expandido no trabalho de Zanetti e Bonacin (2014B), onde foi proposta de maneira mais ampla a sistematização do emprego da SO em um processo, definindo assim a primeira versão completa do PDAP-RP. A Figura 10 apresenta uma visão geral do PDAP-RP, apresentando a aplicação de métodos na visão da SO sobre um ambiente de ensino como um sistema social organizado. Na Figura 10 aparecem quatro áreas com linhas tracejadas, que indicam as quatro fases distintas do PDAP-RP. Cada uma dessas fases está detalhada nas seções a seguir.

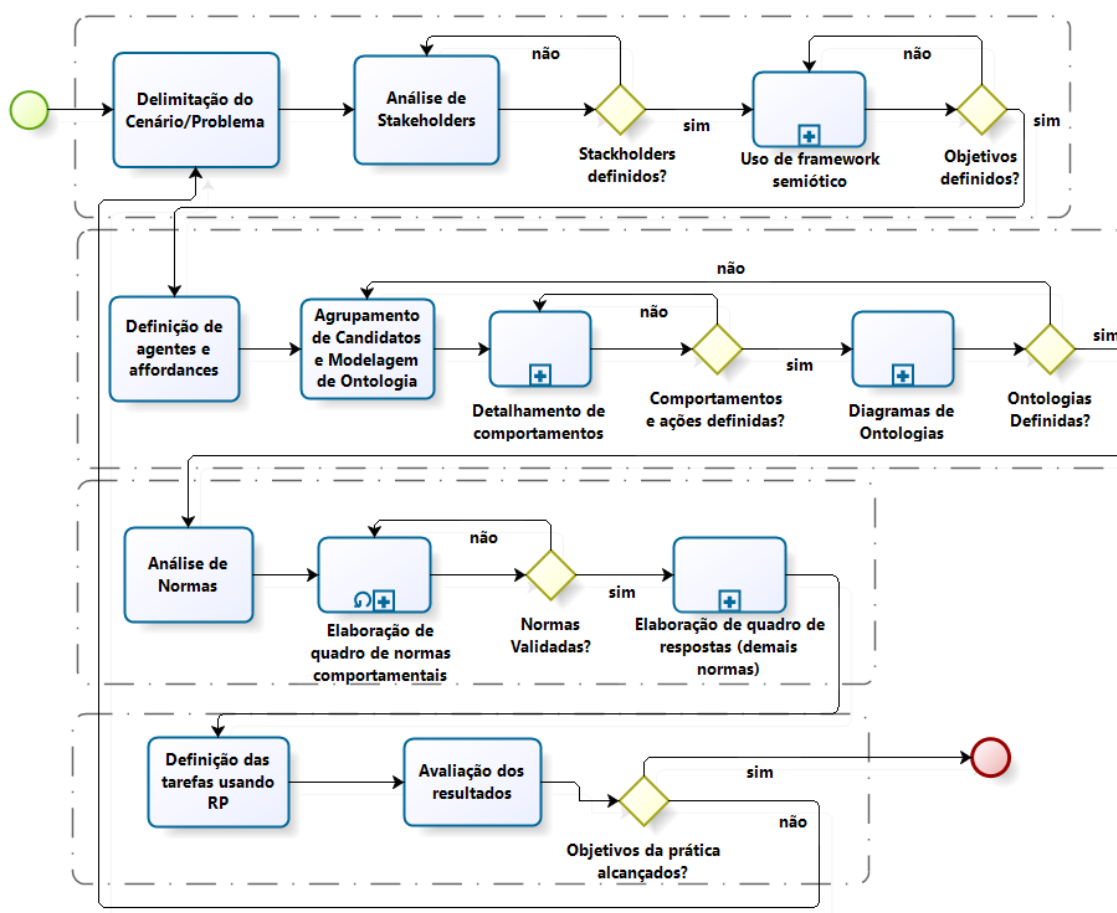


Figura 10 – Visão geral do PDAP-RP

4.1. FASE I – IDENTIFICAÇÃO DO CENÁRIO E PROBLEMAS

O objetivo dessa fase é a identificação precisa do cenário e do problema. Por cenário é compreendido como a soma dos elementos e ambientes necessários para o desenvolvimento da prática. O problema inclui as dificuldades, desafios e conhecimentos necessários na obtenção de um determinado objetivo. Nessa fase o professor-pesquisador faz um levantamento inicial dos recursos existentes e dos objetivos que devem ser alcançados, ou seja, quais conceitos de programação deverão ser compreendidos ao final da prática e quais as pessoas que estarão envolvidas neste processo.

Conforme apresenta a Figura 10, esta fase é composta por três etapas: Delimitação do Cenário/Problema (seção 4.1.1); Análise de *Stackholders* (seção 4.1.2) e o Uso do *Framework* Semiótico (seção 4.1.3), que analisa os problemas e as soluções de acordo a cada nível do *framework*.

4.1.1 DELIMITAÇÃO DO CENÁRIO/PROBLEMA

Nesta etapa é realizada a descrição do cenário e o levantamento dos principais problemas e dificuldades, assim como a descrição inicial do contexto. Essa descrição pode ser feita em formato textual de maneira direta e objetiva, ressaltando quem são as pessoas envolvidas, quais são os elementos do cenário, conceitos envolvidos e os objetivos gerais. É sugerida a seguinte estrutura para o texto:

- Definição dos envolvidos: descrição breve e sucinta das pessoas e das características mais relevantes (*e.g.*, perfil do aluno, conhecimento técnico do professor).
- Elementos do cenário: elementos pertinentes e relevantes ao cenário da prática. A descrição do cenário tem como objetivo listar brevemente os recursos físicos necessários e sua organização
- Conceitos envolvidos: os conceitos que serão trabalhados com a prática. Listar os elementos relacionados ao desenvolvimento de algoritmos que deverão estar presentes e serão utilizados pelos alunos na resolução da tarefa, bem como a maneira que eles serão ensinados.
- Objetivos gerais: descrição dos objetivos esperados e que serão avaliados ao final da execução da tarefa. Serve como uma referência inicial para o processo, mas pode ser revisitado, caso as avaliações precisem ser melhoradas ou redefinidas, durante a aplicação do PDAP-RP.

4.1.2 ANÁLISE DE *STACKHOLDERS*

Nesta etapa ocorre a análise e a identificação dos *stackholders* envolvidos (*e.g.*, aluno, professor) através da “cebola semiótica” (Liu, 2000). Esta atividade deverá ser realizada até que exista consistência na análise dos *stackholders*, como a definição dos papéis do “professor” e “alunos”. Nesse contexto, além de aplicar a prática, o professor tem o papel de facilitador e orientador. Os alunos têm como objetivo a resolução do problema proposto, identificação da solução e execução através da programação do robô. Como mostra a Figura 11, a “cebola semiótica” organiza os problemas relacionados à prática e o envolvimento dos *stakeholders*, sendo categorizados de acordo

com os níveis técnico (*e.g.*, manuseio e comunicação), formal (*e.g.*, procedimentos de ensino) e informal (*e.g.*, cultura do ambiente).

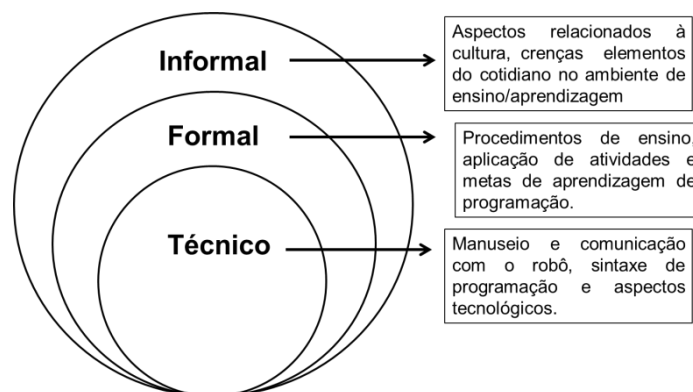


Figura 11 – Cebola Semiótica

Os signos e o conhecimento dos níveis Informal e Formal do domínio são necessários para a formação dos artefatos (recursos a serem disponibilizados aos alunos) e práticas do nível Técnico, criando uma associação de aspectos para compor um cenário de ensino e aprendizagem. Em cada um dos níveis são desenvolvidas análises, conforme descritas abaixo:

- **Nível Informal:** nível onde os significados e intenções das práticas são estabelecidos, convicções são formadas e os compromissos com as responsabilidades entre os envolvidos são estabelecidas. Nesse nível encontram-se relações informais entre alunos e professores, como “a vontade de aprender” e o “compromisso em ensinar”. Estas relações estabelecem compromissos e trazem valores para os envolvidos.
- **Nível Formal:** nível mais burocrático, onde as formas e regras são criadas. No contexto do ensino, são definidos os procedimentos e os meios de aplicação das atividades pedagógicas, assim como as metas a serem atingidas (como e o que o aluno deve aprender).
- **Nível Técnico:** uma vez definido o nível Formal, esse nível define como ocorrerá a comunicação, a manipulação e a programação do robô e demais aspectos tecnológicos (*e.g.*, uso de computadores, ambiente da prática).

4.1.3 USO DE *FRAMEWORK* SEMIÓTICO

A utilização do *framework* semiótico serve como guia para elicitación e análise de problemas e soluções nas práticas usando RP. A tarefa visa classificar e organizar os problemas e soluções relacionadas ao domínio dos objetivos didáticos do nível físico ao nível social de acordo com a escada semiótica (Liu, 2000). Essa classificação e organização serão repetidas para cada nível da escada semiótica e poderão ser criadas questões que ajudem a identificar os elementos do domínio. A Figura 11 mostra como o *framework* semiótico pode organizar em níveis aspectos da plataforma tecnológica (*e.g.*, recursos de hardware e software) e do sistema de informação humano (*e.g.*, atuação profissional do aluno).

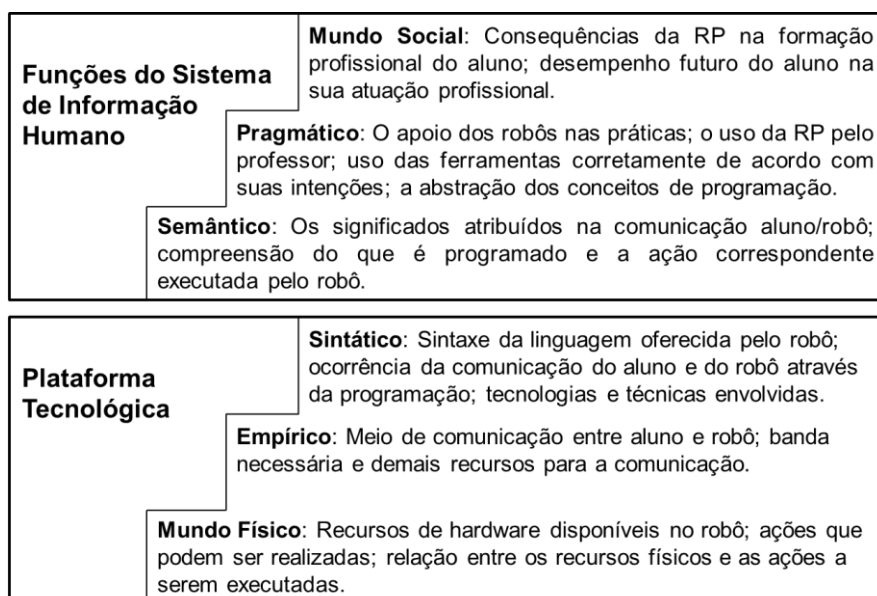


Figura 12 – Framework Semiótico

Cada um dos níveis da escada visa classificar e organizar problemas e soluções relacionados ao domínio dos objetivos didáticos. Como exemplo, no nível físico, são apontados os recursos necessários ao robô (*e.g.*, componentes estruturais, motores e sensores); e no pragmático, a listagem das principais soluções de programação necessárias que à abstração dos alunos (*e.g.*, estrutura de seleção, conceito de operadores lógicos, estruturas de repetição). Em uma prática envolvendo RP, podemos utilizar o *framework* com os seguintes aspectos em cada nível:

- **Mundo Social:** descrever as consequências da RP na formação profissional do aluno. Por exemplo, como o contato com essa tecnologia

pode contribuir para sua formação e no desempenho futuro do aluno na sua atuação profissional e acadêmica, bem como outras consequências sociais.

- **Pragmático:** as intenções e comunicações que envolvem apoio dos robôs nas práticas. Descrever como o uso do robô e das ferramentas deve ocorrer, de acordo com suas intenções e objetivos. Alinhar o uso das ferramentas com a abstração dos conceitos de desenvolvimento de algoritmos necessários para atingir os objetivos da prática.
- **Semântico:** os significados atribuídos na comunicação aluno e robô. Compreensão do que é programado e a ação correspondente executada pelo robô. Definição dos termos técnicos ou não técnicos e os objetos presentes nessa comunicação, que devem ser compreendidos pelo aluno e professor.
- **Sintático:** sintaxe da linguagem oferecida pelo robô. Elementos para a ocorrência da comunicação do aluno e do robô através da programação e tecnologias e técnicas envolvidas. Descrição das ações que podem ser executadas pelo robô e seu *feedback*.
- **Empírico:** meio de comunicação entre aluno e robô. Definir qual banda necessária e demais recursos para a comunicação no ambiente de ensino. Regras de como trabalhar com o robô na arena (*e.g.*, posicionamento do robô na arena, verificação das conexões físicas).
- **Mundo Físico:** recursos de hardware disponíveis no robô. Descrição detalhada dos limites dos recursos físicos e ações que podem ser executadas.

4.2. FASE II – DEFINIÇÃO E MODELAGEM DE AGENTES, COMPORTAMENTO E DEPENDÊNCIAS ONTOLÓGICAS

Durante esta fase o professor-pesquisador tem como objetivo definir quais serão os agentes e, principalmente, os *affordances* relacionados a esses. É realizada uma análise e modelagem dos *affordances* e dependências ontológicas e demais conceitos relacionados ao SAM. Em uma prática com RP, os agentes são, por exemplo, os alunos, o professor e a escola, e os *affordances*, a definição das invariantes de comportamento

dos agentes envolvidos, como o uso de recursos de programação (*e.g.*, estruturas de repetição, seleção, operadores lógicos) e a interação com o robô (*e.g.*, reconhecimento de obstáculo, movimentar-se).

Essa fase possui duas etapas fundamentais, a primeira é a Definição de Agentes e *Affordances* (seção 4.2.1) e a segunda é o Agrupamento de Candidatos e Modelagem de Ontologias (seção 4.2.2.). Esta última é constituída de duas subatividades, uma sendo o Detalhamento de Comportamento e Ações, e a outra, a construção dos Diagramas de Ontologias.

4.2.1. DEFINIÇÃO DE AGENTES E *AFFORDANCES*

Esta fase tem como objetivo definir de maneira ampla e clara os agentes e *affordances*, através de uma análise do domínio. Nesta etapa seguimos as fases da AS, conforme definido por Liu (2000). Essa análise é um método é um dos fundamentos da SO e tem como objetivo descobrir os requisitos do domínio, podendo ser sumarizado em quatro fases: (1) definição do problema; (2) geração de *Affordances* Candidatos; (3) agrupamento dos Candidatos e; (4) criação dos Diagramas de Ontologia.

A AS definida por Liu (2000) é apresentada na Figura 13 e pode ser descrita como:

- **Definição do problema:** descrição do problema, sendo um estudo complementar da definição do problema, podendo ser feita através de análise de documentos ou entrevistas;
- **Geração de *affordances* candidatos:** identificação inicial dos *affordances*, podendo criar-se uma lista com todas as palavras consideradas relevantes, algumas são deixadas de fora, mas podem ser incluídas no modelo posteriormente;
- **Agrupamento de candidatos:** categorização dos candidatos de acordo com os conceitos básicos da AS (i.e., agentes, *affordances*, determinantes, etc), podendo ser organizado em uma tabela simples. Pode-se utilizar pequenos diagramas ontológicos com o propósito de estudar os efeitos dos agrupamentos nas unidades semânticas;

- **Diagramação ontológica:** construção de um modelo completo, conectando os pequenos diagramas em um único diagrama de ontologia;

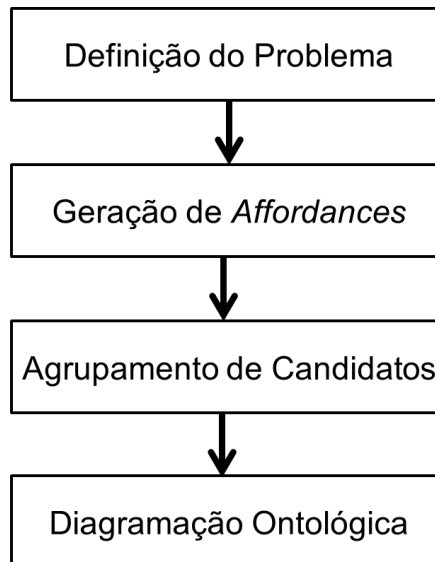


Figura 13 – Fases da Análise Semântica (adaptado de Liu, 2000).

A primeira fase dessa AS é contemplada na Fase I do PDAP-RP, que documenta o cenário e problema do domínio, assim como as pessoas envolvidas e suas participações. Esta análise de cenário pode ser feita através de um enunciado, que sintetize todo o domínio.

Na fase seguinte, a geração de *Affordances* Candidatos, é produzida uma lista com o vocabulário das unidades semânticas que poderão ser usadas para a descrição de agentes e seus padrões. Nesta etapa do PDAP-RP é feita uma descrição superficial e não exclusiva de todos os potenciais agentes, *affordances* e seus relacionamentos, que podem pertencer ao domínio do problema. Essa listagem não demanda nenhum tipo de organização formal, podendo ser aceita qualquer tipo de informação, sendo palavras, termos ou frases, que possam descrever um agente e seus papéis (e.g., alunos, professor, programador), *affordances* (e.g., “aprender programação”, “mover”, “seguir traçado”) ou demais relacionamentos ontológicos.

4.2.2. AGRUPAMENTO DE CANDIDATOS E MODELAGEM DE ONTOLOGIAS

Nesta etapa é executada uma análise das unidades semânticas, ou seja, primeiramente categorizam-se os candidatos como sendo agentes ou *affordances*, conforme o conceito de agrupamento de candidatos definido em Liu (2000). Após essa

categorização, é feita um detalhamento em relação aos comportamentos e às ações, aos determinantes e aos papéis (Detalhamento de Comportamentos e Ações, na seção 4.2.2.1.). Uma vez definido os agentes, *affordances*, determinantes e papéis, esses conceitos serão agrupados em um diagrama organizado a partir de suas dependências ontológicas, formando o Diagrama de Ontologia (seção 4.2.2.2.).

O agrupamento de candidatos será organizado em uma listagem tabular em duas colunas, com a primeira coluna contendo o conceito candidato, e na coluna ao lado sua categoria (agente, *affordance*, determinante ou papel) com uma breve descrição e com informações que possam explicar suas funções. Essa tabela define um início de análise de unidades ontológicas que pode ser refinada na próxima etapa, com o detalhamento de comportamentos e ações e que terá seu modelo final definido no Diagrama de Ontologia.

A subatividade de Detalhamento de Comportamentos e Ações tem como objetivo refinar os resultados da etapa anterior, listando as dependências existentes entre as unidades semânticas, a partir de suas funções, comportamentos e ações. Nesse ponto do processo é necessária uma análise mais consistente, para uma definição mais segura de quais conceitos serão agentes ou *affordances*.

Os conceitos definidos como *affordances* ainda podem ser classificados como determinantes, papéis ou subtipos, e a partir disso definir as dependências. É prioritário que não haja incertezas em relação às funções da unidade, mas podem ser feitas revisões, principalmente no início do mapeamento das dependências.

Uma maneira de auxiliar para minimizar as incertezas é criar estruturas ontológicas parciais, que possam auxiliar no estudo das dependências, através de uma representação sólida e coesa. Por exemplo, a Figura 14 ilustra uma compreensão parcial do modelo final, no qual o agente “Pessoa” pode assumir um papel de Professor na relação de dependência ontológica do *affordance* “Ensina” com a “Pessoa”, *i.e.*, o *affordance* “ensina” só existe enquanto existir pessoas, e damos o nome de “Professor” ao papel da pessoa que ensina.

A definição desses modelos parciais pode levar a uma análise mais criteriosa e, uma vez concretizada essa dependência, pode ser agregado ao modelo maior, ou até mesmo, levar a uma visão de outras dependências. Outro benefício desses modelos parciais é a possibilidade de agrupar unidades correlacionadas, uma vez definidas suas

dependências. Por exemplo, criar modelos parciais para o mapeamento do papel “Aluno” e agrupá-los, formando a uma solução parcial com um maior contexto.

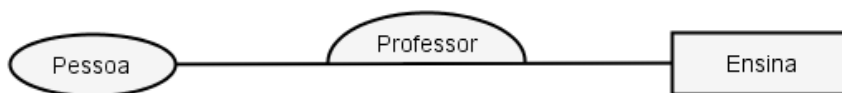


Figura 14 – Diagrama de ontologia parcial

Na subatividade de construção de Diagramas de Ontologias o objetivo principal é a criação de um modelo formal através de uma representação gráfica, envolvendo os agentes, *affordances* e suas dependências ontológicas. Depois da análise dos candidatos, da definição de dependências e dos agrupamentos por contextos, todas as estruturas ontológicas podem ser integrados para formar um Diagrama de Ontologia, ampliando suas conexões e corrigindo possíveis equívocos e duplicidades de conceitos.

Segundo Liu (2000), o modelo deve representar todos os princípios ontológicos e contemplar o domínio estudado por completo. Sempre se tem o agente “raiz” como sendo a sociedade, onde todos os membros compartilham alguns conceitos e culturas fundamentais, caracterizando o princípio de uma organização. A Figura 15, a seguir, ilustra um conjunto de mapeamento de dependências ontológicas integradas, descrevendo o domínio do problema. O objetivo desse modelo é fornecer um meio de comunicação e demonstrar alguns conceitos para os alunos. Os principais elementos (agentes, *affordances* e dependências ontológicas) desse modelo são:

- Sociedade: agente raiz, onde todos agentes e “*affordances*” são dependentes existencialmente/ontologicamente dele;
- Pessoa: agente bastante representativo no modelo, no contexto em questão, assumindo os papéis de professor e aluno. Professor pode ser compreendido como a pessoa responsável pelo ensino e aluno como a pessoa responsável por aprender a programar e pela programação do robô (no papel de Programador).
- Fabricante: agente responsável pela produção dos robôs;

- Robô: *Affordance* do Fabricante, sendo totalmente dependente por sua fabricação. O robô é composto por diversos recursos, como descrito na Seção 2.1.1.;
- Ensina: *Affordance* que qualifica as definições feitas pelo Professor na escolha dos conceitos de programação que serão ensinados;
- Define Prática: *Affordance* ontologicamente dependente de Professor e Robô, ou seja, não é possível sem a presença de um professor e os recursos provenientes dos robôs para a definição de uma prática. O determinante domínio provê a descrição feita pelo professor em análise aos recursos disponibilizados pelo robô;
- Programação Robô: este *Affordance* agrega os recursos disponíveis para a programação, dependentes do Robô e do aluno no papel de Programador, que tem como tipo específico a Programação Visual;
- Aprende Programar: *Affordance* responsável pela definição dos conceitos de programação que serão envolvidos no processo de ensino/aprendizagem;
- Executa Programação: *Affordance* que tem como responsabilidade delimitar todas as ações possíveis ao programar o Robô. São tipos de execução de programação: “Desvia Obstáculo”, “Identifica Traçado”, “Identifica faixa” e “Movimenta Robô”;
- Avalia: *Affordance* ontologicamente dependente de Professor e Executa Programação, que analisa as execuções de programações realizadas e as avaliações pelo Professor e executa a avaliação através de meios avaliativos;

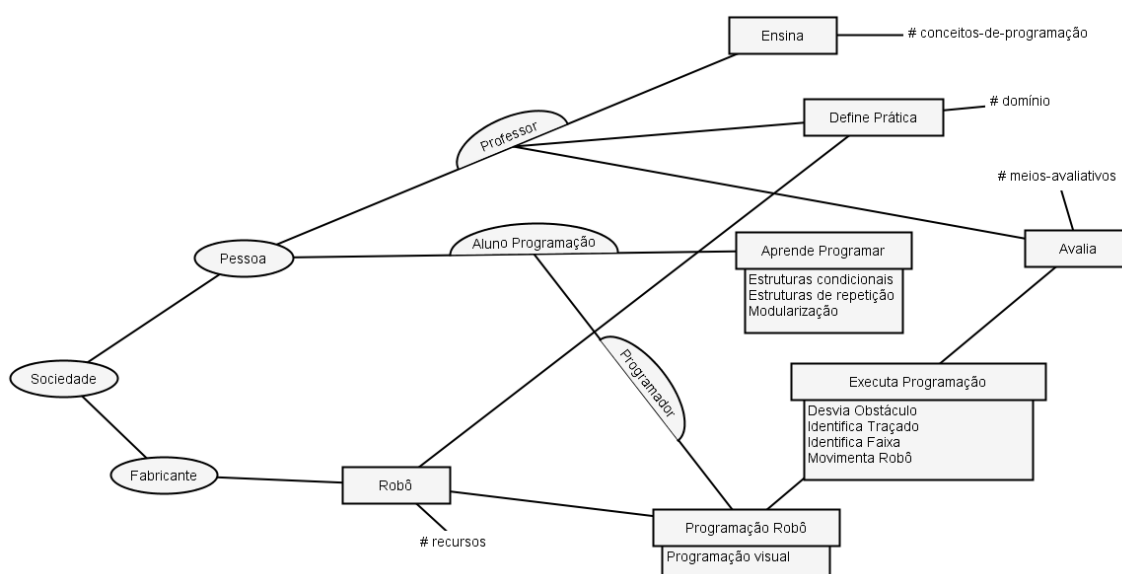


Figura 15 – Diagrama de Ontologia com visão parcial do domínio

4.3. FASE III – ANÁLISE E MODELAGEM DE NORMAS

Esta fase tem como objetivo analisar todo o tipo de norma, sendo primeiramente modeladas as normas comportamentais, que irão compor os quadros de descrição das tarefas a serem realizadas pelos alunos. Estes quadros servem de guia para interpretação das demais normas, no quadro de respostas. As normas em RP referem-se principalmente às relações dos agentes (*e.g.*, aluno) com seus *affordances* (*e.g.*, estruturas de programação). Para tanto, é realizada uma especificação formal, a qual apresenta o domínio em questão, as responsabilidades e os comportamentos de todos os elementos envolvidos. Grande parte dessas relações dizem respeito ao aluno, o robô e à lógica de programação, incluindo aspectos relacionados aos recursos de hardware disponível, a comunicação e a linguagem usada entre aluno e robô, os significados envolvidos, entre demais aspectos representativos.

Esta fase é constituída de uma etapa macro, como outras duas etapas agregadas. A etapa de Análise de Normas (seção 4.3.1.) inicia-se investigando as normas comportamentais em relação aos agentes e *affordances*. O produto dessa investigação induz a elaboração do Quadro de Normas Comportamentais, com uma descrição das características comportamentais. A partir dos quadros de normas comportamentais, que conduzem os meios de avaliação, é realizada a elaboração de Quadros de Respostas (demais normas) com as percepções dos alunos em relação aos aspectos perceptuais, cognitivos, denotativos e avaliativos, caracterizando as demais normas.

4.3.1 ANÁLISE DE NORMAS

Esta etapa inclui identificação, modelagem e análise das normas que são executadas durante a prática pedagógica, em que serão analisadas e descritas normas que representam o comportamento dos agentes no nível pragmático e social (Stamper et al., 2000). A descrição no nível pragmático possui foco na comunicação entre os agentes e seus comportamentos, enquanto no nível social essas normas irão expressar compromissos, expectativas e crenças relacionadas ao contexto social. Esta etapa é composta por outras duas, descritas a seguir.

A subatividade “Elaboração do Quadro de Normas Comportamentais” está fundamentada na aplicação do método NAM, associando as normas comportamentais aos *affordances*. Como produto desta etapa, serão elaborados quadros descritivos dessas normas, contendo os agentes envolvidos, os *affordances* e o detalhamento da norma através do uso da Lógica Deôntica, usada formalmente para analisar proposições acerca de uma norma.

Além da descrição através da Lógica Deôntica, o quadro conta com uma descrição dos *affordances* e dos agentes em um cenário definido por uma norma. O quadro tem o formato mostrado na Tabela 1. Nessa fase o objetivo é definir essas normas através do quadro, proporcionando uma maneira organizada e bem definida da coleção de normas envolvidas no processo de desenvolvimento e solução da prática com RP.

Tabela 1 – Exemplo de Quadro de Normas Comportamentais

Norma	{nome da norma}
Agentes	{lista de agentes}
<i>Affordances</i>	{lista de <i>affordances</i> }
Detalhamento da norma	{expressão em Lógica Deôntica}

Uma vez definidas as normas comportamentais, ocorre a subatividade de “Elaboração de Quadro de Respostas” (demais normas), cujo objetivo é a criação de quadro de respostas com os demais tipos de normas. Nesse quadro serão listados o resumo das respostas dos alunos e suas impressões aos aspectos perceptuais, cognitivos,

denotativos e avaliativos. O objetivo é identificar se todas as ações foram feitas corretamente e se o que foi percebido e compreendido foi o planejado inicialmente.

A organização desses quadros podem trazer uma percepção da compreensão do exercício da prática e os conceitos aprendidos. Tais quadros servem como guias de avaliação e sua condução para criação é orientada pelas normas que serão avaliadas para a criação do questionário. Em relação às normas cognitivas, pode-se criar um questionário que conduza as repostas dos alunos, por exemplo, perguntando se estão claros quais foram os conceitos de desenvolvimento de algoritmos utilizados, ou, em relação às normas perceptuais, se as ações percebidas dos robôs são condizentes à programação executada. O quadro tem formato tabular, com as colunas sendo as respostas das questões em relação às normas, já as linhas organizam as respostas de cada um dos diferentes alunos. A Tabela 2 exemplifica o layout básico de um Quadro de Resposta.

Tabela 2 – Layout do Quadro de Resposta

{Tipo da Norma}				
	Questão 1	Questão 2	...	Questão N
Aluno 1			...	
Aluno 2			...	
Aluno ...			...	

4.4. FASE IV – APLICAÇÃO DE PRÁTICAS

Durante esta fase, é aplicada a prática pedagógica em si. Para tanto, é definido qual o melhor ambiente (*e.g.*, uma arena com obstáculos, marcação com fitas) de aplicação da prática de RP proposta. Essa etapa também engloba a avaliação e verificação dos resultados, que podem levar a uma revisão de outros pontos da metodologia e correções nos modelos produzidos em um processo iterativo.

A aplicação de práticas é dividida em duas etapas, executadas em sequência: (1) a Definições de Práticas usando RP (seção 4.4.1), que é utilizada para a descrição e organização da prática pedagógica e; (2) a Avaliação de Resultados, fase de utilização dos meios avaliativos e verificação dos resultados obtidos com a prática.

4.4.1 DEFINIÇÕES DE PRÁTICAS USANDO RP

Nesta etapa é definida a composição do ambiente da aplicação da prática (arena e demais elementos físicos que a compõe) e os recursos que o robô deverá possuir, como, por exemplo, quais sensores usar e como deve ser sua estrutura física. Para essa definição da prática, são usados todos os artefatos criados anteriormente no PDAP-RP, a fim de criar uma prática, que contemple todos os aspectos elencados até o momento. Para a criação da tarefa são necessários três elementos: (1) enunciado da prática; (2) regras para solução e; (3) objetivos a serem alcançados. Estes elementos são descritos a seguir:

- **Enunciado da prática:** texto que descreve as características e elementos da arena, assim como quais são as ações do robô em relação ao ambiente. Por exemplo, se houver um trecho no qual o robô deve seguir uma linha preta de um ponto A ao ponto B, a descrição desse evento pode ser: “o robô deve partir do ponto A, encontrar e seguir a linha preta utilizando o sensor de luminosidade até encontrar o ponto B”.
- **Regras para solução:** descrição textual das regras e limitações do uso do robô e, conseqüentemente, dos recursos permitidos e/ou necessários para execução da prática. Essa descrição definirá os elementos (*e.g.*, quais sensores serão usados) e a montagem do robô, podendo ser de maneira objetiva e direta (*e.g.*, deverá ser usado o sensor ultrassônico), ou subjetiva (*e.g.*, deverá ser usada um sensor para identificação de obstáculos”), para que os alunos decidam o sensor a ser utilizado a partir do enunciado.
- **Objetivos a serem alcançados:** descrição clara dos objetivos esperados na execução da prática. Esses objetivos podem ser definidos por trechos da prática ou em sua totalidade. A presença da descrição dos objetivos possibilita ao aluno compreender o que se deve alcançar e quais são as etapas da solução a serem desenvolvidas.

4.4.2 AVALIAÇÃO DE RESULTADOS

Assim como ocorre em processos de avaliação em qualquer prática pedagógica, o professor-pesquisador observa e determina, através dos pontos definidos nos quadros de normas (Tabela 1) se as ações desenvolvidas pelos alunos e pelo robô atendem o objetivo. A utilização dos questionários e entrevistas complementam essa avaliação, que é sintetizada através do quadro de respostas (Tabela 2). Os resultados das avaliações podem ser apresentados em forma de relatório, incluindo os artefatos utilizados.

Para a verificação da execução definidas pelas normas comportamentais, podemos usar um *checklist*, contendo a norma e qual foi seu nível de execução. Como auxílio da análise e posterior comparação e refinamento das normas, usa-se uma escala na qual possa ser possível identificar ou não o cumprimento da norma. A escala é composta pelos termos: (1) “não cumpriu”, em caso de não cumprimento das ações esperadas; (2) “cumpriu parcialmente”, em caso de o aluno efetuar parte do que era esperado ou ter apresentado algum tipo de problema que não tenha levado ao cumprimento pleno, e; (3) “cumpriu plenamente”, no caso de cumprimento pleno da norma.

Em relação à análise dos quadros de resposta, eles auxiliam as respostas dos alunos às questões elaboradas sobre os aspectos relativos às normas perceptuais, cognitivas, avaliativas e denotativas. Ao analisar essas respostas, o professor-pesquisador terá a possibilidade de uma compreensão dos conceitos esperados e uma percepção consistente quando considerado a utilização dos conceitos de desenvolvimento de algoritmos, além de efeitos na utilização da RP.

5. APLICAÇÃO DE PRÁTICAS COM MÉTODO PDAP-RP

Este capítulo detalha a aplicação de práticas pedagógicas usando o método PDAP-RP. A seção 5.1 apresenta os estudos preliminares, descrevendo as pesquisas que originaram o método PDAP-RP e suas primeiras aplicações. A seção 5.2 descreve o cenário da aplicação da prática, incluindo quais são as características do grupo de alunos participantes. Na seção 5.3 é apresentada a aplicação do PDAP-RP em práticas usando RP, detalhando todas as etapas envolvidas e os artefatos gerados nesse processo.

5.1. ESTUDOS PRELIMINARES

Nessa seção serão apresentados dois estudos preliminares a essa dissertação, ambos publicados em forma de artigo. O primeiro estudo teve como objetivo verificar o potencial da Análise de Normas (AN) e o uso do *framework* semiótico aliado a práticas usando RP. No segundo estudo, houve o desenvolvimento da metodologia PDAP-RP e sua aplicação prática.

5.1.1 ANÁLISE DE NORMAS E USO DE *FRAMEWORK* SEMIÓTICO

Esse primeiro estudo preliminar foi conduzido com o objetivo de avaliar as potencialidades da proposta de utilizar instrumentos da SO para delimitar e avaliar práticas usando RP. A publicação desse estudo, intitulada como “Uso de Semiótica e Análise de Normas em Práticas de Ensino de Programação de Computadores Utilizando Robótica Pedagógica”, ocorreu na Revista Eletrônica de Tecnologia e Cultura (RETC), 14ª edição, publicada em abril de 2014 e se encontra no Anexo II.

Conforme discutido na Seção 3.1., a RP apresenta diversos benefícios quando utilizada como ferramenta de ensino de programação, mas se faz necessário delimitar dentro de uma prática com robôs, os conceitos que serão trabalhados dentro da resolução do desafio ou problema e determinar meios eficientes de avaliação. Esse estudo demonstrou a composição de uma prática pedagógica utilizando a RP como ferramenta didática e como uma análise semiótica pode trazer meios de avaliação de maneira formal e sistemática, através do uso de Análise de Normas e do *framework* semiótico.

Foi feita uma análise sobre uma das práticas realizadas nas provas da competição R.A.F.A.E.L.A, discutida nessa dissertação na Seção 1.3. Essa atividade foi realizada

usando um robô LEGO® NXT em uma arena, conforme mostra a Figura 16. A descrição dessa prova era: “O robô deverá iniciar a prova no ponto inicial B e percorrer a linha preta até a faixa azul que se encontra no ponto J. A partir do ponto J, o robô deve escolher e seguir dentre as faixas que se originam do ponto J, a faixa que leva menos tempo para chegar na faixa vermelha no ponto E. Ao alcançar a faixa vermelha no ponto E, o robô deverá retornar até o ponto inicial B”.

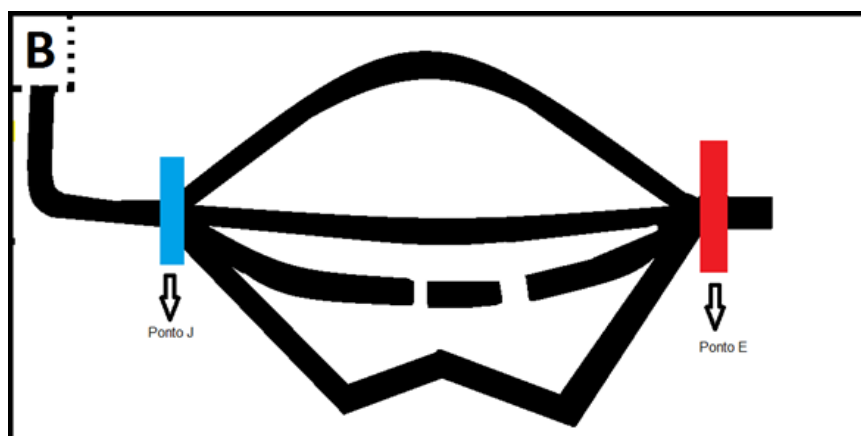


Figura 16 - Exemplo de prática em arena

Foi utilizado o *framework* semiótico como guia para elicitación e análise de problemas e soluções das práticas usando RP. Com isso, é possível organizar os aspectos relacionados ao domínio dos objetivos didáticos e organizar do nível físico ao nível social na escada semiótica. Para a especificação das normas, com as relações dos agentes e seus *affordances*, faz-se necessário uma especificação formal, na qual apresente o domínio envolvido, as responsabilidades e os comportamentos de todos os elementos envolvidos.

Nesse trabalho foi projetado um quadro específico para cada norma e organizado na seguinte estrutura:

- **Agentes:** determina os agentes envolvidos naquele cenário descrito pela norma;
- **Affordances:** aponta quais *affordances* requeridos para o cumprimento da norma;
- **Detalhamento da norma:** descrição através de uma linguagem natural, definindo regras e ações.

Esse quadro é o mostrado na Tabela 1, descrito na Seção 4.3.1.1 (Elaboração do Quadro de Normas Comportamentais), sendo adotado na modelagem final do método PDAP-RP. Esse primeiro modelo de quadro já contava com a descrição completa onde os *affordances* e os agentes de cenário são definidos por uma norma. No item “Detalhamento da norma” do quadro foi utilizada e definida a forma Lógica Deôntica.

Ao final desse trabalho, foi possível identificar que a partir do uso do quadro e das descrições de normas, é possível criar um questionário com questões que representem os pontos a serem avaliados na prática e as relações dos alunos junto a essas normas. Esse primeiro critério de avaliação contava com questões relacionadas à compreensão da solução e ao uso do robô. Os recursos de avaliação evoluíram no trabalho posterior, e foram plenamente introduzidos no modelo final do método PDAP-RP, sendo baseados nas normas perceptuais, cognitivas, avaliativas e denotativas, após a geração dos quadros de normas comportamentais.

5.1.2 CRIAÇÃO E APLICAÇÃO EXPERIMENTAL DO PDAP-RP

Este estudo preliminar teve como objetivo criar o modelo do processo PDAP-RP, com todas as suas fases e etapas sendo definidas e detalhadas, assim como colocá-lo em prática para uma avaliação prévia de sua viabilidade e contribuição ao processo didático e avaliativo. O modelo PDAP-RP e seus resultados foram publicados no XXV Simpósio Brasileiro de Informática da Educação (SBIE 2014), em novembro de 2014, no artigo intitulado “Uma Metodologia Baseada em Semiótica para Elaboração e Análise de Práticas de Ensino de Programação com Robótica Pedagógica”, presente no Anexo III.

O desenvolvimento do método PDAP foi seccionado em quatro fases distintas, cada uma cobrindo um conjunto de etapas, que aplicadas em sequência permitisse, de maneira sistemática, alinhar todos os processos existentes na elaboração e avaliação de uma prática usando RP. Os instrumentos provenientes SO foram organizados em cada fase com o intuito de cobrir etapas da criação de maneira encadeada e coerente para todo o processo. Em um primeiro momento, as fases foram organizadas da seguinte maneira:

- Fase I: Instrumentos, artefatos e atividades que pudessem trazer uma análise inicial do cenário e uma delimitação do problema. Essa fase foi contemplada com instrumentos do método PAM.

- Fase II: Identificar as unidades semânticas, classificar os principais elementos, assim como suas interdependências. As etapas dessa fase estão relacionadas ao método SAM.
- Fase III: Detalhamento dos comportamentos dos elementos anteriormente definidos, relacionar estes através de normas e criar meios de avaliar o desempenho dos alunos. As atividades estão diretamente ligadas ao método NAM.
- Fase IV: Criação formal e aplicação de prática, assim a avaliação e obtenção dos resultados. Nessa fase ocorre uma rotina similar ao usual em um ambiente de ensino, com o professor aplicando a prática perante um conjunto de alunos e, posteriormente, a aplicação de meios de avaliação.

O trabalho ainda contou com o método em uma prática usado com um grupo de seis alunos do curso Técnico em Informática integrado ao Ensino Médio, da Escola Técnica Rosa Perrone Scavone, localizada na cidade de Itatiba-SP, com idades entre quinze e dezessete anos. Todos estavam no 2º ano do curso e já frequentaram disciplinas básicas de Lógica de Programação e Programação de Computadores, e possuíam contato prévio com robótica.

Com o resultado desse experimento, foi possível identificar alguns problemas e adequar algumas etapas no PDAP-RP e aplicá-las ao modelo final do método. As adequações e inclusões ao modelo final ocorreram para formatar o método PDAP-RP de maneira definitiva para os experimentos executados para essa dissertação. O método PDAP-RP está plenamente descrito nas seções anteriores.

5.2. DESCRIÇÃO DOS PARTICIPANTES E CONTEXTO DA APLICAÇÃO

O grupo selecionado para participar da aplicação das práticas são alunos que cursam o Ensino Médio integrado ao Técnico em Informática, na escola técnica Rosa Perrone Scavone, localizada na Rua João dos Santos Rangel, número 66, no bairro Vila Belém, no município de Itatiba – SP. A escola técnica pertence à autarquia estadual Centro Paula Souza e oferece, além do curso Ensino Médio integrado ao Técnico em Informática, o curso de Ensino Médio regular e cursos de nível técnico, nas áreas de Informação e Comunicação, Gestão e Indústria.

O curso Ensino Médio integrado ao Técnico em Informática possui uma grade mista, com as disciplinas pertencentes ao currículo do Ensino Médio, que tem o caráter anual, e o conjunto de disciplinas que integralizam o curso Técnico em Informática. As ementas e cronogramas disciplinas técnicas foram adaptadas para que fossem aplicadas anualmente, ao invés de semestralmente, como é comum em um curso técnico pertencente ao Centro Paula Souza.

Para participar do estudo, quarenta alunos foram divididos em dois grupos distintos. Os discentes tinham idade mínima de 14 anos, e máxima de 17 anos, e cursavam 1º, 2º e 3º anos do curso Técnico em Informática. O critério para a composição dos grupos foi a experiência em Desenvolvimento de Algoritmos de cada aluno.

Os alunos pertencentes ao 1º ano do curso foram classificados como os menos experientes, por estarem no início do curso, portanto não possuem um ciclo completo de aprendizagem de disciplinas como Lógica de Programação ou Programação de Computadores. Esse grupo (Grupo 1) foi subdividido, havendo um grupo de controle (Grupo de Controle), no qual não foi utilizado o método.

O segundo grupo (Grupo 2) é formado por alunos pertencentes ao 2º e 3º ano do curso Ensino Médio integrado ao Técnico em Informática e foram classificados como “experientes”, por terem ao menos uma formação básica, e, conseqüentemente, experiência em conhecimentos relacionados ao Desenvolvimento de Algoritmos. O Grupo de Controle e o Grupo 1 foram compostos por oito alunos cada, escolhidos de maneira aleatória. O Grupo 2 foi formado por vinte e quatro alunos.

As práticas foram aplicadas no formato de oficina, no período de quatro semanas, duas vezes por semana, no período entre agosto e setembro de 2014. Nessa oficina, foram apresentados os principais elementos físicos e recursos do robô, assim como os principais elementos da programação visual do ambiente LEGO®. Essa oficina teve como objetivo ensinar aos alunos como aplicar os conceitos de Desenvolvimento de Algoritmos utilizando a nova sintaxe (programação em blocos), fazendo com que a presença de uma nova “linguagem” não fosse algo que atrapalhasse ou desmotivasse a execução da prática. No primeiro encontro de cada semana era realizada uma oficina para apresentar os elementos e conceitos fundamentais do robô, e no segundo, eram realizadas as práticas. As imagens dos alunos durante as oficinas são apresentadas no Anexo IV.

Nas práticas foram utilizados uma arena e dois robôs LEGO®, ambos do modelo EV3. Nas três primeiras semanas foram aplicadas práticas utilizando o método PDAP-RP e na última semana foi aplicada a prática sem o uso do método para gerar dados comparativos. Todas as práticas ocorreram em um laboratório da própria escola, após o final do período de aulas.

Entre a última semana de setembro e a primeira semana de outubro de 2014 foi realizada a aplicação dos questionários para todos os grupos de alunos participantes. Os questionários são apresentados no Anexo V. O primeiro questionário é relacionado aos quadros de respostas da etapa “Elaboração de Quadros de Respostas (demais normas)” do PDAP-RP. O segundo questionário corresponde à análise da eficiência do método PDAP-RP como ferramenta de auxílio às práticas usando RP. Esse questionário é direcionado para a avaliação do aluno perante o método e a qualidade das práticas desenvolvidas e é usada a escala Likert para abranger respostas negativas e positivas.

5.3. ETAPAS DE APLICAÇÃO DO PDAP-RP

A seguir serão apresentadas e descritas as fases de aplicação do método PDAP-RP, junto ao Grupo 1 e ao Grupo 2 para o estudo de caso.

5.3.1 IDENTIFICAÇÃO DO CENÁRIO E PROBLEMAS NO ESTUDO DE CASO

Essa fase inicia-se com uma breve descrição e delimitação do problema, através dos quatro artefatos definidos na etapa “Delimitação do Cenário/Problema” do PDAP-RP. Foi gerado um documento com toda a descrição, que pode ser apresentada conforme os pontos abaixo:

- Definição dos envolvidos:
 - Professor-pesquisador: responsável pela organização e aplicação dos métodos, assim como a análise dos resultados ao final da prática.
 - Professor: atua como agente facilitador e responsável pela aplicação da prática. Acompanha os alunos durante o processo e aplica as avaliações.
 - Alunos: responsáveis pelo desenvolvimento e apresentação da solução, programação do robô e demonstração em arena.

- Elementos do cenário:
 - Arena, com formato retangular, de piso branco. Trajeto demarcado com fita preta e faixas adesivas coloridas.
 - Obstáculos físicos, no formato paralelepípedo.
 - Robô, em uma montagem característica de um robô móvel (formato veicular).
 - Dois motores, responsáveis pela movimentação do robô pela arena.
 - Computador pessoal com o ambiente de programação LEGO® devidamente instalado e funcional.
- Conceitos envolvidos:
 - Estruturas condicionais: definição dos blocos de programação referentes às estruturas condicionais. Esse conceito é necessário para o reconhecimento de obstáculos e faixas na arena.
 - Estruturas de repetição: definição dos blocos de programação referentes às estruturas de repetição. O uso destas estruturas é necessário para seguir trajetos na arena e retornar, no caso do robô ter que sair do trajeto, por exemplo, quando desviar de um obstáculo no trajeto.
 - Modularização: definir em quais etapas é possível criar soluções modulares, utilizando a programação do ambiente de programação LEGO®. Durante a tarefa, é necessário repetir soluções utilizadas em trechos anteriores, portanto, é possível modularizar ações do robô para uso posterior.
- Objetivos gerais:
 - Identificar o problema;
 - Propor soluções através do uso do robô;
 - Identificar os conceitos de Desenvolvimento de Algoritmos necessários para a solução;
 - Utilizar os conceitos de Desenvolvimento de Algoritmos de maneira coerente.

A definição dos envolvidos foi apresentada nessa seção de maneira breve e objetiva, para assim deixar claro apenas a proximidade desses papéis com as práticas

realizadas. Um detalhamento mais refinado sobre as partes envolvidas (definidos posteriormente como *stackholders*) é feita na próxima etapa. Os elementos do cenário apresenta o que será necessário para a aplicação da prática, principalmente em relação aos recursos físicos. No momento da apresentação dos conceitos envolvidos é importante definir as estruturas necessárias para a resolução da tarefa e os blocos de programação para acompanhar o desempenho dos alunos e orientá-los (papel do professor aplicador). Os objetivos gerais foram listados para que houvesse uma definição inicial do que se era esperado por parte dos alunos, servindo como um guia da avaliação ao final da prática.

A próxima etapa, a de “Análise de *Stackholders*”, contou com uma descrição mais detalhada, utilizando a “cebola semiótica” como instrumento de análise do grau de envolvimento dos *stackholders* nos diferentes níveis (técnico, formal e informal). Para essa análise, foi criada a Tabela 3, organizada com todas as características nos diferentes níveis e atribuídos os *stackholders* relacionados, em especial, professor e aluno. Nesse momento não houve a diferenciação entre agentes, papéis ou *affordances*, dando prioridade apenas a distinção de responsabilidades e partes interessadas. Os aspectos do domínio foram listados de maneira abrangente, entretanto, trazem uma composição consistente do cenário para uma análise inicial.

Tabela 3 – descrição dos níveis da cebola semiótica

Nível	Aspecto do domínio	Responsável
INFORMAL	• Ensinar conceitos de Desenvolvimento de Algoritmos. • Utilizar recursos da RP de maneira efetiva. • Avaliar o desempenho dos alunos.	Professor
	• Aplicar conceitos de Desenvolvimento de Algoritmos no desenvolvimento de soluções. • Ser capaz de programar o robô.	Aluno
FORMAL	• Desenvolvimento da prática. • Apresentação e aplicação da prática. • Definição dos conceitos de Desenvolvimento de Algoritmos envolvidos. • Delimitação dos objetivos.	Professor
	• Compreensão do problema. • Apresentação de soluções. • Aplicação das soluções com o robô.	Aluno
TÉCNICO	• Ensinar a programação em blocos (ambiente LEGO®). • Estruturar uma rotina de avaliação. • Análise dos resultados,	Professor
	• Programação e comunicação com o robô. • Compreensão técnica dos recursos disponíveis no robô	Aluno

Como descrito no método PDAP-RP, após a análise dos *stackholders*, foi utilizado o *framework* semiótico como guia para elicitación e análise de problemas e soluções das práticas usando RP. Através desse instrumento, é possível listar os aspectos relacionados ao domínio dos objetivos didáticos em particular, organizados do nível social ao nível físico da escada semiótica. Por se tratar de uma descrição que detalha objetivos didáticos, foram criadas questões relacionadas a cada um dos níveis, que delimitam o que é esperado no contexto geral da prática. A seguir são apresentadas as questões aplicadas a cada nível do *framework*:

- **Nível social:** Qual a consequência da RP na formação profissional do aluno? Qual o desempenho futuro do aluno na sua atuação profissional?
- **Nível pragmático:** O uso dos robôs apoia corretamente as práticas propostas aos alunos? Como o professor poderá utilizar a RP nessas práticas? O aluno conseguirá distinguir e utilizar todos os recursos das ferramentas (programação e robôs) de acordo com suas intenções? O uso da RP criou um ambiente eficiente de aprendizagem? Os alunos conseguiram abstrair conceitos de Desenvolvimento de Algoritmos e utilizá-los nas tarefas?
- **Nível semântico:** Quais os significados atribuídos através da comunicação e da programação do robô? Há compreensão do que é programado e a ação correspondente a ser executada? A ação do robô corresponde ao esperado?
- **Nível sintático:** Qual é a sintaxe da linguagem que os robôs oferecem? Como ocorrerá essa comunicação entre aluno e robô (programação)? Quais são as tecnologias e técnicas envolvidas (conceitos e aplicação)?
- **Nível empírico:** Qual é o meio de comunicação do aluno com o robô? Qual é a banda de comunicação necessária? Quais são os problemas envolvidos nesse processo?
- **Nível físico:** Quais são os recursos de hardware disponíveis nos robôs? Quais as ações podem ser realizadas utilizando esses recursos? Qual é o hardware necessário para poder trabalhar?

O produto gerado pelo *framework* cria um guia com questões de acompanhamento e avaliação da efetividade do processo. A compilação dessas questões

no diferentes níveis apresenta uma solução para análise de todo o processo de construção e desenvolvimento da prática, até o desempenho dos alunos. A verificação de cada um desses pontos possibilita identificar problemas na fase inicial do desenvolvimento e criação da prática, como por exemplo, a ausência de algum recurso no robô. Também apresenta uma solução para avaliação, principalmente nos níveis semântico e sintático, expressando de forma direta a interação do aluno com o robô e o desenvolvimento de soluções.

Como produto das questões do *framework*, foi criado um quadro, como ilustra a Tabela 4, que apresenta um resumo das ações fundamentais para satisfazer as questões expostas no *framework*. Esse quadro possibilita um detalhamento maior do cenário e dos objetivos a serem alcançados com a prática pedagógica. Essas ações auxiliam na identificação de componentes que influenciam tanto a criação quanto na solução da prática.

Tabela 4 – Quadro criado a partir da análise do framework semiótico

Nível	Ações componentes de cada nível
Social	• Proporcionar ao aluno o contato com novas tecnologias. • Desenvolver a capacidade de resolução de problemas e análise crítica. • Capacitar para discernir soluções e aplicações tecnológicas. • Possibilitar uma visão tecnológica diversificada.
Pragmático	• Apresentar os robôs como ferramenta pedagógica. • Correlacionar conceito de Desenvolvimento de Algoritmos à programação de robôs. • Apresentar e utilizar novas ferramentas (robô e sua linguagem de programação). • Criar tarefas com o robô as quais possam utilizar soluções algorítmicas.
Semântico	• Estruturar uma solução algorítmica com o robô e analisar suas respostas, podendo redefinir a solução através dessa análise. • Proporcionar um nível de abstração, na qual possa compreender conceitos de Desenvolvimento de Algoritmo de maneira isolada. • Promover meios para o aluno ser capaz de analisar os problemas relativos às respostas provenientes dos recursos do robô (identificação de problemas e dificuldades).
Sintático	• Selecionar e expor quais são os recursos de programação do robô, condizentes aos conceitos de Desenvolvimento de Algoritmos. • Definir quais conceitos de Desenvolvimento de Algoritmos serão contemplados na prática. • Aplicar os conceitos de Desenvolvimento de Algoritmos no ambiente de programação do robô.

Empírico	• Analisar se os recursos presentes robô condizem com o que foi programado. • Integrar a programação feita (no ambiente de programação) com o equipamento físico, através do uso de cabo USB.
Físico	• Compreender a função de cada recurso (e.g. sensores e motores) de maneira individual (para identificar sua função na programação) e conjunta (reação do robô e sua consequência com outros recursos). • Identificar, para cada trecho da solução, qual o recurso responsável em fornecer a entrada ao sistema (e.g. sensor de ultrassônico medindo uma distância) e a saída (e.g. o que fazer quando o robô estiver a 15 centímetros do obstáculo).

5.3.2 ANÁLISE SEMÂNTICA NO DOMÍNIO

Essa fase teve como objetivo a continuidade da descrição do domínio, mas com maior refinamento e formalismo, através de uma AS mais completa. Iniciou-se com a definição dos agentes predominantes “professor” e “aluno”, principais responsáveis pelos *affordances*, papéis e dependências ontológicas. A etapa “Definição de Agentes e *Affordances*” e a etapa “Agrupamento de Candidatos e modelagem de Ontologias” (descritas respectivamente nas Seções 4.2.1 e 4.2.2) foram executadas em sequência e tiveram um caráter complementar.

Para a essa primeira fase de AS foi elaborado um enunciado sobre o domínio. A apresentação deste enunciado, mesmo sendo bastante vago no momento inicial, direcionou uma análise com foco no objetivo principal e consequentemente quem são os principais responsáveis. Para a descrição da prática usando RP, o enunciado foi o apresentado a seguir:

“Criar um ambiente motivador, colaborativo, participativo e eficaz para o ensino de conceitos de Desenvolvimento de Algoritmos, utilizando Robótica Pedagógica, através de meio sistemático, com o objetivo de auxiliar o desenvolvimento, a aplicação e a avaliação das práticas executadas pelos alunos.”

O enunciado acima não apresenta grandes detalhes do cenário, mas solicita a presença de entidades que possam solucionar o problema apresentado. Nessa análise nota-se a presença de agentes que assumam papéis de “professor” e “aluno”. O enunciado reforça a presença de um agente que agregue esses papéis, que no contexto ontológico, torna-se um tipo do agente “Pessoa”, confirmando também a Análise de *Stackholders* feita anteriormente. Também foi definida a presença de agente responsável

ao *affordance* robô, definido como “Fabricante”. O refinamento da análise ocorreu na etapa seguinte, devido ao limite do enunciado.

Com todas as informações da etapa anterior, assim como da fase anterior (Fase I), deu-se início o agrupamento de *affordances* candidatos e modelagem de ontologias. Nessa etapa foi feita uma análise mais detalhada das unidades semânticas, definidas anteriormente. Durante o agrupamento de *affordances*, foram surgindo novas unidades semânticas, sendo relevante ao modelo de ontologias final.

A apresentação do diagrama de ontologia ocorreu de forma expositiva, através de uma leitura com os grupos de alunos durante a oficina, explicando o diagrama e suas dependências, que foi acompanhada pela descrição dos elementos ontológicos, como apresentado na Seção 4.2.2.2..

5.3.3 DELIMITAÇÃO DAS AÇÕES E COMPORTAMENTOS

Esta fase de aplicação inicia-se com a definição dos quadros de normas comportamentais, ou seja, a “Elaboração dos Quadros Comportamentais”. Estes quadros descrevem no nível pragmático, os comportamentos esperados dos alunos perante o problema apresentado. Com esse recurso é possível definir as regras de resolução esperadas, utilizando os recursos de programação do robô. Essa delimitação das ações e comportamentos na relação aluno-robô proporciona indicativos de avaliação do desempenho do aluno na execução da prática.

O cumprimento ou não da norma serve de guia para uma atividade avaliativa, já que demonstra a capacidade de aplicação de conceitos direta no uso de RP. Cada quadro remete a uma ação pontual, a qual o robô deve executar (no quadro apresenta como a “Norma”), como por exemplo, “desviar de um obstáculo na arena” ou “seguir linha do trajeto”. No campo “Agentes” são definidos quais são os agentes envolvidos e seus papéis em relação à norma. Na aplicação do método PDAP-RP o agente “Aluno” no papel de “programador” foi o mais recorrente, por representar ação direta no desenvolvimento das soluções. No campo “*Affordances*” foram descritos os elementos que estão presentes na execução da norma. Para uma melhor apresentação e interpretação por parte dos alunos, o termo “*affordance*”, usado no campo do quadro,

pode ser substituído pelo termo “Elementos e ações”, caso tenha a intenção de utilizar o PDAP-RP como guia de aplicação.

A seguir são apresentados os quadros, contemplando todas as normas definidas para a elaboração da prática. Foram definidas quatro normas, que executadas de maneira individual representam um fragmento da solução e englobam um ou mais conceitos de Desenvolvimento de Algoritmos. Estes conceitos são aplicados na programação em bloco, como discutido na Seção 2.1.1. A Figura 17 exemplifica o uso de blocos de programação para que o robô se movimente identificando o traçado no piso e o siga através do sensor de luz, ajustando a posição pelos motores (estrutura de seleção) enquanto não encontra um obstáculo à frente (estrutura de repetição) através do sensor ultrassônico. Caso encontre um objeto o robô para. Maiores detalhes sobre as propriedades dos blocos e a programação em bloco do Lego® EV3 podem ser obtidas em Carnegie (2014).

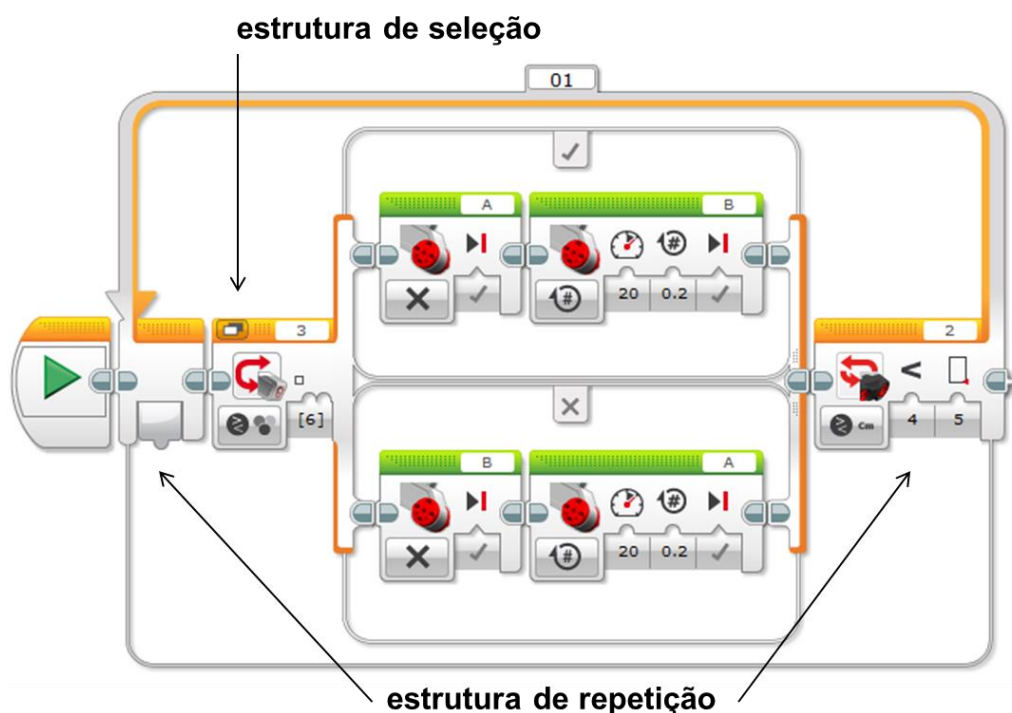


Figura 17 – Exemplo de programação em blocos

A norma “Desviar de obstáculo” na Tabela 5 apresenta a solução necessária para que o robô desvie dos obstáculos presentes na arena. Nela, o aluno precisa aplicar

conceitos relacionados a estruturas de seleção, e utilizar os recursos do sensor ultrassônico, que informa a distância de um objeto em relação a ele mesmo.

Tabela 5 - Quadro da norma “Desviar de obstáculo”

<i>Norma</i>	<i>Desviar de obstáculo</i>
Agentes	Aluno (no papel de programador).
Affordances	Robô (<i>affordance</i> da Sociedade); Programação Robô (<i>affordance</i> aluno com o Robô); Desvia Obstáculo (<i>affordance</i> aluno por meio da programação do robô).
Detalhamento da norma	sempre que robô identificar um obstáculo na arena; se é preciso desviar desse obstáculo; então , o aluno; é obrigado a; utilizar uma estrutura de seleção para determinar o desvio.

A norma “Seguir traçado”, apresentada na Tabela 6, detalha a solução esperada para que o robô siga o traçado dentro da arena. Na execução da prática o robô movimenta-se seguindo um traçado (linha na cor preta) definido no piso, representado o caminho a ser seguido. Era esperado que o aluno utilizasse uma estrutura de repetição, na qual definisse que o robô deveria se mover para frente, enquanto o robô notasse o traçado com o recurso do sensor de luz, que diferencia o nível de luminosidade diferente entre o traçado formado pela linha preta e o piso branco.

Tabela 6 – Quadro da norma “Seguir traçado”

<i>Norma</i>	<i>Seguir traçado</i>
Agentes	Aluno (no papel de programador).
Affordances	Robô (<i>affordance</i> da Sociedade); Programação Robô (<i>affordance</i> aluno com o Robô); Seguir traçado (<i>affordance</i> aluno por meio da programação do robô).
Detalhamento da norma	sempre que robô identificar um traçado no piso; se é preciso seguir esse traçado; então , o aluno; é obrigado a; utilizar uma estrutura de repetição para se siga enquanto houver traçado.

Na Tabela 7 é apresentada a norma “Encontrar traçado”, que descreve a solução necessária para quando sensor de luz não estiver sobre o traçado, devido a algum desvio de obstáculo ou manobra para esquerda ou direita. Muitas vezes, pelo formato do robô, uma manobra para esquerda ou direita faz com o que sensor de luz, que é fixo ao “corpo” do robô, não esteja mais sobre o traçado, nem durante a manobra ou ao final dela. Quando isso ocorrer, é necessário utilizar uma estrutura de repetição, a qual faça o robô de deslocar para esquerda ou direita, até que seja encontrado novamente o traçado.

Tabela 7 – Quadro da norma “Encontrar traçado”

<i>Norma</i>	<i>Encontrar traçado</i>
Agentes	Aluno (no papel de programador).
Affordances	Robô (<i>affordance</i> da Sociedade); Programação Robô (<i>affordance</i> aluno com o Robô); Encontrar traçado (<i>affordance</i> aluno por meio da programação do robô).
Detalhamento da norma	sempre que o robô não identificar mais o traçado; se é preciso retornar ao traçado; então , o aluno; é obrigado a; utilizar uma estrutura de repetição para girar o robô (independente de ser esquerda ou direita) até encontrar o traçado.

A Tabela 8 apresenta a norma “Identificar faixa”, utilizando o sensor de luz para distinguir uma faixa prata localizada no piso da arena. Sempre que encontrar é necessário que o robô pare, pois na prática desenvolvida, essa faixa se torna uma marcação de “fim de traçado”. Para desenvolver essa solução, era esperado que o aluno utilizasse uma estrutura de seleção analisando o valor informado pelo sensor de luz quando estivesse sobre a faixa.

Tabela 8 – Quadro da norma “Identificar faixa”

<i>Norma</i>	<i>Identificar faixa</i>
Agentes	Aluno (no papel de programador).
Affordances	Robô (<i>affordance</i> da Sociedade); Programação Robô (<i>affordance</i> aluno com o Robô); Identificar faixa (<i>affordance</i> aluno por meio da programação do robô).
Detalhamento da norma	sempre que encontrar uma faixa; se for preciso parar ; então , o aluno; é obrigado a; utilizar uma estrutura de seleção para definir se encontrou a faixa e parar o robô.

Tabela 9 detalha a norma “Identificar falha no traçado”. Essa norma diferencia-se da “Encontrar traçado” pelo seu contexto de aplicação. A programação do robô é feita de maneira encadeada, ou seja, cada parte da solução total correspondente a um trecho da arena, tendo essas soluções sequenciadas pelo aluno. Portanto, quando o discente cria a solução para atravessar uma falha no traçado, ele identifica que há uma reta e após ela uma falha, retomando o traçado após atravessar essa falha. Detalhes da arena são apresentados na Seção 5.2.2.

Tabela 9 – Quadro da norma “Identificar falha no traçado”

<i>Norma</i>	<i>Identificar falha no traçado</i>
<i>Agentes</i>	Aluno (no papel de programador).
<i>Affordances</i>	Robô (<i>affordance</i> da Sociedade); Programação Robô (<i>affordance</i> aluno com o Robô); Identificar faixa (<i>affordance</i> aluno por meio da programação do robô).
<i>Detalhamento da norma</i>	sempre que encontrar uma falha em um segmento de traçado; se for preciso continuar em frente ; então , o aluno; é obrigado a; utilizar uma estrutura de seleção identificar uma falha e utilizar uma estrutura de repetição para mover o robô para frente, até encontrar novamente o traçado.

Após a modelagem dos quadros de normas comportamentais, foram definidas as questões de caráter avaliativo relacionado às demais normas (perceptuais, cognitivas, denotativas e avaliativas) para a etapa “Elaboração dos Quadros de Respostas (Demais Normas)”. Para cada um dos aspectos foram criadas questões que pudessem gerar respostas dissertativas que expressassem as impressões dos alunos em relação a eles aspectos. As questões foram apresentadas e as normas relacionadas:

1. Houve dificuldades na compreensão da tarefa? (Relacionada aos aspectos cognitivos).
2. O objetivo da tarefa foi facilmente percebido? (Relacionada aos aspectos perceptuais).
3. Teve alguma dificuldade em perceber os efeitos dos conceitos de programação? (Relacionada aos aspectos perceptuais).
4. Conseguiu compreender plenamente a aplicação dos conceitos de programação no robô? (Relacionada aos aspectos cognitivos).
5. Houve dificuldades em entender as respostas (*feedback*) do robô para as ações programadas? (Relacionada aos aspectos avaliativos).
6. Os resultados obtidos foram satisfatórios para seu aprendizado? (Relacionada aos aspectos avaliativos).
7. A prática ajudou na aquisição de novos conceitos de programação ou ajudou a concretizar ainda mais seu conhecimento? (Relacionada aos aspectos denotativos).
8. Os conceitos de programação foram escolhidos adequadamente de acordo com a categoria/tipo de tarefa? (Relacionada aos aspectos denotativos).

Na elaboração das questões foi utilizado o termo “programação” e “conceitos de programação” ao invés de Desenvolvimento de Algoritmos, pois são termos mais comuns aos alunos e eram mais utilizados no momento das execuções das práticas. Para que não houvesse dupla interpretação ou deficiências no momento de responder o questionário, optou-se por utilizar os termos mais comuns no diálogo com e entre os alunos. Os quadros de respostas são apresentados no Anexo VI e as sínteses das respostas no Capítulo 6.

5.3.4 APLICAÇÃO DA PRÁTICA E AVALIAÇÃO

As atividades executadas nessa fase estão relacionadas à composição e à especificação da proposta, como a criação de um enunciado e descrição dos objetivos da prática pedagógica, e definição dos elementos físicos e recursos necessários. Essas especificações estão relacionadas tanto à montagem da arena, quanto à estrutura do robô. Para a criação da tarefa, utilizou-se o procedimento “Definições de Práticas usando RP” do método PDAP-RP.

A composição da arena foi dividida nas áreas A e B, sendo que a prática inicia-se com a execução na área A e depois execução da B. A divisão da arena em duas áreas teve como objetivo possibilitar que o aluno que desenvolva as rotinas de solução na área A, possa utilizar as soluções já prontas na área B. Com isso, é possível avaliar a capacidade do aluno em analisar o cenário e verificar se consegue aplicar componentes de soluções prontas, princípio esse relacionado à modularização, ou seja, identificar soluções prontas (módulos) e agregá-los em uma solução nova ou maior. Outro benefício da divisão da arena foi a possibilidade de dois alunos executarem suas tarefas ao mesmo tempo, sendo que um utiliza a área A, enquanto o outro, que já utilizou a área A, desenvolve a tarefa na área B. A Figura 18 mostra o desenho conceitual da arena e suas áreas.

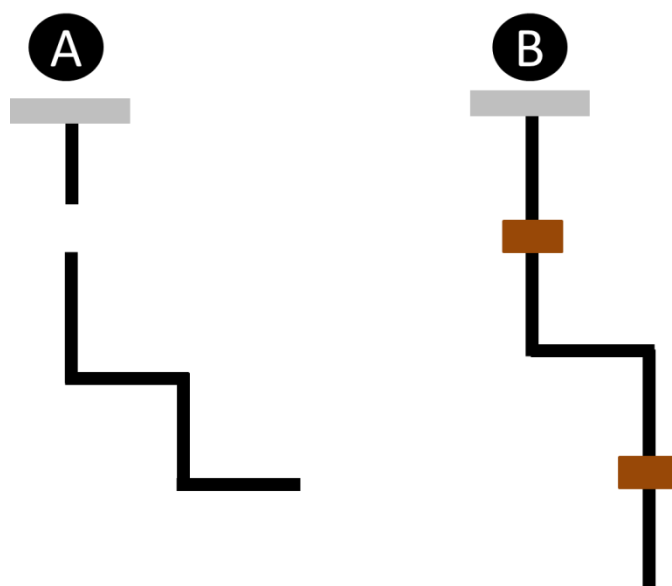


Figura 18 – Desenho conceitual da arena

Os elementos para a montagem da arena seguiram o padrão similar ao apresentado em provas da OBR¹¹, não contemplando todos os elementos existentes, mas utilizando as especificações de dimensões em relação ao piso da arena (placa de madeira MDF branca), traçado (marcado com fita isolante na cor preta), obstáculo (blocos de madeira em formato de paralelepípedo) e faixas pratas (fitas de sinalização prateadas). A Figura 19 mostra uma visão da arena construída, sendo à esquerda a área A e à direita a área B.

¹¹ O material contendo as regras utilizadas nas provas da OBR em 2014 pode ser acessado em http://www.obr.org.br/wp-content/uploads/2013/04/regras_pratica_regionais_v2_Mini2014.pdf.

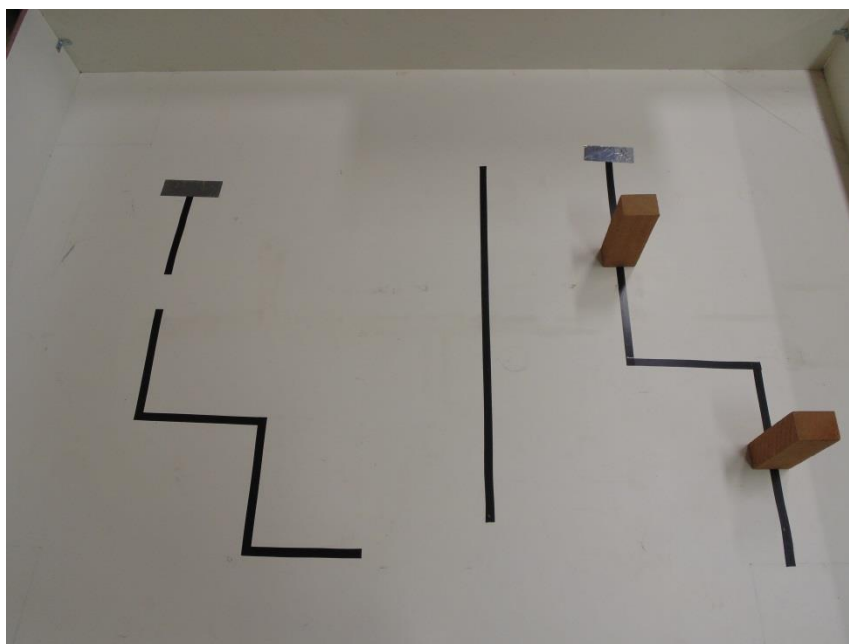


Figura 19 – Visão geral da arena

A Figura 20 detalha os elementos físicos que compõe a arena. Faixas prateadas (número 1) demarcam o final do traçado e o fim da respetiva área. Na área A há uma falha no traçado, ou *gap*, termo usado em competições oficiais da OBR (número 2), na qual o robô deve identificar e seguir em frente. Os obstáculos foram colocados apenas na área B (numero 3), alinhados ao traçado (número 4), feito com um linha preta contínua, presente em ambas as áreas. Os pontos de início do traçado (número 5) serão usados para colocar o robô em sua posição inicial para a execução da prática.

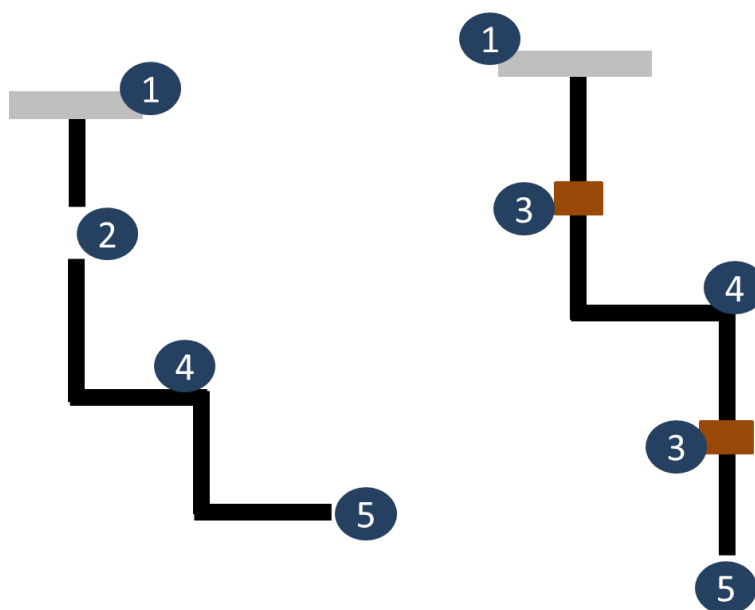


Figura 20 – Detalhamento da arena

O robô foi configurado com uma estrutura similar aos que executam as provas da OBR, ou seja, em um formato de um robô móvel, sendo disponibilizado já estruturalmente pronto. A definição dos elementos físicos, estruturação e montagem do robô não faziam parte da prática a ser executada pelos alunos, apenas a programação do mesmo foi relevante à pesquisa.

Sua composição estrutural era similar a de um veículo com quatro rodas, distribuídos em dois eixos, sendo o eixo traseiro responsável pela tração do robô, pois era onde se localizavam os motores. Os sensores foram colocados na frente do robô. O sensor ultrassônico, responsável pelo reconhecimento de obstáculo e o sensor de luz para identificação do traçado (e sua ausência) e da faixa prateada. Para a tarefa adotou-se apenas robôs do mesmo modelo, para que não houvesse diferenças de estrutura física e de particularidades referentes à programação. A Figura 21 mostra detalhes da estrutura física do robô.

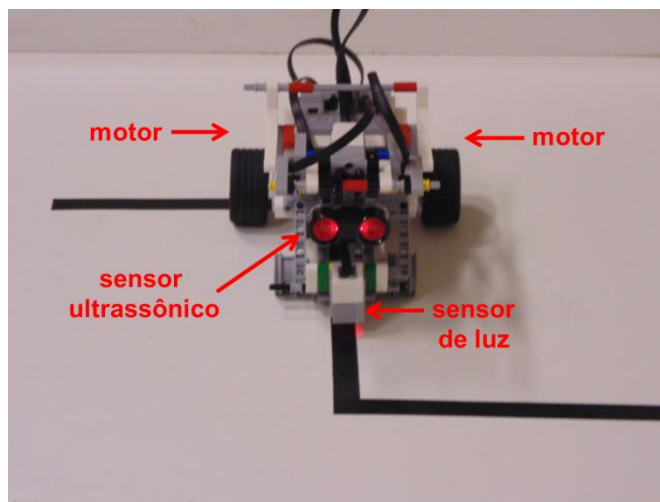


Figura 21 – Detalhes da estrutura física do robô

Após definidos os elementos físicos, foram descritos o enunciado da prática, tais como as regras para solução e os objetivos a serem alcançados, como discutido no processo proposto. As descrições são apresentadas a seguir:

- **Enunciado da prática:** para a área A da arena, o enunciado proposto foi “O robô deve ser colocado em frente ao início do traçado, e deverá seguir esse traçado até encontrar o final, demarcado com uma faixa prata. Durante o trajeto haverá uma falha no traçado que deve ser atravessado e o robô deve encontrar o traçado novamente”. O início do traçado é apresentado na Figura 20, indicado pelo número 5 e o final do traçado é indicado pelo número 1. Para a área B da arena o enunciado proposto foi “O robô deverá ser colocado em frente ao início do traçado, e deverá seguir esse traçado até encontrar o seu final, demarcado por uma faixa prata. Durante o trajeto deverão ser evitados os obstáculos e o robôs deverão retornar ao traçado após feito cada desvio”. Assim como na área A, o início e fim são identificados com os números 5 e 1, respectivamente na Figura 20, e os obstáculos com o número 3.
- **Regras para solução:** As descrições dessas regras refletem diretamente o comportamento esperado por parte dos alunos, portanto foram apresentadas, além da descrição detalhada de regras e limitações do robô, as normas comportamentais a serem seguidas durante a realização da oficina feita antes da execução da prática. Na apresentação das normas, foi explicado o item “Detalhamento da Norma”, havendo uma explicação

do seu significado, devido ao formato apresentado pela Lógica Deôntica. Após as normas, foi apresentado um resumo da estrutura física do robô e quais sensores estavam disponíveis. Também foram apresentadas quais eram as regras de execução na arena, por exemplo, o robô deve sempre andar sobre o traçado, ou que o mesmo deve retornar ao traçado sempre que houver algum desvio.

- **Objetivos a serem alcançados:** Os objetivos foram apresentados para as diferentes áreas da arena em dois contextos: um geral, definindo o objetivo geral de cada área, e vários específicos, diferente para cada trecho (*e.g.*, desviar de obstáculo, identificar faixa, etc.).

Como descrito na etapa “Avaliação de Resultados” do PDAP-RP, foi criado um quadro de avaliação para acompanhamento da execução e desempenho perante as normas comportamentais. O quadro apresenta uma avaliação com os termos “não cumpriu”, “cumpriu parcialmente” e “cumpriu plenamente”. Essa avaliação foi feita durante execução das práticas pelos alunos. O quadro é apresentado no Anexo VII e os resultados discutidos no Capítulo 6.

Os quadros de respostas, referente à etapa “Elaboração dos Quadros de Respostas (Demais Normas)”, foi aplicado através de um questionário, disponibilizado *online* - apresentado no Anexo V - juntamente com um questionário para avaliação quantitativa, utilizando o padrão de respostas na escala Likert, apresentado no Anexo V. O resultado e a síntese dos questionários são apresentados no Capítulo 6.

6. ANÁLISE DOS RESULTADOS E DISCUSSÕES

Esse capítulo apresenta a análise dos resultados obtidos tanto através dos questionários (Anexo V) quanto na avaliação feita pelo professor. As discussões sobre os resultados dos questionários serão divididas pelo caráter qualitativo e quantitativo. Na Seção 6.1 temos a análise das respostas referentes à etapa “Elaboração dos Quadros de Respostas (Demais Normas)”, através de discussões sobre as respostas dissertativas dos grupos participantes do experimento. A análise dos resultados quantitativos será apresentada na Seção 6.2, através de análise comparativa de gráficos e discussões. Na Seção 6.3 discute os resultados da avaliação e do acompanhamento do professor sobre as normas comportamentais. A Seção 6.4 apresenta uma análise de trabalhos relacionados.

O Grupo de Controle e o Grupo 1 possuem integrantes de mesmo perfil (alunos iniciantes) e foram escolhidos aleatoriamente, compondo dois grupos de oito alunos. O Grupo 2 é formado por vinte e quatro alunos com nível de experiência maior do que os outros grupos do experimento. O experimento com o Grupo de Controle e Grupo 1 tem como objetivo comparar os resultados dessas duas populações, para a verificação de indícios da eficácia do método PDAP-RP. Esta verificação foi realizada comparando um grupo que não teve a aplicação do método (Grupo de Controle) com o grupo que teve (Grupo 1). O experimento com o Grupo 2 objetiva uma análise da presença do método em um grupo com maior experiência, mais diversificado e com um maior número de participantes, podendo ser verificados pontos fortes e fracos do PDAP-RP.

É importante salientar que apenas o Grupo 1 e Grupo 2 efetuaram as práticas com o PDAP-RP, enquanto o Grupo de Controle apenas participou das oficinas e resolveram as práticas normalmente, isto é sem a apresentação das normas comportamentais (quadros). Com os resultados da experimentação pode ser feita uma comparação direta entre o Grupo de Controle e o Grupo 1, por serem grupos de mesmo perfil distribuídos aleatoriamente e é possível notar uma relação de efetividade da presença do método. As comparações entre o Grupo 1 e Grupo 2 se apoiam na diferença dos perfis (iniciantes versus experientes) e nas implicações a adoção do método em um grupo de alunos não-iniciantes.

6.1. ANÁLISE QUALITATIVA

A análise qualitativa baseia-se nas respostas obtidas no Questionário 1 (Anexo V), que organiza as questões relacionadas às normas perceptuais, cognitivas, avaliativas e denotativas. Esse questionário foi aplicado aos três grupos, incluindo o Grupo de Controle, para que fosse feita uma análise comparativa relacionada às normas para uma verificação da influência do método PDAP-RP. O quadro de respostas do Grupo de Controle, Grupo 1 e Grupo 2 são apresentados no Anexo VI.

Esta análise baseia-se principalmente nas respostas negativas, pois apontam dificuldades e problemas encontrados pelos alunos. Se analisarmos o Grupo de Controle através dessas respostas de caráter negativo é possível perceber de maneira mais clara em quais aspectos (perceptuais, cognitivos, avaliativos ou denotativos) houve maior dificuldade e apontamentos do que pode ser melhorado em uma prática utilizando RP. As respostas negativas dos grupos 1 e 2 apresentam foram problemas presentes na aplicação da prática com o método PDAP-RP, assim como pontos a serem melhorados e possíveis correções para o modelo.

Foram selecionadas as respostas mais significativas dos alunos, pois o número elevado tornaria inviável a análise individualizada, além de que muitas delas seriam repetitivas. As respostas selecionadas e contextualizadas durante a análise de resultados, sendo categorizadas por questão ou por tipo de norma envolvida, diferenciando também cada um dos três grupos participantes do experimento.

Em relação às normas perceptuais (questões 2 e 3), o Grupo de Controle apresentou alguns problemas em relação a identificação dos objetivos da prática e os efeitos dos conceitos de Desenvolvimento de Algoritmos na programação do robô. Algumas respostas da questão 2, sobre a dificuldade na compreensão dos objetivos, e da questão 3, sobre a percepção dos efeitos dos conceitos de programação, demonstram isso. Entre as repostas, destaca-se aqui uma que aponta a falta de detalhamento sobre os objetivos, mostrando a necessidade de melhorar a sua apresentação.

“Não, foi difícil de perceber, pois tem coisas muito variadas para fazer e não foi explicado como tinha que ser feito.”

Dentre as respostas da questão 3 foi possível notar a dificuldade em perceber a relação entre os conceitos e os efeitos na ação do robô, mostrando a ausência de

elementos que pudessem amparar o aluno no momento da execução da prática, conforme destaca as duas respostas abaixo:

“Um pouco. Pelo fato dos conceitos de programação serem mais complicados. Isso na minha opinião.”

“Tive, pois tem muitos comandos e eu não consegui entender o que cada um fazia.”

Em relação às normas cognitivas, quando questionados sobre a compreensão da tarefa (i.e., o que devia fazer durante a prática) na questão 1, foi possível identificar problemas relacionados à descrição da tarefa (ausência de detalhes) e como atuar com os blocos de programação diretamente na solução. Algumas respostas mais significativas que salientam esse fato são:

“No começo sim, pois não consegui entender direito como fazer e o que deveria ser feito, mas depois eu consegui entender direito.”

“Inicialmente houve dificuldades para o entendimento da função que cada bloco de comando exercia sobre o robô. Porém, após ver o funcionamento do robô, o conteúdo foi fixado mais facilmente.”

Ainda relacionado às normas cognitivas, foi possível identificar dificuldades quando questionados sobre a aplicação dos conceitos de programação (questão 4). É notável a necessidade de um aprofundamento na correlação dos conceitos de Desenvolvimento de Algoritmos, com a aplicação prática na programação do robô, como mostram as respostas a seguir:

“Mais ou menos, às vezes surgia uma pouco de dificuldades.”

“Em relação aos conceitos aprendidos durante o treinamento, houve certa facilidade em sua compreensão. Porém, fica claro como poderíamos ter explorado de forma mais profunda alguns outros conceitos que foram apresentados apenas de forma superficial.”

Na questão 5, relacionada às normas avaliativas, o Grupo de Controle demonstrou alguns problemas. A questão trata das dificuldades de identificar o *feedback* do robô em resposta ao que era programado, ou seja, se o mesmo reagia como a programação determinava. Houve momentos de incertezas e a necessidade de uma vivência com a plataforma para que esse *feedback* fosse compreendido de maneira mais adequada. Entre as principais respostas que apontam o problema estão:

“Um pouco, em algumas vezes o robô não atendia ao programa feito.”

“Em algumas coisas...”

“Sim, no começo, mas depois de um tempo deu para compreender e perceber que o erro estava na programação.”

Em relação à questão 6, também atrelada às normas avaliativas, o Grupo de Controle não apresentou respostas negativas. Esta questão trata da satisfação do aprendizado através da prática e não houve, portanto, rejeições significativas.

As questões 7 e 8, relacionadas às normas denotativas, não tiveram respostas negativas do Grupo de Controle. Às respostas à questão 7 foram todas positivas e afirmavam que a prática ajudou na concretização ou adquirir novos conhecimentos em conceitos de Desenvolvimento de Algoritmos. As respostas em relação à questão 8 também afirmam que eles identificaram corretamente quais conceitos aplicar na resolução da tarefa.

Estendendo a análise para o Grupo 1, as questões 1 e 4 (normas cognitivas) não tiveram respostas negativas, apontando uma melhoria quanto aplicado o método PDAP-RP. Enquanto alguns alunos do Grupo de Controle mostravam dificuldades em compreender o que deveriam realizar durante a tarefa e também dificuldades em compreender a aplicação dos conceitos de Desenvolvimento de Algoritmos, os alunos do Grupo 1 não demonstraram tais dificuldades. A aplicação do método proporciona artefatos que auxiliaram na descrição da prática (geral e tarefas individuais) e ofereceram guias para aplicação dos conceitos na programação dos robôs.

As questões relacionadas às normas perceptuais, especificamente a questão 2, não apresentaram nenhuma resposta negativa por parte dos alunos. Através das respostas, deixa aparente que o objetivo da tarefa estava claro na perspectiva dos alunos, demonstrando uma melhora se comparado ao Grupo de Controle. Na questão 3 as respostas apontam uma melhora com relação a percepção dos alunos na aplicação dos conceitos e efeitos nos robôs. Apenas uma resposta possui um caráter negativo, mas denota uma dificuldade anterior que o aluno já tinha em relação aos conceitos de Desenvolvimento de Algoritmos, conforme demonstrado abaixo:

“Um pouco, encontrei mais dificuldade a esse respeito pois ainda tinha duvidas nos conceitos da programação.”

Nas questões 5 e 6, correspondentes às normas avaliativas, de maneira geral obtiveram respostas positivas, não demonstrando os mesmo problemas encontrados no Grupo de Controle. Já a questão 6 não apresentou nenhuma resposta negativa no Grupo 1 e nem no Grupo de Controle, portanto não é possível tecer uma comparação entre os grupos. Em relação à questão 5, a maioria das respostas foram positivas, mas ainda assim apresentou alguns problemas similares aos do Grupo de Controle. As respostas que apontam isso são:

“No inicio sim, mas depois ficou mais fácil de fazer.”

“Um pouco, pois tinha algumas vezes, que por pequeninas falhas no programa, o robô estava executando outra coisa.”

A exemplo das respostas do Grupo de Controle, as questões 7 e 8 (normas denotativas) não foram respondidas negativamente pelo Grupo 1.

A análise das respostas do Grupo 2 apresenta de maneira geral respostas positivas em todas as questões, com exceção de algumas críticas e apontamentos negativos, principalmente à problemas físicos do robô. Toda prática usando RP deve pressupor a ocorrência de incoerências em ações do robô, já que ele sofre a interferência de elementos externos à programação (e.g., nível baixo de bateria ao acionar motores, variação de luminosidade na arena influenciando o sensor de luz). Muitas das respostas negativas partem desses problemas, os quais o método PDAP-RP não contempla. Também deve-se considerar a experiência e, conseqüentemente, a capacidade crítica desses alunos.

Em relação às questões 1 e 4, houve pouca reprovação por parte do grupo. Na questão 1 foram apenas três respostas negativas, sobre a dificuldade na compreensão da tarefa. Esta questão estava relacionada a clareza no que era proposto e aos critérios de avaliação conforme destacado abaixo:

“Em alguns casos, pois não houve uma explicação satisfatória.”

“Um pouco, pois em alguns casos nem todos os pontos de avaliação eram ditos.”

“Algumas propostas não tiveram explicações coerentes.”

Na questão 4, relacionada aos aspectos cognitivos, houve apenas uma resposta totalmente negativa, que apresenta uma dificuldade do aluno em aplicar os conceitos de

programação no robô, ou em relacionar os conceitos de desenvolvimento de algoritmos na PV do ambiente de programação, conforme demonstrado abaixo:

“Não, alguns deles eram um pouco confusos.”

A questão 2 não obteve nenhuma resposta negativa, mostrando uma eficiência na descrição do objetivo da tarefa. Na questão 3 houve uma resposta negativa, apontando problemas no kit robótico e outra destacando uma dificuldade por parte do aluno na utilização dos blocos da PV. Mas a maioria afirma ter conseguido perceber sem dificuldade os efeitos na programação. As respostas negativas foram:

“Não, alguns deles eram um pouco confusos.”

“Sim, a única dificuldade apresentada se deve ao modo de como se programava utilizando os blocos; sendo que a utilização de códigos seria para mim mais entendível.”

Os maiores índices de respostas negativas esteve presente na questão 5, onde os alunos criticam os problemas e inconsistências que o kit robótico apresenta. Dentre as vinte e quatro respostas, oito eram negativas. Isso apresenta um grande desafio na aplicação da RP, pois pode influenciar diretamente na motivação e participação no aluno no processo de ensino-aprendizagem. Alguns exemplos significativos dessa classe de respostas são:

“Sim, robôs não respondiam de acordo com o programado.”

“Sim, algumas vezes ele não operava corretamente”

“Mais ou menos, muitas vezes o robô falhava ou não executava algum comando inserido.”

“As vezes ele agia de maneira um tanto como aleatórias”

“um pouco, por mais que tentávamos entender o que o robô ia fazer às vezes ele dava uma resposta diferente do que queríamos.”

A questão 6 apresentou apenas duas respostas negativas, sendo que as demais corroboram que o resultado da prática colaborou com o processo de aprendizagem. Uma das respostas negativas visivelmente é influenciada pelos problemas físicos apresentados pelo robô, como pode ser observado abaixo:

“Mais ou menos, o robô apresentou dificuldades nas respostas, dificultando o aprendizado.”

A outra resposta negativa apresentou a insatisfação com a plataforma robótica utilizada, mas não em específico com o uso de RP, como podemos observar abaixo:

“Não, gostaria de ter utilizados Kits que pudessem melhorar realmente meus conceitos tanto de logica de programação como a programação em si, os Kits utilizados não ofereceu os resultados que eu esperava.”

As questões 7 e 8 tiveram apenas respostas positivas nesse grupo.

A Tabela 10 apresenta uma síntese das respostas dos grupos analisados, apresentando a distribuição das respostas negativas e positivas de cada grupo. É possível notar a incidência maior das positivas no Grupo 1 em relação ao Grupo de Controle, no que tange a maioria das questões. Também é notável a presença de alto índice de respostas positivas nas questões respondidas pelo Grupo 2 (exceto a questão 5, como discutido na análise anteriormente).

Tabela 10 – Tabela síntese das respostas do questionário

Questão	Grupo de Controle		Grupo 1		Grupo 2	
	Positivas	Negativas	Positivas	Negativas	Positivas	Negativas
1	6	2	8	0	21	3
2	7	1	8	0	24	0
3	5	3	7	1	22	2
4	6	2	8	0	22	2
5	5	3	6	2	14	10
6	7	1	7	1	22	2
7	8	0	8	0	24	0
8	8	0	8	0	24	0

Algumas das respostas apresentadas pelos alunos muitas vezes não possuem um detalhamento suficiente que possibilite a identificação do problema ou das dificuldades de uma maneira precisa. Entretanto, mesmo com a baixa granularidade nas respostas, é possível notar um cenário predominantemente positivo nos grupos que utilizaram o PDAP-RP, mostrando a potencialidade de se ter um processo formal para o desenvolvimento de práticas usando RP.

Essa análise qualitativa ainda pode ser revista e melhorada, através de métodos que possam torná-la mais precisa na avaliação em relação aos aspectos semânticos, cognitivos e comportamentais dos alunos na presença do método, refinando os resultados apresentados nessa seção.

6.2. ANÁLISE QUANTITATIVA

A análise quantitativa foi feita com base nos dados extraídos no Questionário 2 (Anexo V), no qual foi utilizando a escala Likert para compor as opções de respostas de múltipla escolha pelos alunos. Esse questionário possibilita mapear aspectos relacionados ao uso RP e a aceitação dessa ferramenta no ensino de Desenvolvimento de Algoritmos, nos diferentes grupos de alunos analisados.

Como explicado anteriormente, nas questões foram utilizados os termos “programação” e “conceitos de programação” ao invés de Desenvolvimento de Algoritmos, para não haver uma interpretação errônea por parte dos alunos. As questões estão relacionadas abaixo, na ordem que foram apresentadas no questionário:

1. A Robótica Pedagógica é um instrumento interessante para ensino de programação?
2. A comunicação com o robô ocorreu sem problemas ou dificuldades?
3. Os recursos (sensores, motores, etc.) do robô foram percebidos de maneira clara?
4. As respostas dadas pelo robô à programação executada foram claras e coerentes?
5. A utilização da Robótica Pedagógica auxiliou na aprendizagem ou aumentou seu conhecimento nos conceitos de programação?
6. A forma de avaliação da prática era bastante clara e coerente?
7. Não há problemas/dificuldades em aplicar os conceitos de Lógica de Programação na Programação Visual por blocos?
8. Não houve problema em compreender os objetivos da prática proposta?
9. Não houve dificuldade em elaborar a solução da prática proposta?
10. Não houve dificuldades em aplicar os conceitos de Lógica de Programação (estruturas de decisão, estruturas de repetição e modularização) na solução?

A Figura 22 mostra a porcentagem de respostas do Grupo de Controle (alunos inexperientes na aplicação do método). É possível notar que mesmo sem a aplicação do método PDAP-RP a utilização da RP é relevante no processo de ensino-aprendizagem, quando percebemos que houve maioria de respostas positivas (39% de “concordo plenamente” e 44% de “concordo parcialmente”). Não há quase reprovação no uso da RP, quando analisamos o percentual de respostas exclusivamente negativas (1% de “discordo parcialmente” e nenhuma resposta sendo “discordo totalmente”).

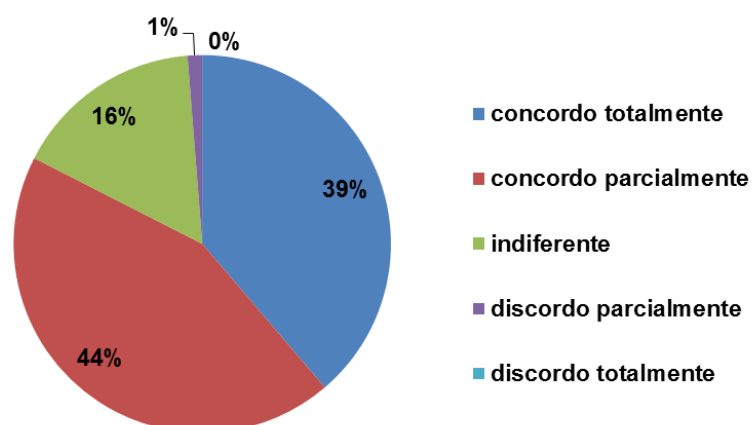


Figura 22 – Gráfico com a porcentagem das respostas do Grupo de Controle

A Figura 23 mostra a porcentagem das respostas do Grupo 1 (grupo de alunos inexperientes na utilização do método PDAP-RP). Nele, é possível perceber que, a exemplo do grupo anteriormente analisado, a aceitação da RP foi positiva, com 51% das respostas sendo “concordo plenamente” e 41% sendo “concordo parcialmente”. Também nota-se um baixo índice de reprovação do uso de RP quando analisamos exclusivamente as respostas negativas, sendo apenas 3% de respostas “discordo parcialmente” e nenhuma resposta “discordo totalmente”.

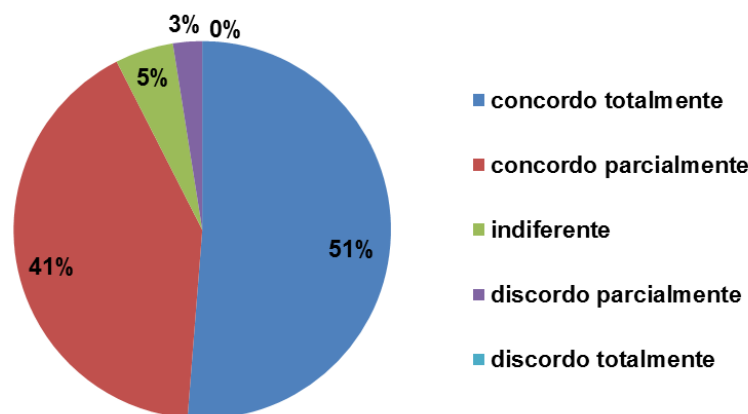


Figura 23 - Gráfico com a porcentagem das respostas do Grupo 1

Comparando os dois grupos com perfis similares é possível notar uma sensível melhoria quando aplicado o método PDAP-RP. As correlações relevantes nessa análise são:

- Porcentagem de respostas positivas: No Grupo 1 houve uma predominância da resposta “concordo totalmente” como 51% e no total das respostas mais positivas, com 92% contra 83% do Grupo de Controle, demonstrando uma melhoria na aceitação do uso de RP.
- Porcentagem de respostas “indiferente”: Outro fator relevante é a diminuição da porcentagem de respostas “indiferente” no Grupo 1. Essa diminuição representa diretamente o aumento nas respostas positivas, o que permite verificar uma sensível melhoria na percepção dos benefícios da prática e do uso da RP na presença do PDAP-RP.

A mesma avaliação foi realizada com o Grupo 2 (alunos experientes que utilizaram o método PDAP-RP. Conforme a Figura 24 houve uma aceitação predominantemente positiva na utilização da RP, como 43% de respostas “concordo totalmente” e 36% das respostas sendo “concordo parcialmente”, com um total de 79%. Houve baixa reprovação, com 7% de respostas “discordo parcialmente” e 2% de “discordo totalmente”.

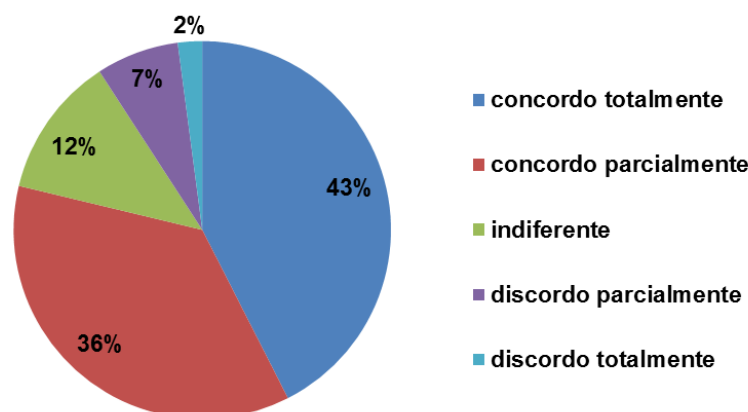


Figura 24 - Gráfico com a porcentagem das respostas do Grupo 2

Em uma comparação com o Grupo de Controle, as porcentagens de respostas positivas foram bem próximas, assim como a porcentagem de respostas “indiferente”. Quando comparamos com o Grupo 1 há uma diferença sensível, tanto nas respostas positivas (diferença de 13%) quantas nas negativas (diferença de 7%). Essas diferenças nos resultados eram esperadas devido à diferença do perfil em relação ao nível de experiência em Desenvolvimento de Algoritmos.

Os alunos pertencentes ao Grupo 2 são consequentemente mais críticos em relação aos instrumentos e métodos de aprendizagem oferecidos. Também é relevante a experiência que possuem em ambientes de programação avançados, onde já se sentem confortáveis, sendo que a utilização de um ambiente menos abstrato pode não apresentar vantagens. Entretanto, análise dos resultados em um contexto geral demonstra que mesmo com alunos experientes, a utilização da RP pode ser praticada e, consequentemente, o método PDAP-RP é viável, ainda que com menos efetividade em relação ao grupo inexperiente (Grupo 1).

A diferença entre os grupos participantes do experimento também é perceptível em uma análise individual sobre as questões. A Figura 25 mostra o número de ocorrências das respostas por questão do Grupo de Controle.

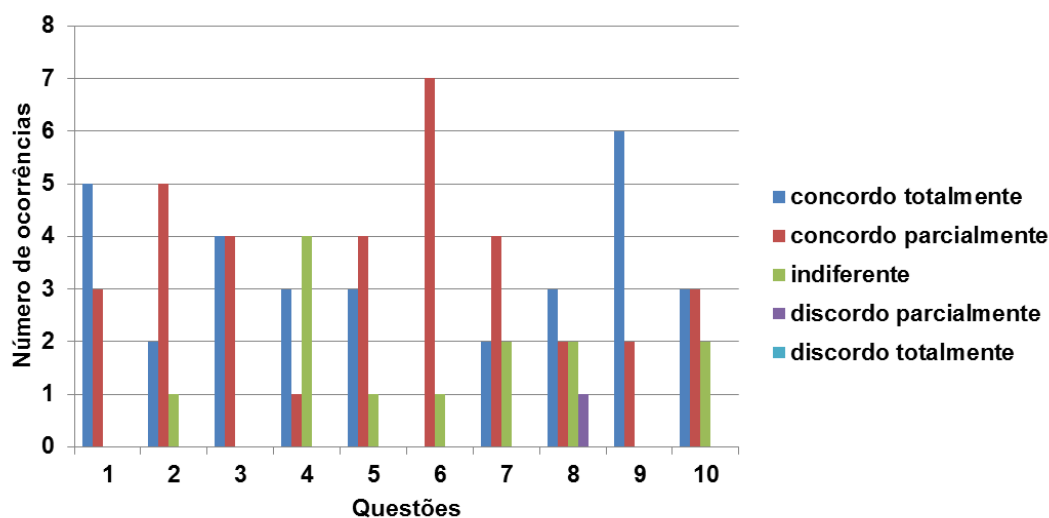


Figura 25 – Gráfico do número de ocorrências das respostas por questão do Grupo de Controle

Alguns aspectos podem ser notados e discutidos a partir da análise do gráfico exibido na Figura 25. A predominância de respostas positivas discutida anteriormente é facilmente notada ao longo das questões. Entretanto, também é perceptível em quais questões houve uma maior reprovação, com a maior ocorrência de respostas “indiferente” e onde ocorreram as respostas negativas.

A Figura 26 abaixo apresenta o gráfico dos números de recorrências das respostas por questão do Grupo 1.

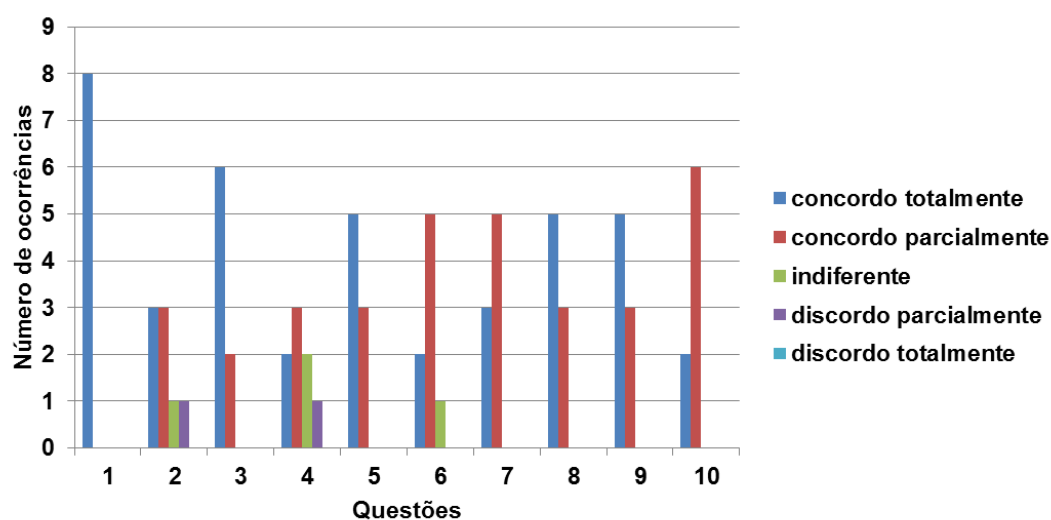


Figura 26 - Gráfico do número de ocorrências das respostas por questão do Grupo 1

A partir da análise dos gráficos apresentados nas Figuras 25 e 26 alguns aspectos podem devem ser ressaltados:

- Questões 2 e 4: Essas questões correspondiam à clareza nas respostas dadas pelo robô aos comandos executados através da programação e à aspectos técnicos. Na questão 2 houve uma diminuição das respostas positivas entre os dois grupos. Na questão 4 houve uma ocorrência de respostas “indiferente” maior do que as respostas positivas no Grupo de Controle. No Grupo 1 também ocorreram respostas “indiferente” e “negativa” na questão 4. Estas questões estão diretamente relacionadas a aspectos tecnológicos particulares ao robô, como, por exemplo, comportamentos inesperados oriundos de mau funcionamento de sensores, dificuldades na comunicação física ou presença de carga baixa na bateria. A ausência de uma melhoria significativa do método PDAP-RP pode ser justificada devido à falta de uma descrição mais técnica em relação à plataforma robótica, por não ter a intenção de diferenciar a aplicação em diferentes plataformas.
- Questão 7: Esta questão está relacionada com a utilização da PV como interface de programação do robô. Entretanto, o Grupo 1 teve a disposição recursos como descrições e detalhamento das ações esperadas pelo robô que poderiam orientá-los melhor em relação aos blocos de programação, justificando uma sensível melhora se comparado ao grupo de controle.
- Questões 8, 9 e 10: Estas questões estão diretamente relacionadas à compreensão do objetivo da prática e desenvolvimento da solução. Uma possível explicação é que a melhoria percebida com o Grupo 1 em comparação ao Grupo de Controle pode ser atribuída ao fato do PDAP-RP fornecer artefatos que expõem o objetivo de maneira clara, através de uma descrição inicial detalhada e o uso de normas comportamentais para orientar a execução da solução.

Na Figura 27 temos a distribuição das respostas por questão dadas pelos alunos do Grupo 2.

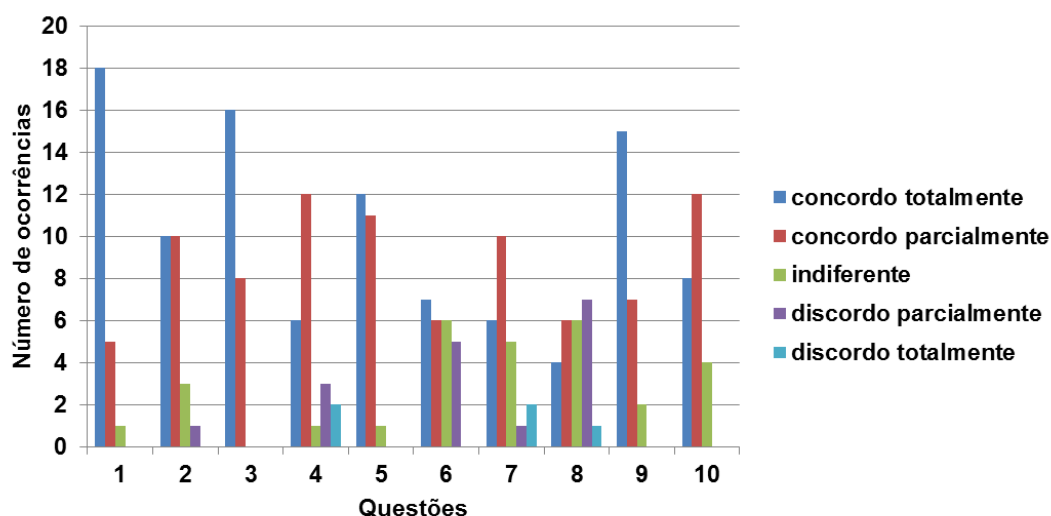


Figura 27 - Gráfico do número de ocorrências das respostas por questão do Grupo 2

É perceptível uma concentração de respostas negativas nas questões 6, 7 e 8. A questão 6 é relacionada ao método de avaliação, que por não ter um caráter exclusivamente técnico, pode ter sido mal compreendida por alguns alunos do grupo.

A questão 7 apresentou um número considerável de ocorrência de respostas negativas e está relacionada ao uso de blocos de programação. Como dito anteriormente, a experiência com programação através de linhas de código pode influenciar no uso blocos de programação, pois se difere na maneira a qual eles dominam.

A questão 8, por sua vez, apresentou o maior número de respostas negativas entre as questões e está relacionada à compreensão dos objetivos da prática. A proposta do PDAP-RP em relação à descrição dos objetivos da prática pode não ser plenamente adequada a grupos de alunos que já possuem vivência com exercícios de Desenvolvimento de Algoritmos mais avançados e usando recursos mais sofisticados, com foco no desenvolvimento de software, tarefa mais complexa, que exige total domínio dos conceitos básicos.

6.3. AVALIAÇÃO SOBRE AS NORMAS COMPORTAMENTAIS

Como proposto no método PDAP-RP houve um acompanhamento da execução das tarefas e avaliação do cumprimento ou não das normas comportamentais que delimitam ações fundamentais a serem executadas pelo robô. A tabela com a coleta de dados encontra-se no Anexo VI e foi aplicada ao Grupo 1, Grupo 2 e também ao Grupo de Controle, mesmo este não tendo sido aplicado o método. A aplicação dessa avaliação

no Grupo de Controle teve o objetivo de criar dados que possam ser comparados com os demais grupos, para uma análise exploratória e verificação se há diferenças significativas entre os desempenhos nos diferentes grupos.

A Figura 28 mostra o desempenho geral do Grupo 1 em relação a execução das normas comportamentais.

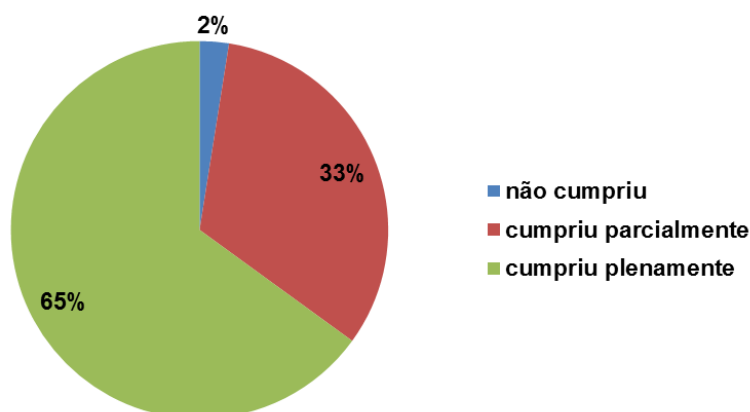


Figura 28 – Desempenho geral no cumprimento das normas comportamentais do Grupo 1

O gráfico demonstra que o desempenho durante a tarefa foi satisfatório, pois há uma predominância de cumprimento pleno em 65%. A porcentagem de cumprimento parcial encontra-se em 33%, demonstrando que mesmo com alguma dificuldade, os alunos, na maioria das vezes, encontravam a solução esperada.

O não cumprimento integral da norma ocorreu apenas 2% das tentativas. Esse cenário denota uma eficiência na apresentação das relações das normas comportamentais como instrumento de orientação aos alunos para o desenvolvimento de soluções.

Essa avaliação baseada em normas possibilita ao professor identificar em quais ações os alunos apresentam as maiores dificuldades, podendo avaliar e estruturar possíveis alterações e correções, assim como fazer uma intervenção maior nessas, concentrando esforços e trabalhando nos conceitos de Desenvolvimento de Algoritmos. Essa avaliação também fornece recursos para definir se problemas na execução da ação em questão estão diretamente relacionadas à programação do robô (solução) ou aos aspectos físicos e técnicos.

A Figura 29 mostra a distribuição das ocorrências dos níveis de cumprimento de cada norma. Com isso, é possível identificar que as normas “Seguir traçado”, “Encontrar

traçado” e “Identificar falha no traçado” apresentaram um maior nível de dificuldade na execução.

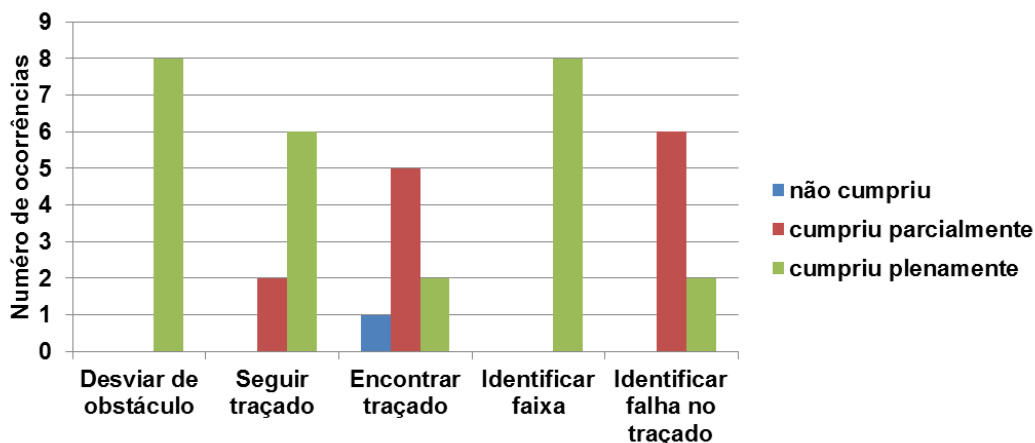


Figura 29 - Número de ocorrências dos níveis de cumprimentos das normas do Grupo 1

O mesmo acompanhamento foi feito com o Grupo 2, conforme a Figura 30, que destaca o desempenho geral do grupo no cumprimento das normas comportamentais. A exemplo do Grupo 1, é possível perceber que o desempenho durante a tarefa foi positivo, com a predominância de cumprimento pleno em 58% das tentativas. A porcentagem de cumprimento parcial ficou em 36% e o não cumprimento integral da norma ocorreu em 6% das tentativas.

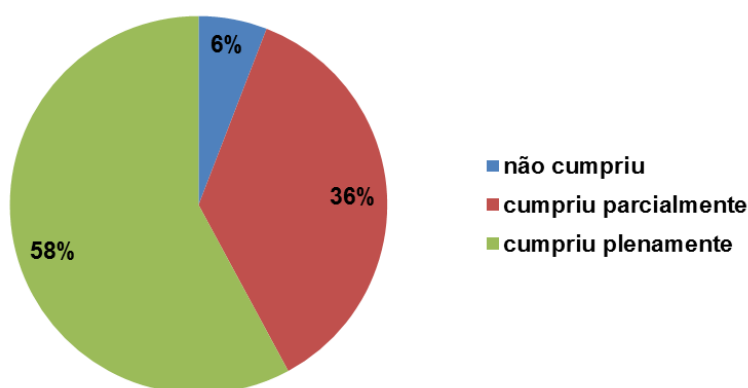


Figura 30 - Desempenho geral no cumprimento das normas comportamentais do Grupo 2

O gráfico aponta um desempenho próximo ao do Grupo 1, com uma maior incidência de normas não cumpridas ou cumpridas parcialmente. É importante salientar

que as variâncias dos grupos foram: 0,0085 para o Grupo de Controle, 0,0278 para o Grupo 1 e 0,0623 para o Grupo 2. A Tabela 11 apresenta os números absolutos dos grupos na execução das cinco normas especificadas. Com isso é possível notar que o Grupo 2 é mais disperso, podendo justificar uma maior incidência de normas não cumpridas e cumpridas parcialmente. Mesmo assim, é possível observar que mais da metade do Grupo cumpriu plenamente a normas.

Tabela 11 – Incidências dos cumprimentos das normas nos grupos

Grupos	Incidência			Total
	Não cumpriu	Cumpriu Parcialmente	Cumpriu Plenamente	
Controle	3	21	16	40
Grupo 1	1	13	26	40
Grupo 2	7	44	69	120

A Figura 31 apresenta a distribuição por normas, executadas pelo Grupo 2. É possível notar que as normas “Seguir traçado”, “Encontrar traçado” e “Identificar falha no traçado” foram as que apresentaram maior dificuldade, visto que nelas, os alunos tiveram o pior desempenho, igualmente ao Grupo 1.

A identificação das dificuldades sobre essas normas de maneira igual entre os dois grupos é um indicativo interessante sobre o qual o professor pode se apoiar para criar melhorias e revisar os aspectos que as envolvem.

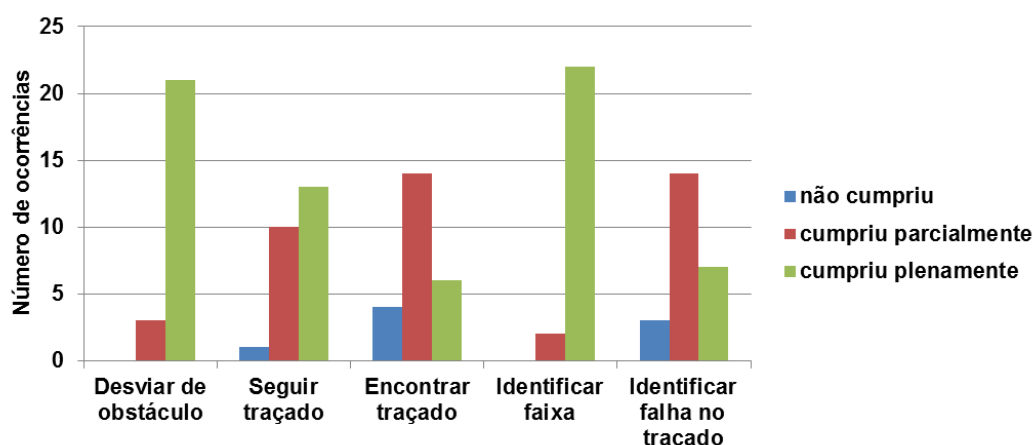


Figura 31 - Número de ocorrências dos níveis de cumprimentos das normas do Grupo 2

A Figura 32 apresenta o desempenho do Grupo de Controle. No gráfico é possível notar que houve uma predominância do cumprimento parcial das normas (53%), sendo superior aos demais grupos (33% do Grupo 1 e 36% do Grupo 2). O índice

do não cumprimento das normas foi o maior entre os grupos (7%), superando do Grupo 1 por 5% e o Grupo 2 por apenas 1%. O índice de cumprimento pleno foi o mais baixo entre os grupos, sendo de 40%.

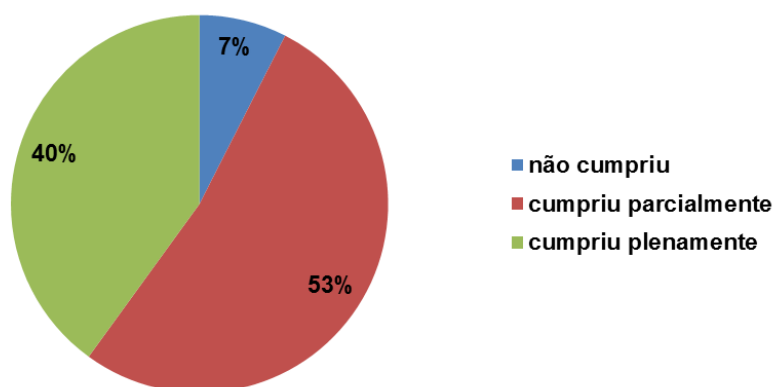


Figura 32 - Desempenho geral no cumprimento das normas comportamentais do Grupo de Controle

Na Figura 33 temos a distribuição por normas do Grupo de Controle. É possível notar um desempenho geral inferior aos demais grupos. O grupo apresentou dificuldades na execução de todas as normas, com índice de cumprimento parcial sendo superior nas normas “Encontrar traçado” e “Identificar falha no traçado”, e igualando os índices de cumprimento pleno nas demais normas.

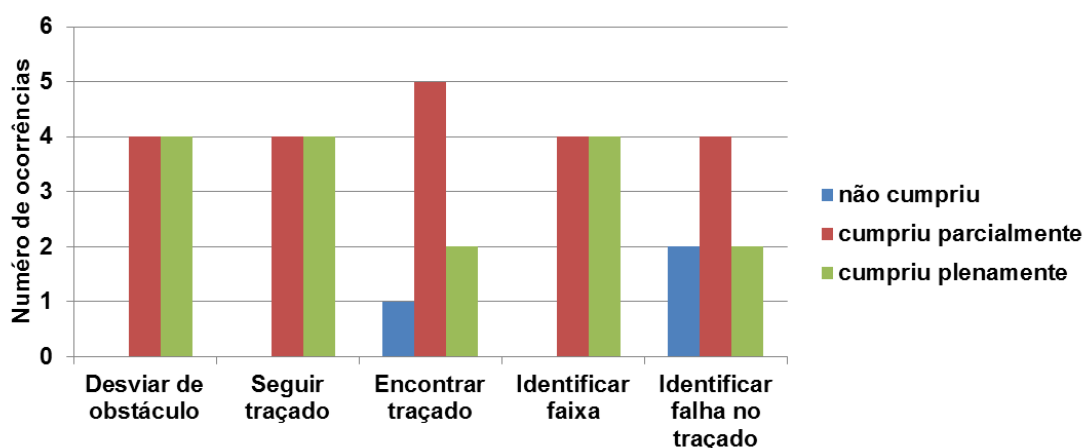


Figura 33 - Número de ocorrências dos níveis de cumprimentos das normas do Grupo de Controle

Para a população estudada, é notável, a partir de uma análise pelos gráficos, que o Grupo 1 e Grupo 2 possuíram um desempenho superior ao Grupo de Controle,

comprovando a interferência positiva nos resultados pela presença do método PDAP-RP. Com o intuito de comprovar a relação determinística entre as médias dos grupos e validar a influência do método PDAP-RP nos resultados, foi utilizado o Teste t de Student, através da proposição de uma hipótese nula, para comprovar a diferença significativa entre as amostras. Para essa análise foi utilizada a seguinte hipótese nula:

- H_0 : Não existe diferença no desempenho dos alunos com o uso do método PDAP-RP ($H_0: \mu_1 = \mu_2$)

Também foi utilizada uma hipótese alternativa, sendo:

- H_A : Existe uma diferença no desempenho dos alunos com o uso do método PDAP-RP ($H_A: \mu_1 \neq \mu_2$)

O nível de significância adotado foi de 5% ($\alpha = 0,05$) e a análise foi feita entre o Grupo de Controle e o Grupo 1, pois possuem os perfis de indivíduos com maior proximidade, tornando a análise mais fidedigna. A Tabela 12 mostra alguns resultados da análise com os grupos, como média e desvio padrão.

Tabela 12 – Análise dos grupos

	Grupo de Controle	Grupo 1
Indivíduos	8	8
Média	2,35	2,62
Desvio padrão	0,0926	0,1669

Tendo em vista a diferença obtida entre as médias com valores que variavam de 1 a 3 (sendo 1 para “não cumpriu”, 2 para “cumpriu parcialmente” e 3 para “cumpriu plenamente”), faz-se necessário aplicar o Teste t para determinar se a diferença entre essas médias é significativa ou não. A Tabela 13 mostra os resultados do Teste t utilizado para medir a significância da diferença entre essas médias.

Tabela 13 – Resultados do Teste t

(t)	Grau de liberdade (gl)	(p) bi-caudal
-4,0753	14	0,0011

Nessa análise o Teste t sugere que existe uma diferença significativa entre as médias de desempenho dos grupos. Com o *valor-p* bi-caudal (0,0011) sendo menor que

o nível de significância α (0,05), rejeitando a hipótese nula (H_0), isto é, as médias dos dois grupos são significativamente diferentes, reforçando a melhoria na presença do método PDAP-RP.

6.4. DISCUSSÃO E TRABALHOS RELACIONADOS

O método PDAP-RP abrange o desenvolvimento e avaliação de práticas, através de um meio sistemático e formal, utilizando artefatos e bases teóricas que envolvem SO e RP. Esta abordagem possui um caráter inédito, tendo em vista pesquisas em trabalhos relacionados à Semiótica aplicada em ensino e aprendizagem, RP para aprendizagem em Desenvolvimento de Algoritmos ou abordagens mistas dessas duas frentes de pesquisa. Consultas em bases indexadoras de pesquisas acadêmicas como SciELO¹², IEEE Xplore¹³, Scopus¹⁴, ACM¹⁵ e Web of Science¹⁶ não retornam resultados nas consultas de trabalhos que relacionem a SO em um método formal exclusivo no desenvolvimento e avaliação de práticas usando RP.

Em uma análise minuciosa de pesquisas relacionadas, duas apresentam abordagens próximas ao proposto nessa dissertação, entretanto não possuem similaridade com o método PDAP-RP, mas utilizam metodologias com propósitos e abordagens com objetivos comuns. Dois trabalhos relevantes para discussão são apresentados nesta seção em ordem cronológica de publicação: (1) o trabalho de Catlin e Blamires (2010), no qual foi desenvolvido um *framework* para a orientação do desenvolvimento e avaliação de práticas com RP e; (2) o trabalho de Gaudiello, Zibetti e Pinaud (2012) que explora as maneiras de controlar robôs educacionais, através de programação e as respostas que o dispositivo pode fornecer.

Em Catlin e Blemires (2010), a discussão principal possui enfoque na efetividade do uso de dispositivos robóticos e o envolvimento dos alunos e professores. Para a pesquisa, foi utilizado um conjunto de princípios de aplicação de RP, analisando aspectos pedagógicos e comportamentais. Os princípios pertencentes a esse *framework* eram:

¹² Scientific Electronic Library Online. Disponível em: <http://www.scielo.org/>

¹³ IEEE Xplore Digital Library. Disponível em: <http://ieeexplore.ieee.org/>

¹⁴ Scopus Digital Library. Disponível em: <http://www.scopus.com/>

¹⁵ Teh ACM Digital Library: Disponível em: <http://librarians.acm.org/>

¹⁶ Web of Science. Disponível em: <http://wokinfo.com/>

1. Explicações
 - a. Como os robôs ajudam os alunos a aprender.
 - b. Os benefícios da RP para professores.
2. Oferecer um “*check list*” para aqueles que querem:
 - a. Projetar robôs educacionais.
 - b. Desenvolver atividades com RP.
3. Ajuda para justificar o investimento feito por escolas em tecnologia robótica.
4. Sugestões referentes aos processos cognitivos e de desenvolvimento.
5. Fornecer aos pesquisadores um conjunto de alegações para avaliar.

A utilização da Semiótica está presente no trabalho que explora os aspectos na perspectiva cultural e de interação e comunicação entre alunos e robô, em específico, com o sistema LOGO. Essa plataforma é tratada como um sistema semiótico no contexto do trabalho, mas o foco principal fica na interação e comunicação entre os elementos aluno e robô, denotando um estudo voltado para Interface Humano-Computador (IHC) e Interface Humano-Robô (IHR), quando relacionado à aplicação da Semiótica.

Entretanto, o trabalho de Catlin e Blamires (2010) não aborda todos os níveis do *framework* semiótico, no qual se baseia o método PDAP-RP. Também não explora recursos formais e organizados no desenvolvimento das tarefas, como os fornecidos pelos métodos PAM, SAM e NAM da SO. Consequentemente, esta pesquisa avança no estado da arte no tema, ao apresentar recursos avaliativos com o nível de descrição oriundo da análise de normas. Catlin e Blamires (2010) têm como foco comprovar e qualificar quais são as habilidades que podem ser trabalhadas em conjunto (aluno, professor e robô), sem se preocupar com uma avaliação detalhada. Isto decorre do fato dos autores não direcionarem a utilização desse *framework* especificamente para o ensino de Desenvolvimento de Algoritmos, mas para uma aplicação generalista em currículos escolares.

Em Gaudiello, Zibetti e Pinaud (2012), é apresentado um estudo cujo objetivo é explorar as maneiras de controlar robôs educacionais, através de programação, e as respostas que o dispositivo pode fornecer. Para isso são definidos dois aspectos fundamentais relacionados ao controle de robôs: transparência (“programabilidade”) e interatividade (eminência de *feedback*), com um foco primário na interatividade, através

das heurísticas de programação, e como objetivo secundário, a transparência. Como transparência, o trabalho compreende quais são as ações (programar), os meios (programação) necessários para o controle dos robôs, e as dificuldades envolvidas nesse processo. A interatividade é estudada essencialmente através das respostas que o robô fornece, geralmente imediatas e concretas, e quais os benefícios são obtidos, tanto na visão do aluno quanto na do professor.

O estudo utilizou quarenta crianças entre 6 e 10 anos para práticas usando os kits LEGO® Mindstorm NXT, para o estudo direcionado à três heurística distintas: (i) *procedural-oriented heuristics*, (ii) *conceptual-oriented* e, (iii) *metacognitive-oriented*. Cada uma dessas heurísticas aborda um aspecto do uso e da criação do conhecimento, e também como as crianças atuavam no controle dos robôs. Pode-se distinguir cada uma das heurísticas de programação através de sua abordagem abaixo:

- *Procedural-oriented heuristics* (heurísticas de orientação procedural): crianças utilizam comandos em certa ordem lógica para cumprir a tarefa. Tem como objetivo avaliar se a criança consegue programar a sequência inteira (com uma sequência de blocos de programação que permitem comandar sensores e atuadores, por exemplo, "avançar se detectado que a cor é verde"), executando toda a sequência da tarefa;
- *Conceptual-oriented heuristics* (heurísticas de orientação conceitual): crianças usam os comandos a fim de compreender o seu funcionamento, e, em seguida, para realizar a tarefa. Objetiva verificar se as crianças conseguem programar e executar partes da sequência separadamente (por exemplo, "detectar cor"), a fim de testá-las antes de passar para executar toda a sequência;
- *Meta-cognitive-oriented heuristics* (heurísticas de orientação meta-cognitiva): as crianças usam os comandos a fim de explorar o funcionamento do robô, ou seja, o que ele pode fazer e o que ele não pode fazer. Tem como objetivo verificar se elas conseguem explorar todos os parâmetros de um bloco (por exemplo, se conseguem explorar o parâmetro intensidade de luz do sensor de luz) para entender seu funcionamento antes resolver uma determinada tarefa.

O trabalho apresenta um *framework* para avaliação que explora aspectos interessantes em um processo pedagógico, mas não foca em aspectos técnicos apresentados nesta dissertação, como o referente à construção da solução e os conceitos de programação envolvidos, e também não define nenhuma metodologia de desenvolvimento de atividades ou tarefas. O trabalho discute o potencial semiótico da utilização da RP, mas não explora nenhum tipo de processo, como aqueles existentes na SO. Possui um caráter de aplicação mais generalista, tendo o processo de ensino de Desenvolvimento de Algoritmos como algo auxiliar ou paralelo, inserido em um contexto pedagógico mais amplo.

Ambos os trabalhos analisados apresentam alguns aspectos semelhantes ao método proposto nessa dissertação, mas nenhum deles contempla todos os elementos do PDAP-RP e podem ser aplicados na mesma abrangência. O trabalho de Catlin e Blamires (2010) apresenta um *framework* com um caráter formal de orientação do desenvolvimento e avaliação de práticas, mas não se fundamenta nos principais processos da SO (*i.e.*, PAM, SAM e NAM), com isso não contempla muito das soluções que o PDAP-RP fornece nas etapas de desenvolvimento da prática e principalmente nos recursos avaliativos.

Em Gaudiello, Zibetti e Pinaud (2012) é apresentado apenas um método de avaliação com foco em questões pedagógicas, sem abranger conceitos técnicos e o desenvolvimento das práticas. Se comparado com o método PDAP-RP, esse método atua apenas nas etapas de avaliação, mas sem oferecer um processo formal e sequenciado, que pode apresentar componentes interessantes, tanto no aprendizado dos alunos, quanto ao fornecer recursos de avaliação ao professor.

7. CONCLUSÃO

Este capítulo apresenta a conclusão desta dissertação, contendo as contribuições da pesquisa, trabalhos futuros e considerações finais. A Seção 7.1 destaca as principais contribuições da pesquisa sobre as questões que delinearão todo o trabalho e os avanços resultantes deste estudo. A Seção 7.2 apresenta uma discussão sobre os trabalhos futuros, destacando limitações da pesquisa e método proposto, assim como demandas por novas pesquisas. Na Seção 7.3 são apresentadas as considerações finais com discussões de caráter pessoal e crítico-analítico.

7.1. CONTRIBUIÇÕES DA PESQUISA

Com o objetivo de responder duas principais questões de pesquisa apresentadas no início dessa dissertação: (1) “Como a RP contribui como ferramenta de aprendizado e apoio no ensino de Desenvolvimento de Algoritmos?” e; (2) “Como um método formal e sistemático pode auxiliar no desenvolvimento e avaliação de práticas usando RP?”, este trabalho conduziu um estudo capaz de analisar as potencialidades da RP como ferramenta pedagógica e a efetividade de um método em um ambiente de ensino de Desenvolvimento de Algoritmos apoiado pela RP.

Entre os desafios relacionados à aplicação da RP destacava-se, principalmente, a ausência de uma metodologia capaz de delimitar todos os objetivos e nortear as avaliações. Com a proposta de um processo como o PDAP-RP, baseado nos métodos da SO, foi possível apresentar uma solução, que produz avanços sobre as questões relacionadas a esse desafio. Através da SO, foi possível articular um método que pudesse abranger toda a demanda do desenvolvimento (*i.e.*, definição do problema, delimitação do cenário, apresentação dos elementos, definição de procedimentos) e avaliação (*i.e.*, objetivos gerais e específicos, métricas de avaliação) de práticas.

Dentro desta metodologia, é necessário, para o desenvolvimento de práticas, o levantamento de quais conceitos devem ser abordados, de que maneira serão trabalhados e quem será envolvido nesse processo (método PAM). A partir dessa etapa, uma análise semântica (AS) apresenta uma maneira sistemática de formação de práticas pedagógicas e identifica os elementos do domínio e seus comportamentos (método SAM). A definição de normas apresenta-se como uma maneira, a qual o professor pode definir os

comportamentos esperados nas tarefas em uma determinada prática, consequentemente, elencando as responsabilidades do aluno (agente) e os recursos a serem utilizados na execução (*affordances*) (método NAM).

Dentre as principais contribuições deste trabalho, é possível citar em resumo:

- Desenvolvimento de um método sistemático capaz de orientar o desenvolvimento e avaliação de práticas usando RP, por meio de uma proposta inovadora, utilizando técnicas provenientes da SO;
- Metodologia que pode ser aplicada através da visão de professor-pesquisador ou de um professor-aplicador. O método PDAP-RP possibilita a um professor-pesquisador, com conhecimento e vivência sobre as áreas de RP e SO, várias soluções que podem ser exploradas e adaptadas em cenários específicos de ensino-aprendizagem. Para o professor-aplicador, apresenta uma solução que encadeia uma série de passos para a formação, aplicação e avaliação de práticas utilizando a RP;
- Resultados de estudo do impacto da RP no ensino de Desenvolvimento de Algoritmos no nível técnico-profissionalizante, através da influência dos aspectos semânticos e pragmáticos em um ambiente de ensino/aprendizagem, e a síntese desses resultados tanto no caráter qualitativo quanto quantitativo.

Os resultados obtidos através dos experimentos mostraram que o método PDAP-RP tem potencial para o desenvolvimento e o fornecimento de instrumentos de avaliação de práticas com RP. O diferencial deste processo é a utilização SO na formatação de práticas, assim como a avaliação de resultados através de uma perspectiva orientada a normas.

7.2. TRABALHOS FUTUROS E LIMITAÇÕES DA PESQUISA

Entre as principais limitações da metodologia, podemos citar a deficiência em tratar as situações em que ocorrem respostas inesperadas ou inconsistentes em ações do robô, sem que a programação fosse responsável por tais erros. O PDAP-RP não possui meios de intervir ou prever nessas situações, dado que o método não é direcionado a nenhuma plataforma robótica. Estudos futuros poderão agregar maneiras de intervir em casos em que ocorram essas situações adversas, para que não haja a ruptura no processo

de aplicação ou que não interfira nos resultados esperados através da aplicação do método.

Outra limitação apresentada durante a pesquisa foi o nível de granularidade obtida nos resultados das avaliações. Portanto, é necessário um refinamento no desenvolvimento de formas de avaliação as quais consiga demonstrar de forma mais clara as percepções dos alunos perante o método, de modo a identificar com maior precisão falhas no processo. Por exemplo, entre as respostas dadas pelos alunos no questionário para a avaliação qualitativa, muitas delas não apontavam as causas das insatisfações ou dificuldades apresentadas por eles.

Como próximo passo deste trabalho, serão feitos estudos que possam revisar todas as etapas do PDAP-RP, para apontar melhorias ou integrar novos elementos que possam colaborar com o processo. Para isso, serão conduzidos experimentos com grupos maiores e mais heterogêneos em relação aos níveis de proficiência em Desenvolvimento de Algoritmos. Também é esperada a condução do experimento em um público diferente, em alunos que estejam cursando o ensino superior, fornecendo uma nova visão da aplicação do método.

7.3. CONSIDERAÇÕES FINAIS

A RP para o ensino de programação de computadores é um recurso pedagógico de grande potencial, tanto em questões ligadas a motivação dos alunos, quanto na diversidade de ferramentas e suporte que podem ser oferecidos aos professores. Entretanto, sua aplicação em sala de aula não é uma tarefa trivial e deve ser amparada por métodos que possam conduzir a aprendizagem e fornecer instrumentos avaliativos eficazes. A existência de um método formal no desenvolvimento de práticas com RP possibilita ao professor um recurso metodológico, que conduz a criação dessas práticas de maneira mais consistente e coesa com os objetivos pedagógicos esperados pelos alunos.

Na literatura encontramos diversos trabalhos que atestam os benefícios educacionais e motivacionais que a RP provê, mas ainda faltam pesquisas que mostrem a sua eficiência de maneira sistemática e envolvida em um cenário de condução do ensino de Desenvolvimento de Algoritmos.

Os métodos discutidos nesta dissertação tiveram como objetivo a formatação de práticas e a avaliação de resultados através de uma perspectiva orientada a aspectos pragmáticos, sociais e comportamentais, formando um modelo de avaliação no ensino de Desenvolvimento de Algoritmos.

Como resultado da pesquisa realizada neste trabalho foi desenvolvido um método (o PDAP-RP) que pudesse contemplar etapas de desenvolvimento e avaliação de práticas utilizando RP, amparado pelos principais recursos da SO. Este método proporciona um conjunto de artefatos e guias que trazem uma nova perspectiva ao desenvolvimento de práticas, com etapas que cobrem desde a identificação do cenário inicial até a aplicação e avaliação junto aos alunos.

A análise semântica e de normas da SO trazem recursos metodológicos que podem favorecer o desenvolvimento e condução de práticas com robôs, pois possibilitam ao professor criar uma análise do cenário, conseguindo previamente definir as responsabilidades e o comportamento esperado dos alunos diante de uma prática pedagógica.

Entretanto, as soluções propostas pelo PDAP-RP não esgotam o problema em absoluto, mas possibilitam uma visão de como a adoção de um método sistemático pode proporcionar benefícios em um ambiente de ensino/aprendizagem amparado pela RP. Este trabalho remete a uma proposta de solução que avança no sentido de sistematizar esse processo pedagógico e, ao mesmo tempo, fornecer recursos precisos de construção e avaliação.

Através do experimento realizado foi possível evidenciar aspectos positivos na adoção do método PDAP-RP. Os resultados apontaram uma melhoria significativa no desempenho e melhor aceitação dos grupos de alunos que utilizaram o PDAP-RP, tanto em uma análise em comparação com o grupo de controle (que realizou as práticas sem a utilização do método), quanto em uma observação por parte do professor-aplicador. Para evidenciar a eficiência do PDAP-RP foi conduzido um experimento com quarenta alunos com diferentes níveis de conhecimento e proficiência em Desenvolvimento de Algoritmos, que teve o intuito de avaliar o desempenho e as respostas dos alunos na presença de um método formal, e também identificar deficiências e apontar melhorias.

Com os resultados obtidos a partir desse experimento foi possível visualizar aspectos positivos na adoção do PDAP-RP, assim como pontos a serem melhorados e aspectos relacionados à RP que o método não abrange, não colabora ou é limitado. Esta pesquisa teve um caráter exploratório, utilizando uma amostra pequena para o experimento, mas que permitia a definição do problema e a formulação de uma proposta de solução, através de uma investigação sobre as potenciais dificuldades que os alunos possuem em práticas utilizando RP.

A utilização da RP não é um algo trivial e demanda grande dedicação por parte do professor, portanto é essencial prover meios para auxiliar nesse processo. O uso de uma metodologia como o PDAP-RP proporciona uma visão na qual o ambiente de aprendizagem se apresenta como um sistema de signos, o qual pode ser analisado através de seus aspectos semânticos, pragmáticos e sociais descritos em *affordances* e normas. A compreensão desse sistema auxiliou na identificação de quais são as etapas fundamentais para a construção de práticas que possam contemplar toda estrutura didática, da concepção à avaliação.

Além disso, o método contribuiu, potencializando a uso da RP, além dos aspectos motivacionais, proporcionando a possibilidade de estabelecer atividades organizadas que podem ser revistas para uma melhoria contínua. A busca pela sistematização da criação de práticas possibilita identificar pontos específicos no qual houve erros durante sua elaboração.

Por fim, em uma análise final e pessoal, acredito que esse trabalho colabora em ambas as áreas que o mesmo abrange (RP e SO). O método PDAP-RP incentiva futuros trabalhos que possam explorar ainda mais o problema principal o qual ele se propõe a resolver, que é aprimorar cada vez mais os meios de aplicação da RP. Em relação aos estudos sobre SO, este trabalho contribui por apresentar uma aplicação em um cenário pouco explorado, o de Educação, podendo motivar assim a criação de outros métodos através de técnicas oriundas da SO, que possam colaborar com essa área de estudo.

Para finalizar, saliento a importância desse trabalho em minha formação pessoal e profissional, tanto como pesquisador quanto como educador. Ao final do estudo, pude ter uma visão mais abrangente e clara da construção do conhecimento por parte de meus alunos e a importância da busca por inovação em um ambiente de ensino, motivando-me a continuar pesquisas com esse propósito.

8. REFERÊNCIAS BIBLIOGRÁFICAS

- Alimisis, D. (2013). Educational robotics: Open questions and new challenges. *Themes in Science and Technology Education*, vol 6(1), 63. Disponível em: <http://earthlab.uoi.gr/theste/index.php/theste/article/download/119/85>. Acesso em: 23 mai 2014.
- Alimisis, D. (2009). Teacher Education on Robotics-Enhanced Constructivist Pedagogical Methods. *School of Pedagogical and Technological Education, Athens*. Disponível em: http://dide.ilei.sch.gr/keplinet/education/docs/book_TeacherEducationOnRobotics-ASPETE.pdf. Acesso em: 23 mai 2014.
- Ambrósio, A. P. L., Almeida, L. S., Macedo, J., Santos, A., & Franco, A. H. R. (2011). Programação de computadores: compreender as dificuldades de aprendizagem dos alunos. *Revista Galego-Portuguesa de Psicoloxia e Educación*. vol 19 (1). Disponível em: http://repositorium.sdum.uminho.pt/bitstream/1822/15554/1/Programa%C3%A7%C3%A3o%20de%20computadores_compreender%20as%20dificuldades%20de%20aprendizagem%20dos%20alunos.pdf. Acesso em: 10 jan 2014.
- Benitti, F. B. V. (2012). Exploring the educational potential of robotics in schools: A systematic review. *Computers & Education*, vol 58(3), 978-988.
- Benitti, F. B. V., Krueger, M. L., e Leonardo, D. (2010). Robótica como Elemento Motivacional para Atração de Novos Alunos para Cursos de Computação. Disponível em: https://www.inf.furb.br/dsc/download/ciesc2010_submission_16.pdf. Acesso em: 22 fev 2014.
- Bini, E. M. (2010). *Ensino de programação com ênfase na solução de problemas*. Dissertação de Mestrado Profissional de Ensino de Ciência e Tecnologia, Universidade Tecnológica Federal do Paraná, Ponta Grossa – PR.
- Blikstein, P. (2013). Digital fabrication and ‘making’ in education: The democratization of invention. In J. Walters-Herrmann & C. Boching (eds.), *Fablabs: Of Machines, Makers and Inventors*, 1-21. Bielefeld: Transcript Publishers.
- Bredenfeld, A., Hofmann, A., e Steinbauer, G. (2010). Robotics in education initiatives in europe-status, shortcomings and open questions. In *Workshop Proceedings of Intl.*

- Conf. on Simulation, Modeling and Programming for Autonomous Robots (SIMPAR 2010)*, 568-574.
- Brennan, K., e Resnick, M. (2012). New frameworks for studying and assessing the development of computational thinking. In *Proceedings of the 2012 annual meeting of the American Educational Research Association, Vancouver, Canada*.
- Bonacin, R. (2004). *Um modelo de desenvolvimento de sistemas para suporte a cooperação fundamentado em design participativo e semiótica organizacional*. Tese de Doutorado em Ciência da Computação, Instituto de Computação, Universidade Estadual de Campinas, Campinas-SP.
- Bonacin, R., Simoni, C. A. C., Melo, A. M., Baranauskas, M. C. C., e Liu, K. (2006). Organisational Semiotics: guiding a service-oriented architecture for e-government. In: (Ed.). *International Conference on Organisational Semiotics*, 1., 2006, Unicamp/IC. p. 7-58. Campinas-SP
- Borges, M. A. (2000). Avaliação de uma metodologia alternativa para a aprendizagem de programação. In *VIII Workshop de Educação em Computação–WEI*.
- Buzin, P. F. W. K. (2001) A epistemologia da Ciência da Computação: Desafio do Ensino dessa Ciência. *Revista de Educação, Ciência e Cultura*, vol 6 (2). Centro Universitário La Salle. Canoas, RS.
- Campos, R. L. B (2010). *ERM2C: Uma metodologia para melhoria do ensino-aprendizado de lógica de programação*. Disponível em: <http://www.fejal.br/erbase2010/papers/weibase/65502.pdf>. Acesso em: 03 dez 2013.
- Carnegie (2014). Carnegie Mellon Robotic Academy. *Introduction to Programming LEGO® Mindstorm® EV3 – Teacher’s Guide* [Apostila]. Disponível em: <http://www.education.rec.ri.cmu.edu/content/lego/ev3/files/EV3%20teachers%20guideWEB.pdf> . Acesso em: 12 ago 2014.
- Catlin, D., e Blamires, M. (2010). The Principles of Educational Robotic Applications (ERA): A framework for understanding and developing educational robots and their activities. In *Proceedings for Constructionism 2010 The 12th EuroLogo conference*, Paris, França.
- Chandler, D. (2007). *Semiotics: the basics*. London: Routledge (ed.).

- d'Abreu, J. V. V., e CHELLA, M. (1999). Desenvolvimento de ambientes de aprendizagem baseados no uso de dispositivos robóticos. *Anais do X Simpósio Brasileiro de Informática na Educação*, 9-16.
- d'Abreu, J. V. (2002). *Integração de Dispositivos Mecatrônicos para Ensino-Aprendizagem de Conceitos na Área de Automação*. Tese de Doutorado em Engenharia Mecânica, Faculdade de Engenharia Mecânica, Universidade Estadual de Campinas, Campinas-SP.
- d'Abreu, J. V. V., e de Fátima Garcia, M. (2010). Robótica Pedagógica e Currículo. In *Workshop de Robótica Educacional–WRE. Proceedings of the Joint Conference 2010-SBIA-SBRN-JRI, Workshops*, 01-06.
- Delgado, C., Xexeo, J. A., Souza, I. F., Campos, M., e Rapkiewicz, C. E. (2004). Uma abordagem pedagógica para a iniciação ao estudo de algoritmos. In *XII Workshop de Educação em Computação*.
- Deters, J. I., da Silva, J. M. C., de Miranda, E. M., e Fernandes, A. M. R. (2008). O Desafio de Trabalhar com Alunos Repetentes na Disciplina de Algoritmos e Programação. *Simpósio Brasileiro de Informática na Educação*.
- dos Santos, R. P., e Costa, H. A. X. (2006). Análise de Metodologias e Ambientes de Ensino para Algoritmos, Estruturas de Dados e Programação aos iniciantes em Computação e Informática. *Infocomp, Journal of Computer Science*, vol 5(1).
- Eco, U. (1976) *Tratado geral de Semiótica*. Editora Perspectiva.
- Eguchi, A. (2010). What is Educational Robotics? Theories behind it and practical implementation. In *Society for Information Technology & Teacher Education International Conference* vol. 1 (1), 4006-4014.
- Falkner, K., e Palmer, E. (2009). Developing authentic problem solving skills in introductory computing classes. In *ACM SIGCSE Bulletin*, vol. 41 (1), 4-8. ACM.
- Ferreira, J. J., de Souza, C. S., de Castro Salgado, L. C., Slaviero, C., Leitão, C. F., e de Moreira, F. (2012). Combining cognitive, semiotic and discourse analysis to explore the power of notations in visual programming. In *Visual Languages and Human-Centric Computing (VL/HCC), 2012 IEEE Symposium*, 101-108. IEEE.

- Freire, F. M., e Prado, M. E. B. B. (1996). Professores construcionistas: a formação em serviço. In *Actas do III Congresso Ibero-Americano de Informática Educativa*. Disponível em: <http://lsm.dei.uc.pt/ribie/docfiles/txt200352145836PROFESSORES%20CONSTRUCIONISTAS.pdf>. Acesso em: 08 jan 2014.
- Forbellone, A. L. V., e Eberspächer, H. F. (1993). *Lógica de programação: a construção de algoritmos e estruturas de dados* (Vol. 3). São Paulo: Makron Books.
- Gaudiello, I., Zibetti, E., & Pinaud, C. A. (2012). Control heuristics for educational robots: a pilot study. In *Proceedings of 3rd International workshop teaching robotics, teaching with robotics*. Integrating robotics in school curriculum, Riva del Garda, Trento, Italy, 67-75.
- Gibson, J. J. (1968). *The Ecological Approach to Visual Perception*. Houghton Mifflin Company, Boston, Massachusetts, 127-143.
- Gomes, A., e Mendes, A. J. (2007). Learning to program-difficulties and solutions. In *International Conference on Engineering Education–ICEE* (Vol. 2007). Disponível em: <http://ineer.org/Events/ICEE2007/papers/411.pdf>. Acesso em: 12 fev 2014.
- Gomes, A., Henriques, J., e Mendes, A. (2008). Uma proposta para ajudar alunos com dificuldades na aprendizagem inicial de programação de computadores. *Educação, Formação & Tecnologias – ISSN 1646-933X*, 93-103. Disponível em: <http://www.eft.educom.pt/index.php/eft/article/download/23/16>. Acesso em: 12 fev 2014
- Gomes, A., Areias, C., Henriques, J., e Mendes, A. J. (2008). Aprendizagem de programação de computadores: dificuldades e ferramentas de suporte. *Revista Portuguesa de Pedagogia*, v. 42, n. 2.
- Gudwin, R. R. (2002). *Semiônica: Uma Proposta de Contribuição à Semiótica Computacional*. Tese para obtenção do Título de Professor Livre-Docente, Faculdade de Engenharia Elétrica e de Computação, Universidade Estadual de Campinas, Campinas-SP.
- Huet, I., Pacheco, O. R., Tavares, J., e Weir, G. (2004). New challenges in teaching introductory programming courses: a case study. In *Frontiers in Education, 2004. FIE 2004. 34th Annual*, T2H-5. IEEE.

- Jenkins, T., (2002). On the difficulty of learning to program. *Proceedings of the 3rd Annual LTSN_ICS Conference*, Loughborough University, United Kingdom, 53-58.
- Júnior, J. C. R. P., e Rapkiewicz, C. E. (2004). O Processo de Ensino-Aprendizagem de Fundamentos de Programação: Uma Visão Crítica da Pesquisa no Brasil. In *Anais do XII Workshop sobre Educação em Computação (SBC)*.
- Kim, S. H., e Jeon, J. W. (2007). Programming LEGO Mindstorms NXT with visual programming. In *Control, Automation and Systems, 2007. ICCAS'07. International Conference on*, 2468-2472. IEEE.
- Lego (2013), LEGO.com Website. Disponível: www.lego.com/en-us/. Acesso em: 16 set 2013.
- Lego (2014), LEGO.com Website. Disponível: <http://www.lego.com/en-us/mindstorms/products/ev3/>. Acesso em: 05 set 2014
- Liu, K. (2000). *Semiotics in Information System Engineering*. Cambridge: Cambridge University Press, XII, 218 p.
- Lopes, D. Q. (2008). A exploração de modelos e os níveis de abstração nas construções criativas com robótica educacional. *Informática na educação: teoria & prática*, vol. 11(2).
- Maloney, J. H., Peppler, K., Kafai, Y., Resnick, M., e Rusk, N. (2008). Programming by choice: urban youth learning programming with scratch. In *ACM SIGCSE Bulletin*, vol. 40 (1), 367-371. ACM.
- MEC/CNE (2014A). *Ministério da Educação – Conselho Nacional de Educação*. Relatório: Diretrizes Curriculares Nacionais para os cursos de graduação em Computação. Disponível em: <http://portal.mec.gov.br/index.php>. Acesso em: 10 jan 2014
- MEC/CNE (2014B). *Ministério da Educação – Conselho Nacional de Educação*. Diretrizes curriculares nacionais para a Educação Profissional de Nível Técnico. Disponível em: <http://portal.mec.gov.br/setec/arquivos/pdf1/ceb016.pdf>. Acesso em: 10 jan 2014.

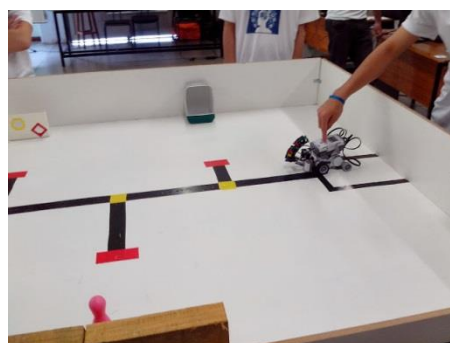
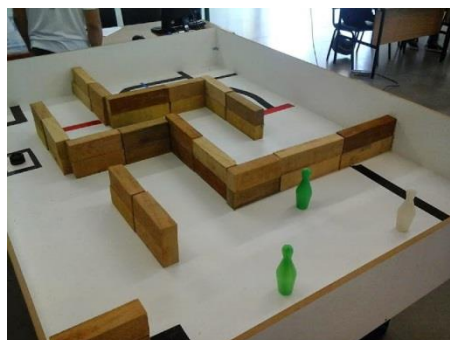
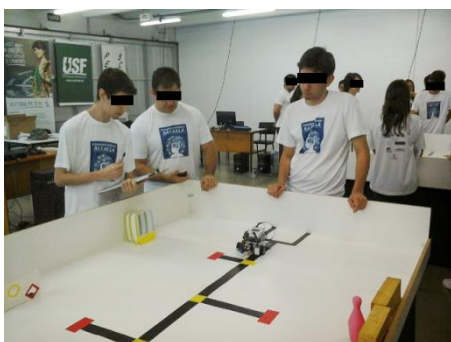
- Melo, A. M. (2007). *Design inclusivo de sistemas de informação na Web*. Tese Doutorado em Ciência da Computação, Instituto de Computação, Universidade Estadual de Campinas, Campinas-SP.
- Menezes, E. T. e Santos, T.H. (2002). "Avaliação somativa" (verbetes). Dicionário Interativo da Educação Brasileira - EducaBrasil. São Paulo: Midiamix Editora, Disponível em: <http://www.educabrasil.com.br/eb/dic/dicionario.asp?id=299>. Acesso em: 6 nov 2014.
- Morris, C. W. (1938). *Foundations of the Theory of Signs, International Encyclopedia of Unified Science*. University of Chicago Press.
- NIED (2014). *Núcleo de informática aplicada a Educação*. Disponível em: <http://www.nied.unicamp.br>. Acesso em: 24 jan 2014.
- Ortiz, J., Bustos, R., e Ríos, A. (2011). System of indicators and methodology of evaluation for the robotics in classroom. In *Proceedings of the 2nd International Conference on Robotics in Education (RiE 2011)*, 63-70.
- Papert, S. (1980). *Mindstorms: Children, computers, and powerful ideas*. New York: Basic Books, Inc.(ed.).
- Papert, S. (1994). *A máquina das crianças: repensando a escola na era da informática*. Porto Alegre: Artmed.
- Peirce, C. S. (1931-1958). *Collected Papers*, Cambridge, Mass: Harvard University Press.
- Resnick, M., Maloney, J., Monroy-Hernández, A., Rusk, N., Eastmond, E., Brennan, K., Millner, A., Rosenbaum, E., Silver, J., Silverman, B., e Kafai, Y. (2009). Scratch: programming for all. *Communications of the ACM*, vol. 52(11), 60-67.
- Saeli, M., Perrenet, J., Jochems, W. M., e Zwaneveld, B. (2011). Teaching programming in secondary school: a pedagogical content knowledge perspective. *Informatics in Education-An International Journal*, vol. 10 (1), 73-88.
- Santaella, L. (1995), *Teoria Geral dos Signos: Semiosis e Autogeração*. São Paulo: Editora Ática.

- Silva, A. F. (2009). *RoboEduc: Uma Metodologia de Aprendizado com Robótica Educacional*. Tese de Doutorado em Engenharia Elétrica, Universidade Federal do Rio Grande do Norte, Natal-RN.
- Silva Filho, R. L. L., Motejunas, P. R., Hipólito, O., e Lobo, M. B. C. M. (2007). A evasão no ensino superior brasileiro. *Cadernos de Pesquisa*, vol. 37(132), 641-659.
- Sleeman, D. (1986). The challenges of teaching computer programming. *Communications of the ACM*, vol. 29(9), 840-841.
- Soares, R. F., e Borges, M. A. F. (2011). Robótica: aprendizado em informática de forma lúdica. In *WEI-XXXI Congresso da Sociedade Brasileira de Computação*, Natal-RN: SBC.
- Sommerville, I. (2011). *Engenharia de software*. Vol 6. São Paulo: Pearson Prentice Hall.
- Stamper, R. K. (1973). *Information in Business and Administrative Systems*, Hoboken: John Wiley & Sons (ed.).
- Stamper, R. (1994). Social norms in requirements analysis: an outline of MEASUR. In *Requirements engineering* (pp. 107-139). San Diego: Academic Press Professional, Inc..
- Stamper, R., Liu, K., Hafkamp, M., e Ades, Y. (2000). Understanding the roles of signs and norms in organizations-a semiotic approach to information systems design. *Journal of Behaviour & Information Technology*, vol 19(1), 15-27.
- Saussure, F. (2004) *Curso de linguística geral*. 26ª. ed. São Paulo: Pensamento - Cultrix.
- Softex (2012). Software e Serviços de TI: A Indústria Brasileira em Perspectiva, vol. 2 (1). Disponível em: http://publicacao.observatorio.softex.br/_publicacoes/arquivos/2012/completa/Sw&ServTI_versao_completa.pdf. Acesso em: 28 set 2014.
- Utting, I., Cooper, S., Kölling, M., Maloney, J., e Resnick, M. (2010). Alice, Greenfoot, and Scratch--a discussion. *ACM Transactions on Computing Education (TOCE)*, vol. 10(4), 17.
- Valente, J. A. (1993). Por quê o computador na educação. *Computadores e Conhecimento: repensando a educação*. Campinas: Gráfica da UNICAMP.

- Wainer, J. (2007). Métodos de pesquisa quantitativa e qualitativa para a Ciência da Computação. *Atualização em informática*, 221-262.
- Winslow, L. E. (1996). Programming pedagogy—a psychological overview. *ACM SIGCSE Bulletin*, vol. 28(3), 17-22.
- Zanetti, H.A.P., Bonacin, R.(2014A). Uso de semiótica e análise de normas em práticas de ensino de Programação de Computadores utilizando Robótica Pedagógica. *Revista Eletrônica de Tecnologia e Cultura*, vol. 1(1).
- Zanetti, H.A.P., Bonacin, R.(2014B). Uma Metodologia Baseada em Semiótica para Elaboração e Análise de Práticas de Ensino de Programação com Robótica Pedagógica. *Anais do Simpósio Brasileiro de Informática na Educação*, vol. 25 (1).
- Zanetti, H.A P, de Souza, A. L., d'Abreu, J. V., e Borges, M. A. (2012). Uso de robótica e jogos digitais como sistema de apoio ao aprendizado. *Jornada de Atualização em Informática na Educação*, vol. 1(1), 142-161.
- Zilli, S. R. (2004). *A robótica educacional no ensino fundamental: Perspectivas e prática*. Dissertação de Mestrado em Engenharia, Universidade Federal de Santa Catarina, Centro Tecnológico, Florianópolis-SC.

ANEXO I – INFORMAÇÕES SOBRE A COMPETIÇÃO R.A.F.A.E.L.A.

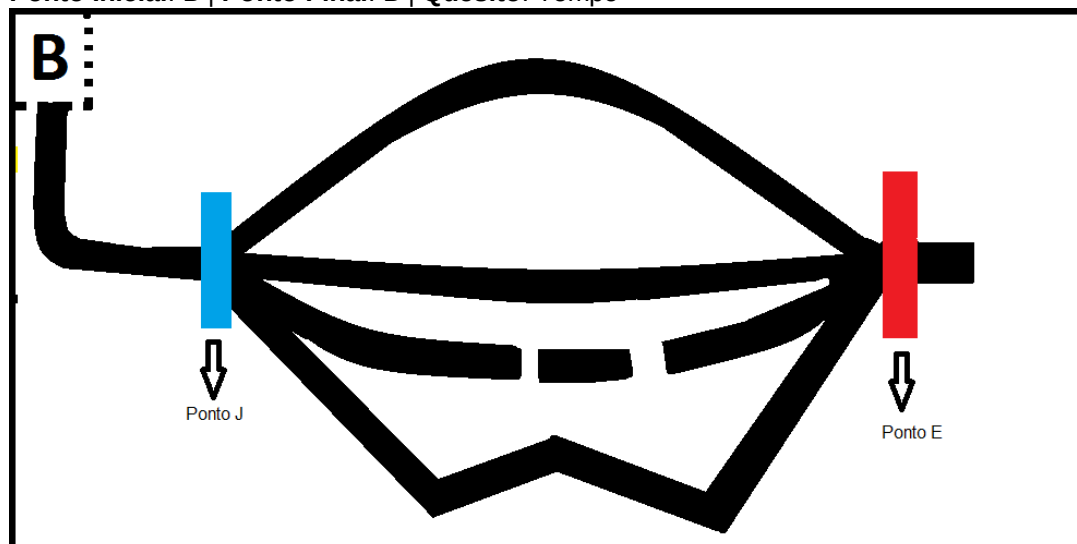
ITEM A – IMAGENS DA COMPETIÇÃO



ITEM B – EXEMPLO DE PROVA REALIZADA DA COMPETIÇÃO

Prova P1 – LineFollower

Ponto Inicial: B | Ponto Final: E | Quesito: Tempo



Descrição:

O robô deverá iniciar a prova no ponto inicial B e deverá percorrer a linha preta até a faixa azul que se encontra no ponto J. A partir do ponto J, o robô deve escolher e seguir dentre as faixas que se originam do ponto J, a faixa que leva menos tempo para chegar na faixa vermelha no ponto E. Ao alcançar a faixa vermelha no ponto E, o robô deverá retornar até o ponto inicial B.

Pontuação:

Objetivo	Pontos
Completar a prova	200
Chegar somente até a faixa vermelha	100
Completar a prova em menor tempo	+50
Completar a prova sem utilizar as faixas como parâmetro	-200
Colisões na arena	-50
*Desvio da faixa preta	-10

Pontuação Máxima: 200/250 pts para quem completar a prova em menor tempo.

*Está pontuação só é válida caso o robô desvie da faixa preta e retorne sem desviar de seu curso de execução. Caso o robô saia da faixa preta e não retorne nenhuma pontuação será retirada e o robô será desclassificado por não seguir o objeto principal da prova.

Regras:

- ✓ O robô será montado pela equipe organizadora e não poderá ser alterado.
- ✓ Após o início da prova e a execução do programa, os competidores não poderão entrar mais em contato com o robô. O programa deverá finalizar-se automaticamente ao completar a tarefa proposta.
- ✓ As duplas deverão nomear o programa com o padrão: **dupla0X_P1**, sendo X a identificação numérica da equipe que será sorteada no dia.

Tempo Máximo de Apresentação: 10 minutos.

ITEM C – QUESTIONÁRIO APLICADO AOS ALUNOS

Questionário 1- Avaliação do Evento RAFAELA

Prezado colega, ao final do evento RAFAELA, gostaríamos muito de conhecer a sua opinião sobre alguns aspectos relacionados à organização, comunicação infraestrutura etc. para que possamos aprimorar esta atividade em edições futuras. Para isso, pedimos que responda à avaliação abaixo. Obrigado!

*Obrigatório

1. Organização do evento: *

- ☐ Excelente
- ☐ Muito Boa
- ☐ Boa
- ☐ Razoável
- ☐ Péssima

2. Infraestrutura (computadores, mesas, espaços, arenas): *

- ☐ Excelente
- ☐ Muito Boa
- ☐ Boa
- ☐ Razoável
- ☐ Péssima

3. Comunicação entre a organização e os competidores: *

- ☐ Excelente
- ☐ Muito Boa
- ☐ Boa
- ☐ Razoável
- ☐ Péssima

4. Atendimento/suporte aos competidores: *

- ☐ Excelente
- ☐ Muito Boa
- ☐ Boa
- ☐ Razoável
- ☐ Péssima

5. Alimentação: *

- ☐ Excelente
- ☐ Muito Boa
- ☐ Boa
- ☐ Razoável
- ☐ Péssima

6.Premiação: *

- ☐ Excelente
- ☐ Muito Boa
- ☐ Boa
- ☐ Razoável
- ☐ Péssima

7.Duração do evento: *

- ☐ Muito Longa
- ☐ Longa
- ☐ Adequado
- ☐ Breve
- ☐ Muito Breve

8.Tempo para preparação e para a execução das provas: *

- ☐ Muito Longo
- ☐ Longo
- ☐ Adequado
- ☐ Breve
- ☐ Muito Breve

9.Clareza no texto utilizado na descrição das provas. *

- ☐ Excelente
- ☐ Muito Boa
- ☐ Boa
- ☐ Razoável
- ☐ Péssima

10.Clareza no texto e explicação oral das regras da competição. *

- ☐ Excelente
- ☐ Muito Boa
- ☐ Boa
- ☐ Razoável
- ☐ Péssima

11.Nível de dificuldade das tarefas: *

- ☐ Muito Difícil
- ☐ Difícil
- ☐ Adequado
- ☐ Fácil
- ☐ Muito Fácil

12.Você participaria novamente deste evento, caso ele seja novamente realizado? *

- ☐ Sim
- ☐ Não

12. Você participaria novamente deste evento, caso ele seja novamente realizado? *

- ☐ Sim
- ☐ Não

13. Qual era seu nível de interesse na área de Engenharia antes de sua participação, sendo 1 correspondendo a nenhum interesse e 5 interesse total: *

- ☐ 1
- ☐ 2
- ☐ 3
- ☐ 4
- ☐ 5

14. Se você tinha algum interesse na área de Engenharia antes de participar do evento, indique qual a modalidade que mais lhe atraía: *

- ☐ Civil
- ☐ Computação
- ☐ Elétrica
- ☐ Mecânica
- ☐ Outro:

15. Qual é seu atual nível de interesse na área de Engenharia após a sua participação no evento, sendo 1 correspondendo a nenhum interesse e 5 interesse total: *

- ☐ 1
- ☐ 2
- ☐ 3
- ☐ 4
- ☐ 5

16. Após a sua participação no evento, a sua modalidade de interesse mudou? Se sim indicar abaixo:

OBS: Caso você não selecione nenhuma alternativa abaixo, entenderemos como: a sua modalidade de interesse não mudou.

- ☐ Civil
- ☐ Computação
- ☐ Elétrica
- ☐ Mecânica
- ☐ Outro:

17. Comentários: *

Enviar

Nunca envie senhas em Formulários Google.

Powered by
 Google Drive

Este conteúdo não foi criado nem aprovado pelo Google.
[Denunciar abuso](#) - [Termos de Serviço](#) - [Termos Adicionais](#)

ANEXO II – ARTIGO PUBLICADO NA RETC

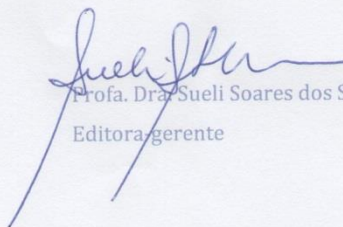


Certificamos para os devidos fins que o artigo intitulado *Uso de semiótica e análise de normas em práticas de ensino de Programação de Computadores utilizando Robótica Pedagógica*, de autoria de Humberto Augusto Piovesana Zanetti e Rodrigo Bonacin, foi publicado na 14a. Edição da Revista Eletrônica de Tecnologia e Cultura (RETC), cuja política editorial segue a Licença *Creative Commons Attribution*, permitindo o compartilhamento do trabalho para fins acadêmicos, podendo ser reproduzida ou transmitida por qualquer meio, sem previa autorização por escrito, desde que citadas as fontes e os autores do trecho reproduzido.

Atenciosamente



Prof. Dr. Emerson Freire
Editor gerente



Profa. Dra. Sueli Soares dos Santos Batista
Editora gerente



RETC

Revista Eletrônica de Tecnologia e Cultura
www.revista-fatecjd.com.br

USO DE SEMIÓTICA E ANÁLISE DE NORMAS EM PRÁTICAS DE ENSINO DE PROGRAMAÇÃO DE COMPUTADORES UTILIZANDO ROBÓTICA PEDAGÓGICA

Humberto Augusto Piovesana ZANETTI
Fatec Jundiaí e Etec Rosa Perrone Scavone - CEETEPS
Mestrando em Ciência da Computação
prof.humberto@fatecjd.edu.br

Rodrigo BONACIN
Faculdade de Campo Limpo Paulista - FACCAMP
Pós-doutor em Ciência da Computação
rbonacin@cc.faccamp.br

RESUMO

Este trabalho aborda o uso de Robótica Pedagógica no ensino de programação de computadores, utilizando conceitos de normas e *affordances* da Semiótica Organizacional para análise de práticas pedagógicas e avaliações do aprendizado. Entre os grandes desafios no ensino de programação de computadores estão a complexidade dos conceitos e o alto nível de abstração necessário. Com o uso de Robótica Pedagógica é possível trazer vários recursos benéficos para lidar com esses problemas. A utilização de métodos da Semiótica Organizacional torna possível criar meios formais de analisar e definir metas de aprendizagem e meios avaliativos. A presente pesquisa utilizou-se das práticas e resultados de uma competição de robótica envolvendo alunos de ensino médio e técnico em fase inicial de aprendizagem de programação de computadores.

Palavras chaves: Robótica Pedagógica; Semiótica Organizacional; ensino de programação de computadores.

ABSTRACT

This paper discusses the use of Educational Robotics for teaching computer programming by using concepts of norms and affordances from Organizational Semiotics aiming to improve teaching practices analysis and learning assessments. Among the major challenges in teaching computer programming are the complexity of concepts and high level of abstraction necessary. Educational Robotics can provide many beneficial features for these problems. While Organizational Semiotics methods makes it possible to create formal means for analyzing and defining learning goals and evaluation methods. This research based on practices and results of a robotics competition involving high school and technician formation students in initial stage.

Keywords: Educational robotics; Organizational Semiotics; teaching computer programming.

I. INTRODUÇÃO

Algumas das grandes dificuldades no ensino

de Computação são justamente em disciplinas fundamentais que envolvem Lógica de Programação e Programação de Computadores, que muitas vezes representam o principal fator de insucesso de vários alunos na área. Essas disciplinas tem caráter introdutório e são apresentadas aos alunos logo no início do curso, e o processo de aprendizagem dos conceitos iniciais de programação se mostra complexo, abstrato e marcado pela presença constante de dificuldades para os iniciantes. A imperícia e desinteresse por essas disciplinas tornam-se o fator mais marcante nos índices de evasão de cursos de Computação e Informática.

Nesse cenário, se torna relevante buscar meios de lidar com a complexidade que envolve o processo de ensino-aprendizagem e ao mesmo tempo estimular o aluno. Como aponta Benitti (2011), entre as soluções possíveis para esse problema, a utilização da Robótica Pedagógica (RP) é umas das mais promissoras e mundialmente adotadas. Para teóricos na área de Educação, como Papert (1993), a utilização de robôs como artefatos educacionais, tem um grande potencial para prover um ambiente favorável ao aprendizado dentro da sala de aula. Benitti (2011) e Eguchi (2010) salientam ainda que a RP fornece uma vasta gama de possibilidades de práticas que podem atrair jovens estudantes, trazer uma abordagem pedagógica não tradicional e estimular a busca por soluções.

O objetivo deste artigo é discutir o uso da RP no ensino de programação de computadores através do conceito de normas e *affordances* da Semiótica Organizacional (SO). Por meio desses conceitos espera-se conceitualizar e explicar o uso da RP em uma visão ainda inexplorada. A partir de práticas reais, realizadas em uma competição de robótica com alunos de ensino médio, foram desenvolvidas práticas com robôs que contemplassem princípios fundamentais de programação e que fosse possível criar suas especificações com o auxílio de métodos da SO, para posterior avaliação da aprendizagem do aluno.

II. A ROBÓTICA PEDAGÓGICA

A robótica é uma ciência que cada vez mais ganha espaço na mídia e vem sendo aplicada em diversos ramos de atividades econômicas. Hoje é muito comum encontrar notícias de robôs auxiliando médicos em cirurgias delicadas ou em recuperação de pacientes; a exploração espacial só é viável devido ao uso de robôs; aplicação de monitoramento de clima, auxiliando na agricultura; dentre outras áreas as quais o uso da robótica se torna imprescindível.

Na área de educação, a robótica vem se mostrando forte aliada, podendo prover recursos tecnológicos eficazes e ambientes pedagógicos com grandes resultados. Segundo Silva (2009) a utilização da robótica no processo educativo possui três aspectos positivos:

- O robô, como elemento tecnológico, trás uma série de conceitos científicos, cujo princípios básicos de diversas áreas são dificilmente explorados com outros recursos;
- O uso de robôs cria novas formas de interação, e exigem uma nova maneira de gerar conhecimento, e;
- Envolve um processo motivação, colaboração, construção e reconstrução.

A utilização da RP não é algo novo, há décadas instituições de ensino e pesquisa a usam como ferramenta para criar um ambiente de ensino/aprendizagem mais atraente e completo. Segundo d'Abreu (2002) um ambiente no qual a RP é inserida propicia a montagem, automação e controle de dispositivos mecânicos. Ela também auxilia na interação entre aluno e professor junto a ferramentas tecnológicas, construindo conhecimento e caracterizando esse ambiente como um ambiente pedagógico que não existia a princípio. Trata-se de um processo interativo, conciliatório, entre o concreto e o abstrato, na resolução de um problema que envolve etapas como: concepção, implementação, construção, automação e controle de um

mecanismo.

Com a utilização da RP, além das questões de motivação e compreensão de conhecimento, torna-se um ambiente de trabalho colaborativo e atividades interpessoais que muitas vezes é impossível com instrumentos educacionais convencionais. A prática pode explorar trabalho em equipe, cooperação, criatividade, expressão escrita e oral, organização e habilidades relacionadas à execução de projetos. A RP trabalha competências além daquelas que os currículos escolares trabalham, oferecendo aos alunos experiências reais de áreas da informática, eletrônica, mecânica e design, através das possibilidades de conexão de peças mecânicas e de componentes eletrônicos, para realização de uma determinada tarefa ou resolução de um determinado problema ou desafio (PAPERT, 2008).

a) A RP no ensino de Programação de Computadores

A aprendizagem de programação de computadores é um processo difícil, sendo necessário um alto nível de abstração e compreensão de uma linguagem de programação. Como aponta Sleeman (1986), desde os anos 1980 há pesquisas que mostram que aprender a programar não é uma tarefa simples para novatos. Além disso, Gomes et al. (2008) acrescenta que que falta aos alunos iniciantes competências necessárias para resolução de problemas, assim como a inadequação dos métodos pedagógicos que são apresentados. O maior entrave se mostra no fato de um conteúdo com caráter dinâmico ser apresentado em de forma estática, o que não favorece o desenvolvimento dos alunos, por falta de compreensão ou desestímulo.

Em um processo tradicional de aprendizagem, uma vez que os alunos tem posse do conhecimento, o próximo passo é elaborar soluções de problemas. O processo resolutivo pode ser resumido nos seguintes passos (AMBRÓSIO et al, 2011):

- Compreensão do problema: leitura e compreensão do enunciado;
- Representação lógica: elaboração de um algoritmo que descreva uma sequência de comandos;
- Verificação e validação: manipulação de entradas e saídas, simulação, identificação de erros e correções.

Em um ambiente tradicional de ensino de programação é muito comum utilizar fluxogramas e pseudocódigos para demonstração de conceitos abstratos, o que colabora para aumentar as principais dificuldades que os alunos possuem. A RP proporciona melhores condições de ensino quando analisamos conceitos abstratos e aplicação de teoria. Com ela é possível observar o impacto real de instruções em um objeto concreto, e dessa maneira, associar os comandos abstratos aos movimentos e ações do mesmo (GOMES, HENRIQUES & MENDES, 2008).

A presença de um ambiente para Programação Visual (PV) baseada em uma interface de programação em blocos também apresenta uma vantagem relevante ao ensino de programação. A PV baseada em blocos, segundo Gomes, Henriques e Mendes (2008), aproxima os conceitos abstratos que envolvem a programação aos conceitos concretos de blocos, minimizando assim as dificuldades associadas aprendizagem, como a complexidade da sintaxe e conceitos abstratos.

Os conceitos que apoiam o trabalho e desenvolvimento de alunos durante as aulas podem ser potencializados com o uso da RP. Conforme Zili (2004) a utilização da RP contribui para o desenvolvimento de competências necessárias para alunos iniciantes em programação, como: desenvolvimento do raciocínio lógico; representação e comunicação; resolver problemas por meio de erros e acertos; aplicação das teorias formuladas em atividades práticas; e capacidade crítica.

Para aplicação da RP como instrumento didático se faz necessário associar quais os

conceitos mais significativos e fundamentais no ensino de programação com os recursos de programação oferecidos pelo robô e sua linguagem de programação. Segundo Brennan & Resnick (2012), há sete conceitos computacionais comuns entre as linguagens de programação procedurais (sequências, loops, eventos, paralelismo, condicionais, condicionais, operadores e dados).

b) O contexto da aplicação de RP como ferramenta didática

Há várias linhas de pesquisa que estudam metodologias de ensino que possam motivar e facilitar a aprendizagem de conceitos complexos. Seymour Papert foi um dos precursores nessa área com a teoria do Construcionismo. Segundo Valente (1993), o Construcionismo defende que o aprendizado deve vir de um processo de formulação de hipóteses, teste e avaliação do resultado. Essa teoria baseia-se no indivíduo criar significados por meio de experiências e ações de autoaprendizagem, sendo elemento ativo na construção do seu conhecimento (PAPERT, 2008).

Complementar à linha de pesquisa do Construcionismo, temos a metodologia *Problem Based Learning* (PBL), eficaz para o ensino de conceitos tecnológicos e propõe o aprender a aprender (SOARES & BORGES, 2011). Nessa metodologia o próprio aluno é responsável pela criação do aprendizado e o papel do professor é agir como um facilitador, selecionando os problemas a serem resolvidos. O objetivo da metodologia é proporcionar, por meio de um problema, um ambiente motivador no qual o aluno gera soluções com recursos disponíveis.

Segundo Huet et al. (2004), as aulas expositivas em salas de aulas comuns e sessões de práticas de programação em laboratórios são os métodos mais comuns de ensino em disciplinas introdutórias à Programação de Computadores, mas são comumente pouco efetivas. Alguns dos problemas apontados são: aula muitos

extensas com muito conteúdo para assimilar; grandes grupos de alunos inibindo assim a interação com o professor; e pouco tempo para aplicação prática. Huet et al. (2004) nota que tais pontos negativos nas práticas mais comuns de ensino tornam o aluno menos motivado e menos produtivo, ocasionando uma construção de conhecimento deficitária.

O uso da RP apresenta-se como uma alternativa para agregar as premissas propostas pelo aprendizado Construcionista e pela metodologia PBL, além de possibilitar a utilização de programação visual. A RP consiste em um processo de interação com dispositivos robóticos para motivar o aprendizado e desenvolver processos cognitivos (d'ABREU, 2007). Ela agrega um conjunto de recursos que visa o aprendizado científico e tecnológico integrado às demais áreas do conhecimento, utilizando-se de atividades como design, construção e programação de robôs (LOPES, 2008).

Atualmente há diversas pesquisas sobre RP, com diferentes abordagens. Eguchi (2011) define as três abordagens mais recorrentes como sendo:

- Abordagem curricular (*theme-based curriculum approach*): pesquisas que visam integrar diferentes áreas de conhecimento sobre um tópico de aprendizagem específico, sendo estudada principalmente através da investigação e comunicação;
- Abordagem baseada em projeto (*problem-based approach*): estudantes trabalhando em grupos e de maneira colaborativa para explorar problemas reais;
- Abordagem orientada a objetivos (*goal-oriented approach*): análise de resultados em competições com robôs, realizados em fora das escolas ou universidades.

Neste artigo, as práticas e pesquisas seguem a abordagem orientada a objetivos, utilizando-se de tarefas que pertencentes a

uma competição de robótica, que será descrita melhor na seção IV.

III. A SEMIÓTICA ORGANIZACIONAL

A Semiótica é uma teoria filosófica desenvolvida no século XIX pelo filósofo Charles S. Pierce, que trata dos estudos dos signos e sua representação, e o seu uso na cognição e comunicação (GUDWIN, 2002). A Semiótica tem sido explorada por diversas áreas do conhecimento, mais presente em ciências humanas, tornando-se uma filosofia científica. Notoriamente muitos trabalhos em Antropologia, Linguística, Filosofia, Educação, entre outras frentes do grupo de ciências humanas. Mas a Semiótica tem grande relevância também em Computação, como em Interface Humano-Computador (IHC) e Semiótica Computacional (BONACIN, 2004).

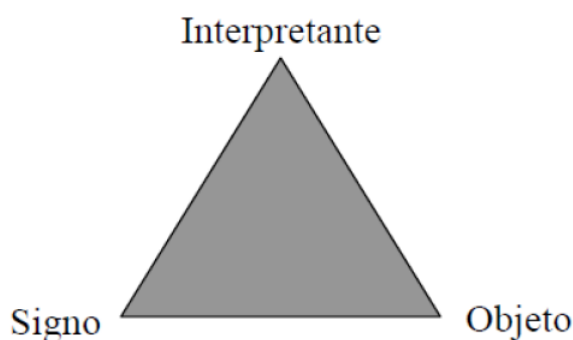
Semiótica Organizacional (SO) é um dos ramos da Semiótica e compreende uma organização como sendo um sistema de signos, e tem como objetivo estudar como a interpretação e comunicação deles pelas pessoas, de maneira individual ou em grupo (BONACIN, 2004). A SO utiliza conceitos e técnicas baseados na Semiótica Pierciana

(Pierce, 1990), que é baseada na lógica, na filosofia e na ciência da linguagem. É uma teoria que objetiva classificar e descrever todos os diferentes tipos de signos possíveis, considerando uma relação triádica do signo, definida por Pierce como Semiose.

Na teoria Peirceana (Peirce 1931-1958, cf 2.228) *“Um signo, ou representâmen, é aquilo que, sob certo aspecto ou modo, representa algo para alguém. Dirige-se a alguém, isto é, cria, na mente dessa pessoa, um signo equivalente, ou talvez um signo mais desenvolvido. Ao signo assim criado denomino interpretante do primeiro signo. O signo representa alguma coisa, seu objeto. Representa esse objeto não em todos os seus aspectos, mas com referência a um tipo de ideia que eu, por vezes denominei fundamento do representâmen (...)”*.

No processo cognitivo e perceptivo o signo criará na mente da pessoa envolvida um segundo signo equivalente a si mesmo, isto é, um signo mais desenvolvido, chamado interpretante. Tanto o signo quanto seu interpretante refere-se da mesma forma a um terceiro elemento, chamado de objeto (GUDWIN, 2002). Com o signo, o interpretante e o objeto a tríade de Pierce pode ser graficamente representada por meio do triângulo na Figura 1 a seguir:

Figura 1: Representação da tríade de Pierce.



Fonte: GUDWIN (2002)

Stamper (1973) propõe a divisão da Semiótica em seis níveis. Além dos níveis sintático, semântico e pragmático abordado

tradicionalmente na teoria semiótica, Stamper (1973) propõe o estudo dos signos em uma organização adicionando três outras

subdivisões: o nível físico, o nível empírico e o nível social. Na SO esses níveis são compreendidos como uma escada onde cada nível representa um “degrau” no *framework*:

- Nível físico: contendo aspectos físicos relacionados à mídia em que os signos são expressos;
- Nível empírico: contendo propriedades estatísticas dos signos quando estes são utilizados em mídias diferentes;
- Nível sintático: contendo as operações baseadas em regra entre signos de um sistema de signo;
- Nível semântico: mostrando a relação entre os signos e o mundo externo ao sistema de signos;
- Nível pragmático: a avaliação do signo com relação aos propósitos de seus usuários.
- Nível social: onde os efeitos do uso dos signos no relacionamento humano são estudados.

Liu (2000) descreve que os diferentes níveis do *framework* são analisados por meios de métodos específicos, permitindo uma modelagem sob diversos pontos de vista e detalhando diferentes características semióticas em relação ao uso de signos na organização. Para isso se faz necessário a utilização de uma ferramenta ou conjunto de métodos, como o MEASUR (*Method for Eliciting, Analysing and Specifying User Requirements*), orientados as normas e conceitos da SO para lidar principalmente com signos e seus significados (Semântica), intenções (Pragmática), e consequências sociais de seu uso (Social) (STAMPER, 1993; LIU, 2000).

Dentre os métodos que constituem o MEASUR temos o método de análise semântica (*Semantic Analysis Method* – SAM) e análise de normas (*Norm Analysis Method* - NAM). O método SAM tem como foco o agente e suas invariantes de comportamento. O NAM possibilita a especificação de padrões comportamentais

dos agentes em sociedade que participam do sistema organizacional, por meio da análise das regularidades comportamentais e sua especificação na forma de normas de comportamento.

Os conceitos básicos relacionados a Semiótica Organizacional adotados para os métodos SAM e NAM podem ser resumidos, segundo Liu (2000):

- Agente: define algo responsável por um comportamento. Nesse artigo pode-se tomar como exemplo “aluno”, “professor” ou mesmo uma instituição. É importante notar que esta definição de agente da SO difere da definição de agente artificial comumente utilizado em inteligência artificial. Aqui o que se identifica são responsabilidades sobre *affordances* ;
- *Affordance*: conceito introduzido por Gibson (1968) que pode ser utilizado para denominar a propriedade ou o comportamento de um objeto, elemento, sistema ou organismo combinada com as características de seu ambiente, e que permita a identificação de seu funcionamento ou ação dentro de um contexto. Em seu trabalho Gibson pretendia descrever a relação do “animal” com seu ambiente. O conceito de *affordance* tem como função nomear alguma coisa que se refere tanto ao ambiente quanto ao animal, e a complementaridade entre eles. Gibson exemplifica essa complementaridade com o formato das superfícies terrestres. Dependendo do tipo de superfície, o animal ou o homem tem um tipo diferente de comportamento ou ação: andar em uma trilha na floresta, deitar em um gramado, nadar no mar, etc. Neste trabalho podemos utilizar como exemplo “utilizar uma estrutura de seleção” (*affordance* do agente *aluno* possível via o artefato/outro *affordance ambiente de programação*) ou “movimentar-se para frente” (*affordance* do agente

aluno possível via o artefato/outro *affordance robô*).

O conceito de *affordance* é um elemento presente na Análise Semântica definida no MEASUR. O conceito de Gibson foi expandido por Stamper para incluir invariantes que são percebidas no mundo social. No conceito original, o enfoque era a percepção do mundo físico; entretanto a noção de *affordance* adotada por Stamper generaliza para incluir invariantes que nós percebemos no mundo social, incluindo conceitos abstratos como os relacionados com a programação.

a) A Análise Semântica

A Análise Semântica (SAM) é um método que determina as relações existentes entre agentes e *affordances*. Uma vez definido um problema ou cenário, é possível descrever uma visão dos agentes do domínio e seus *affordances*. Normalmente é utilizado um diagrama de ontologia, o qual representa as relações de agentes e *affordances* combinadas com outros conceitos, como dependência ontológica, parte-todo, generalização, determinantes, entre outros (LIU, 2000).

A identificação dos padrões dos agentes nos ajuda a eleger comportamentos que posteriormente se tornaram os *affordances*. Para este trabalho, a escolha dos potenciais *affordances* foi delimitada pelos componentes necessários para a aprendizagem de programação de computadores por parte dos alunos e os recursos que o robô deve fornecer.

b) A Análise de Normas

Como referência para avaliação de resultados será adotada uma perspectiva orientada a normas de aspectos pragmáticos, sociais e comportamentais, formando um modelo formal de avaliação cognitiva. Esse modelo incorpora diferentes tipos de normas em diferentes tipos de camadas semióticas. A análise utiliza o método NAM e a definição das normas segue a teoria baseada em normas criada por Stamper et al. (2000),

que inclui normas perceptuais, normas cognitivas, normas avaliativas, normas denotativas e normas comportamentais.

O conceito de normas define as responsabilidades dos agentes envolvidos em alguma tarefa, ou as condições sob as quais certas ações podem ou não ser realizadas por um agente. Normas correspondem no nível social à ideia de *affordance* no nível individual (LIU, 2000).

Como referência para avaliação de resultados será adotada uma perspectiva orientada a normas de aspectos pragmáticos, sociais e comportamentais, formando um modelo formal de avaliação cognitiva. Esse modelo incorpora diferentes tipos de normas em diferentes tipos de camadas semióticas. A análise utiliza o método NAM e a definição das normas segue a teoria baseada em normas criada por Stamper et al. (2000), que inclui normas perceptuais, normas cognitivas, normas avaliativas, normas denotativas e normas comportamentais. Neste trabalho cada um dos tipos de normas terá um objetivo específico no processo:

- **Normas perceptuais:** normas perceptuais ajudam a compreender como a pessoa (aluno) consegue identificar as diferentes ações executadas pelo robô e aos processos de programação envolvidos. O robô proporciona uma gama de reações que podem representar diferentes fenômenos envolvidos na programação.
- **Normas cognitivas:** a análise sob a ótica das normas cognitivas possibilita verificar aspectos de compreensão do aluno das ações programadas e as respostas dadas pelo robô. A tarefa de programar exige um alto nível de cognição, envolvendo fatores como atenção, percepção, raciocínio e linguagem para desenvolver soluções. Com uso de robôs, as respostas dadas podem ser mais representativas, auxiliando na interpretação e compreensão de conceitos. Por exemplo, o ato de

encontrar um obstáculo e o mesmo ser evitado pelo robô, demonstra a “realidade” por trás de uma estrutura condicional atuando, assim como uma ação repetitiva pode representar a ação de um laço de repetição;

- **Normas avaliativas:** as normas avaliativas demonstram as expectativas e respostas dadas pelos robôs em sua programação e seu contexto. Essas normas estão relacionadas a como a pessoa irá escolher as funções corretas em determinado contexto para atingir seu objetivo, desenvolvendo uma coerência pragmática na construção da solução;
- **Normas denotativas:** as normas denotativas serão importantes para a verificação da compreensão e escolha correta dos elementos sintáticos (regras dos componentes e funções de programação) e sua aplicação semântica (o uso correto da programação);
- **Normas comportamentais:** toda prática de programação exige uma sequência de ações para atingir um objetivo. Nesse trabalho as normas comportamentais definem quais serão os comportamentos dos alunos diante às adversidades encontradas no momento da resolução de um problema.

IX. PRÁTICAS PEDAGÓGICAS AVALIADAS ATRAVÉS DO USO DE NAM

a) Contextualização das práticas

Para aplicação da RP como instrumento didático se faz necessário associar quais os conceitos mais significativos e fundamentais no ensino de programação com os recursos de programação oferecidos pelo robô e sua linguagem de programação. Segundo Brennan & Resnick (2012), há sete conceitos computacionais comuns entre as linguagens de programação (sequências, loops, eventos, paralelismo, condicionais,

condicionais, operadores e dados). Para Forbellone e Eberspächer (2005), fundamentalmente a lógica de programação e a composição de um algoritmo se agrupam em: (1) tópicos preliminares ou fundamentais (tipos primitivos, variável, expressões aritméticas, expressões lógicas, comando de atribuição e comandos de entrada e saída); (2) estruturas de controle (estruturas de seleção e estruturas de repetição) e; (3) modularização de algoritmos (módulos). Nesse trabalho será adotado a como referência a composição definida por Forbellone e Eberspächer (2005) por refletir o melhor o currículo básico de disciplinas introdutórias de programação de computadores.

b) Exemplo de prática utilizando RP

As práticas utilizando RP devem englobar, além dos aspectos motivacionais anteriormente citados, a delimitação dos aspectos de mais fundamentais que norteiam o aprendizado de Programação.

Práticas que envolvem robôs móveis são executadas em uma arena, a qual é projetada com elementos mapeáveis, como linhas, obstáculo, áreas demarcadas, cores, etc. Uma vez desenvolvida a arena é fundamental disponibilizar uma descrição de qual é o objetivo a ser alcançado e o que cada elemento representa na arena.

A Figura 1 ilustra uma prática utilizando RP. Essa construção de arena foi utilizada na competição desenvolvida pelo projeto Forma-Engenharia¹ (CNPq), em parceria com a Universidade São Francisco com vinte alunos de primeiro e segundo anos do curso técnico de Informática Integrado ao Ensino Médio, na escola técnica ETEC Rosa Perrone Scavone, localizada na cidade de Itatiba, interior do estado de São Paulo, utilizando kits robóticos LEGO® Mindstorms em atividades relacionadas às práticas de engenharia e computação. Essa competição ocorreu em outubro de 2013, com duração de três dias. A competição seguiu os moldes de grandes competições, como a realizada pela Olimpíada Brasileira

de Robótica (OBR). Nesse formato são reunidas duplas de participantes que devem programar seu robô para executar tarefas em uma arena. Dentre essas tarefas estão desvios de obstáculos, reconhecimento de cores e áreas, seguir linhas demarcadas no piso, entre outras.

Para a análise de normas desse trabalho foi utilizado umas das provas aplicadas nessa

competição. A descrição dessa prova é: “O robô deverá iniciar a prova no ponto inicial B e deverá percorrer a linha preta até a faixa azul que se encontra no ponto J. A partir do ponto J, o robô deve escolher e seguir dentre as faixas que se originam do ponto J, a faixa que leva menos tempo para chegar na faixa vermelha no ponto E. Ao alcançar a faixa vermelha no ponto E, o robô deverá retornar até o ponto inicial B”.

Figura 2: Exemplo de prática em arena



Fonte: Acervo dos autores

c) A aplicação da Análise de Normas

Utilizando o *framework* semiótico como guia para elicitación e análise de problemas e soluções das práticas usando RP, podemos organizar os aspectos relacionados ao domínio dos objetivos didáticos e organizar do nível físico ao nível social na escada semiótica. Abaixo são apresentadas questões que orientam uma análise nesse domínio:

- Nível físico: Quais são os recursos de hardware disponíveis nos robôs? Quais as ações podem ser realizadas utilizando esses recursos? Qual é o hardware necessário para poder trabalhar?
- Nível empírico: Qual é o meio de comunicação do aluno com o robô? Qual é a banda de comunicação necessária?
- Nível sintático: Qual é a sintaxe da *linguagem* que os robôs oferecem? Como ocorrerá essa comunicação entre aluno e robô (programação)? Quais são as tecnologias e técnicas envolvidas?
- Nível semântico: Quais os significados atribuídos através dessa comunicação? Há compreensão do que é programado e a ação correspondente?
- Nível pragmático: O uso dos robôs apoiam corretamente as práticas propostas aos alunos? Como o professor poderá utilizar a RP nessas práticas? O aluno conseguirá distinguir e utilizar todos os recursos das ferramentas (programação e robôs) de acordo com suas intenções? O uso da RP criou um ambiente eficiente de aprendizagem? Os alunos conseguiram abstrair

conceitos de programação e utilizá-los nas tarefas?

- Nível social: Qual a consequência da RP na formação profissional do aluno? Qual o desempenho futuro do aluno na sua atuação profissional?

Para a especificação das normas, como as relações dos agentes com seu *affordances*, se faz necessário uma especificação formal, na qual apresente o domínio envolvido, as responsabilidades e os comportamentos de todos os elementos envolvidos. Essa especificação corresponde a uma norma específica e é organizado pela seguinte estrutura:

- **Agentes:** determinada quais são os agentes envolvidos naquele cenário descrito pela norma;
- **Affordances:** aponta quais *affordances* requeridos para o cumprimento da norma;
- **Detalhamento da norma:** descrição através de uma linguagem natural, definindo regras e ações.

A Tabela 1 mostra um exemplo de como o quadro pode ser organizado. Como exemplo, temos um cenário relacionado ao um desvio de obstáculo em uma arena que deve ser realizado pelo robô.

Tabela 1: Quadro de descrição de normas

Norma	Desviar de um obstáculo na arena
Agentes	aluno
<i>Affordances</i>	robô (<i>affordance</i> do fabricante) estrutura de seleção, expressão lógica (aluno por meio do mecanismo de programação) identificação de obstáculo (dependente do robô e do aluno)
Detalhamento da norma	sempre que robô identificar um obstáculo na arena se é preciso desviar desse obstáculo então , o aluno é obrigado a utilizar uma estrutura de seleção para determinar o desvio

No quadro apresentado na Tabela 1 nota-se uma descrição de quais são os *affordances* e os agentes em um cenário definido por uma norma. Nesse exemplo a identificação do obstáculo só é possível porque existe o robô (*affordance*) e um aluno que o programou (agente), utilizando-se de conceitos de programação como estrutura de seleção e expressões lógicas através de um ambiente de programação (todos também definidos como *affordances*).

No item “Detalhamento da norma”, podemos descrever as normas incluindo aspectos ligados à autoridade e delegação de responsabilidade dos agentes envolvidos através do uso de uma linguagem natural ou Lógica Deontica. Segundo Liu (2000), essa descrição pode seguir formato utilizado para

a especificação de normas de comportamento. Tal formato se apresenta como:

**<Norma>::= sempre <condição> se
<estado> então <agente> é <D> fazer
<ação>**

onde <Norma>, corresponde ao nome da norma. O campo <condição> especifica uma certa condição que a norma é aplicada em um determinado estado <estado>. O campo <agente> especifica qual agente tem a obrigação, permissão ou proibição, de acordo com operador deontico <D>, de realizar uma determinada ação <ação> (BONACIN, 2004). O operador deontico <D> pode ser classificado como obrigatório, permitida e proibida.

A partir do uso do quadro e das descrições de normas é possível criar um questionário com questões que representem quais são os pontos a serem avaliados na prática e as relações dos alunos junto a essas normas. Exemplos de questões relacionadas ao exemplo anteriormente citado seriam: “É possível identificar qual é a solução e os recursos necessários para que o robô desvie do obstáculo?” e “Se encontrada a solução, quais foram os recursos de programação utilizados?”.

CONSIDERAÇÕES FINAIS

A RP para o ensino de programação de computadores é um recurso pedagógico com muito potencial, tanto na questão motivacional para os alunos, quanto na diversidade de ferramentas e suporte que oferece aos professores. Entretanto, sua aplicação em sala de aula não é uma tarefa trivial e sua aplicação deve ser amparada por métodos formais que possam conduzir a aprendizagem e fornecer instrumentos avaliativos eficazes. A análise semântica e análise de normas da SO trazem recursos metodológicos que podem favorecer o desenvolvimento e condução de práticas com robôs, pois possibilita ao professor criar uma análise do cenário, conseguindo previamente definir quais são as responsabilidades e comportamento esperado do aluno diante de uma prática pedagógica.

Os métodos discutidos neste trabalho têm como objetivo a formatação de práticas e a avaliação de resultados através de uma perspectiva orientada a normas de aspectos pragmáticos, sociais e comportamentais, formando um modelo formal de avaliação cognitiva no ensino de programação de computadores. A definição de normas apresenta-se como uma maneira a qual o professor pode definir quais são as tarefas esperadas em uma determinada prática, consequentemente elencando quais são as responsabilidades do aluno e outros agentes, e quais são os recursos que serão necessários para a execução (*affordances*).

O mapeamento de todas as possibilidades e cenários para o desenvolvimento de práticas efetivas utilizando RP ainda necessita especificações mais detalhadas. Como trabalho futuros é esperado um processo mais amplo com métodos da SO, contemplando apuração mais detalhada do escopo de problema (cenários de práticas e seus objetivos) e especificação de diagramas de ontologia para descrição de contexto de normas.

¹ Chamada pública do Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq), com a referência “CNPQ/VALE S.A. nº 05/2012 - FORMA-ENGENHARIA”, que teve como objetivo selecionar propostas para apoio financeiro a projetos para estimular a formação de engenheiros no Brasil, combater a evasão e despertar o interesse vocacional dos alunos de ensino médio pela profissão de engenheiro e pela pesquisa científica. Site do projeto: <http://roboticaeducacional.webs.com/>

REFERÊNCIAS

- AMBRÓSIO, A., ALMEIDA, L., MACEDO, J., SANTOS, A., FRANCO, A. (2011). Programação de computadores: compreender as dificuldades de aprendizagem dos alunos. *Revista Galego-Portuguesa de Psicoloxía e Educación*, 19 (1). 185-197.
- BENITTI, F. B. V. (2011). Exploring the educational potential of robotics in schools: A systematic review. *Computers & Education*, 58(3), 978-988.
- BONACIN, R. (2004). Um Modelo de Desenvolvimento de Sistemas para Suporte a Cooperação Fundamentado em Design Participativo e Semiótica Organizacional. Tese (Doutorado em Ciência da Computação) – Universidade Estadual de Campinas.
- BRENNAN, K.; RESNICK, M. (2012). New frameworks for studying and assessing the development of computational thinking. AERA 2012. Disponível em: http://web.media.mit.edu/~kbrennan/files/Brennan_Resnick_AERA2012_CT.pdf . Acesso em: 12 Out. 2012.
- D'ABREU, J. V. V. (2007). Ambiente de aprendizagem baseado no uso de dispositivos robóticos automatizados. 2007. In:

- Aprendizagem na era das tecnologias digitais. Ed. Valente, J.A., Mazzone, J., Baranauskas, M. C. C., Cortez/FAPESP, São Paulo.
- D'ABREU, J.V.V. (2002). Integração de dispositivos mecatrônicos para ensino-aprendizagem de conceitos na área de automação. 2002. Tese (Doutorado) - Faculdade de Engenharia Mecânica, Universidade Estadual de Campinas.
- EGUCHI, A. (2010). What is educational robotics? Theories behind it and practical implementation. In D. Gibson & B. Dodge (eds.), *Proceedings of Society for Information Technology & Teacher Education International Conference 2010* (pp. 4006-4014). Chesapeake,VA:AAACE.
- FORBELLONE, A. L. V., EBERPÄCHER, H. F. (2005). *Lógica de Programação: a construção de algoritmos e estruturas de dados*. 3ª edição. São Paulo: Pearson Prentice Hall, 2005.
- GIBSON, J. J. (1986). *The ecological approach to visual perception*. Hillsdale: Lawrence Erlbaum Associates, 1986.
- GOMES, A.; HENRIQUES, J.; MENDES, A. J. (2008). Uma proposta para ajudar alunos com dificuldades na aprendizagem inicial de programação de computadores. *Educação, Formação & Tecnologias*, ed. 1, 2008, pág. 93 - 103.
- GUDWIN, R. R. (2002). *Semiônica: Uma Proposta de Contribuição à Semiótica Computacional*, Tese (obtenção do Título de Professor Livre-Docente) Faculdade de Engenharia Elétrica e de Computação, Universidade Estadual de Campinas, Campinas. 2002.
- HUET, I.; PACHECO, O.R.; TAVARES, J.; WEIR, G. (2004). New Challenges in Teaching Introductory Programming Courses: a Case Study. *Frontiers in Education Conference*, 34th ASEE/IEEE 2004.
- LIU, K. (2000). *Semiotics in Information Systems Engineering*. Cambridge University Press. Cambridge.
- LOPES, D. Q. (2008). A exploração de modelos e os níveis de abstração nas construções criativas com robótica educacional. 2008. 326f. Tese (Doutor em Informática na Educação) - Universidade Federal do Rio Grande do Sul-URGS, Porto Alegre, 2008.
- PAPERT, S. (1993). *Mindstorms: Children, computers, and powerful ideas*. 2ª ed. New York, NY: Basic Books
- PAPERT, S. (2008). *A máquina das crianças: repensando a escola na era da informática*. Edição Revisada. Porto Alegre: Artmed, 2008. 220 p.
- PIERCE, C.S. (1990). *Semiótica – Coleção Estudos*. 2ª ed. Editora Perspectiva, 1990. Tradução parcial de: *The Collected Papers of Charles Sanders Pierce em vários volumes*, edições e datas diversas.
- SILVA, A. F. (2009) *RoboEduc: Uma Metodologia de Aprendizado com Robótica Educacional*. 2009. Tese (Doutorado em Engenharia Elétrica e de Computação) - Universidade Federal do Rio Grande do Norte.
- SLEEMAN, D. (1986) "The challenges of teaching computer programming", In *Communication*. ACM Volume 29, Issue 9, pages 840–841.
- SOARES, R. F.; BORGES, M. A. F. (2001). *Robótica: aprendizado em informática de forma lúdica*. WEI – XXXI Congresso da Sociedade Brasileira de Computação, Natal-RN: SBC, 2011. Disponível em: http://www.dimap.ufrn.br/csbc2011/anais/eventos/contents/WEI/Wei_Secao_1_Artigo_4_Soares.pdf. Acesso em: 03 de outubro de 2012.
- SAMPER, R. K. (1973) *Information in Business and Administrative Systems*, John Wiley & Sons (ed.).
- STAMPER, R.K. (1993) . Social Norms in requirements analysis – an outline of MEASUR, in Jirotko, M., Goguen, J. and Bickerton, M. (eds.), *Requirements Engineering, Technical and Social Aspects*. Academic Press, New York.
- VALENTE, J.A. (1993). *Computadores e Conhecimento: repensando a educação*. Campinas: Gráfica central da Unicamp, Campinas-SP, Brasil, 1993. p. 24-44: Por quê o computador na educação.
- ZILLI, S. R. (2004). *A Robótica Educacional no Ensino Fundamental: Perspectivas e Práticas*. Dissertação de Mestrado – Florianópolis: UFSC, 2004.

ANEXO III – ARTIGO PUBLICADO NO SBIE

CONTRATO DE CESSÃO DE DIREITOS AUTORAIS

DAS PARTES

CESSIONÁRIO: **Sociedade Brasileira de Computação**, doravante denominada SBC, inscrita no CNPJ sob o no. 29.532.264/0001-78, Inscrição Estadual isenta e CCM sob o nº 18115128, com sede à Av. Bento Gonçalves, 9500 – Setor 4 – Sala 116 – Prédio 43424 –Agronomia – CEP 91501-970 – Porto Alegre – RS, neste ato representada por Sean Wolfgang Matsui Siqueira, Coordenador do comitê de programa do evento 25º Simpósio Brasileiro de Informática na Educação (SBIE 2014), 2014. **CEDENTE(S):** Humberto A. P. Zanetti, professor, e Rodrigo Bonacin, professor, doravante denominado(s) autor(es).

DO OBJETO

TÍTULO DO ARTIGO (doravante denominado “a obra”): **Uma Metodologia Baseada em Semiótica para Elaboração e Análise de Práticas de Ensino de Programação com Robótica Pedagógica**

NOME DA PUBLICAÇÃO: **Anais do XXV Simpósio Brasileiro de Informática na Educação (Anais do SBIE 2014)**

O(s) autor(es) da obra transfere(m) os direitos autorais para a Sociedade Brasileira de Computação (SBC). O(s) autor(es) representa(m) e garante(m) que a obra é original e de sua (vossa) autoria, com a exceção de algumas passagens de texto, figuras e dados cuja a fonte esteja claramente referenciada, e acompanhadas da permissão de reprodução do material emitida pelos detentores dos direitos autorais quando necessário. O(s) autor(es) afirma(m) que ele(s) ou ela(s) têm o poder e autoridade para executar esta tarefas.

A SBC reconhece e retém os direitos descritos nos itens 1 e 2 e permite ao(s) autor(es) a utilização do material segundo os itens 3 e 4.

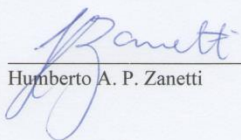
1. O(s) autor(es) retém todos os direitos proprietários sobre qualquer processo, procedimento ou artigo descrito na obra.
2. Em todos os casos não cobertos pelos itens 3 e 4, o(s) autor(es) devem solicitar permissão para reprodução da obra, incluindo figuras e tabelas, diretamente a SBC.
3. O(s) autor(es) podem reproduzir ou autorizar terceiros a reprodução de parte do material extraído desta obra, material extraído de forma *verbatim* ou derivados para o uso do(s) autor(es), uma vez que a fonte e os direitos da SBC sejam devidamente referenciados. As cópias não devem ser utilizadas de nenhum modo que implique o endosso da SBC. Cópias da obra não podem ser colocadas à venda sem a autorização expressa da SBC.
4. O(s) autor(es) podem distribuir de forma limitada todo ou partes da obra antes de sua publicação, desde de que informem a SBC da natureza da utilização, extensão e propósito da distribuição.

DO FORO

Para as questões resultantes deste contrato, será competente o Foro da Cidade de Porto Alegre - RS, renunciando as partes contratantes, desde logo, a qualquer outro, seja qual for a sua sede.

Porto Alegre, 12 de setembro de 2014.

Assinatura Autorizada
Sociedade Brasileira de Computação



Humberto A. P. Zanetti

Empregador para o qual a obra foi criada (se aplicável)

AUTORIA CONJUNTA

Uma Metodologia Baseada em Semiótica para Elaboração e Análise de Práticas de Ensino de Programação com Robótica Pedagógica

Humberto A. P. Zanetti¹, Rodrigo Bonacin^{1,2}

¹Faculdade Campo Limpo Paulista – FACCAMP – Campo Limpo Paulista – SP – Brasil

²Centro de Tecnologia da Informação Renato Archer (CTI) – Campinas – SP – Brasil

hzanetti@gmail.com, rodrigo.bonacin@cti.gov.br

Abstract. *This paper presents a methodology for conception, elaboration and analysis of pedagogical practices for teaching computer programming using Educational Robotics. This methodology uses Organizational Semiotics concepts and methods aiming to provide a systematic process to analyse, from the semiotics' point of view, the development of programming skills on practices integrated to disciplines of technological and higher education courses. The paper also illustrates the application of the process in a practice with six students of a higher education technical school.*

Resumo. *Este artigo apresenta uma metodologia para a concepção, elaboração e análise de práticas pedagógicas para ensino de programação de computadores por meio de Robótica Pedagógica. Essa metodologia utiliza métodos e conceitos provenientes da Semiótica Organizacional com o objetivo de constituir um processo sistemático para analisar, do ponto de vista da semiótica, o desenvolvimento de habilidades de programação em práticas integradas a disciplinas de cursos superiores em tecnologia e cursos técnicos do ensino médio. O artigo também ilustra a aplicação do processo em uma prática com seis alunos de uma escola técnica.*

1. Introdução

Grande parte dos alunos ingressantes em cursos de Computação tem contato com programação de computadores pela primeira vez nas disciplinas introdutórias nos primeiros semestres. Infelizmente, como aponta o trabalho de Gomes, Henriques e Mendes (2008), essas disciplinas são responsáveis por altos índices de evasão e reprovações, que promovem muitas vezes a desmotivação por parte do aluno. Vale destacar que começar a ensinar detalhes sintáticos antes que os alunos percebam qual são as finalidades e utilidades dos conceitos de programação pode potencializar essa desmotivação.

O processo de aprendizagem dos conceitos iniciais de programação é complexo e, muitas vezes, conforme destaca o trabalho de Gomes et al. (2008), necessita de um nível de abstração que não está presente na maioria dos alunos iniciantes. Várias pesquisas apontam soluções que visam colaborar para a criação de um ambiente mais diversificado e motivador para o aluno. D'Abreu (2002) e Benitti (2012) indicam que entre as soluções e ferramentas possíveis destaca-se a Robótica Pedagógica (RP). A RP

apresenta um vasto conjunto de recursos e possibilidades para o desenvolvimento de práticas pedagógicas mais estimulantes que as tradicionais execuções simuladas em quadro negro. No aspecto teórico da Educação, Papert (1993) aponta que a utilização de robôs como artefatos educacionais tem um grande potencial para prover um ambiente favorável ao aprendizado dentro da sala de aula.

Entretanto, conforme destacado por Benitti (2012), ainda é preciso um método sistemático de elaboração e avaliação de práticas pedagógicas utilizando robótica. Em um currículo de cursos técnicos ou de graduação a construção dessas práticas devem refletir o conteúdo programático envolvido, bem como os conceitos fundamentais que devem ser aprendidos. Consequentemente, há a necessidade de criar meios avaliativos que sejam efetivos e que colabore para a apresentação de resultados fidedignos.

Neste contexto, o artigo propõe uma metodologia baseada em semiótica para concepção, elaboração e análise de práticas. Como parte desta metodologia, é proposto um processo que visa desenvolver meios para pesquisadores analisar os efeitos destas práticas com alunos. O objetivo é apresentar uma proposta e discutir como a semiótica pode auxiliar na identificação de benefícios e desafios do uso da RP no ensino de programação. Para tanto, são empregados métodos da Semiótica Organizacional (SO) em conjunto com a RP. O artigo ainda apresenta um estudo com seis alunos e respectivas respostas a questionamentos sobre o uso da RP no processo proposto.

O artigo apresenta na Seção 2 uma breve descrição sobre a RP e sua aplicação no ensino de programação de computadores; na Seção 3 são abordados métodos da SO; na Seção 4 é apresentada a metodologia; na Seção 5 é destacada a aplicação em uma prática pedagógica; e por fim, na Seção 6 são apresentadas as considerações finais.

2. A Robótica Pedagógica no ensino de Programação de Computadores

A RP já está presente em diversas instituições de ensino e pesquisa, sendo usada como artifício para criar um ambiente de ensino mais atraente e diversificado. Segundo d'Abreu (2002), a RP propicia a montagem, automação e controle de dispositivos mecânicos. Ela também auxilia na interação entre aluno e professor com as ferramentas tecnológicas, construindo conhecimento e possibilitando um ambiente pedagógico que não existia a princípio. Assim, com a RP os alunos podem trabalhar a conexão de peças mecânicas e de componentes, além de estimular o desenvolvimento de habilidades de liderança e trabalho em equipe [Papert 1994].

A aprendizagem de programação de computadores é um processo difícil, sendo necessário um alto nível de abstração e compreensão de linguagens com sintaxe complexa. Gomes, Henriques e Mendes (2008) salientam que falta aos alunos iniciantes competências necessárias para resolução dos problemas. Um dos maiores entraves está no fato de um conteúdo com caráter dinâmico ser apresentado de forma estática.

Métodos tradicionais de ensino de programação utilizam fluxogramas e pseudocódigos para demonstração de conceitos abstratos. A adoção de RP traz novos recursos de ensino que ajudam a apresentar esses conceitos. Com a RP é possível observar o impacto de instruções em um objeto concreto e associar os comandos abstratos aos movimentos e ações do mesmo [Gomes, Henriques e Mendes 2008]. Segundo Zilli (2004) a utilização da RP contribui para o desenvolvimento de

competências necessárias para alunos iniciantes em programação, como: raciocínio lógico, representação e comunicação, resolver problemas por meio de erros e acertos, aplicação das teorias formuladas em atividades práticas e capacidade crítica.

A RP pode ser utilizada em conjunto com o Construcionismo, teoria desenvolvida por Seymour Papert que baseia-se no indivíduo criar significados por meio de experiências e ações de autoaprendizagem, sendo elemento ativo na construção do seu conhecimento [Papert 1994]. Segundo Valente (1993), o Construcionismo promove o aprendizado através de um processo de formulação de hipóteses, testes e avaliação.

Com um foco mais prático e complementar à teoria do Construcionismo, temos a metodologia *Problem Based Learning* (PBL), que é eficaz para o ensino de conceitos tecnológicos e que propõe o “aprender a aprender”. Segundo Soares e Borges (2011), nessa metodologia o próprio aluno é responsável pela criação do aprendizado, e o principal papel do professor é agir como um intermediador ou facilitador.

3. A Semiótica Organizacional aplicada ao Ambiente de Ensino

A SO utiliza conceitos e técnicas baseadas na Semiótica para analisar organizações e o comportamento organizado. De acordo com Liu (2000), no contexto da SO entende-se como organização qualquer domínio que demande comportamento organizado, tal como um ambiente de ensino. Para Stamper (1973) a comunicação pode ser dividida em seis níveis que formam um framework semiótico: social, pragmático, semântico, sintático, empírico e físico.

O MEASUR (*Methods for Eliciting, Analysing and Specifying User Requirements*) é um conjunto de métodos orientados às normas para lidar com signos e seus significados, intenções e consequências sociais. Com o PAM (*Problem Articulation Methods*) é possível identificar os objetos a serem trabalhados no ambiente de ensino nos diversos níveis do *framework* semiótico. Enquanto o método SAM (*Semantic Analysis Method*) tem como foco os agentes e suas invariantes de comportamento, como por exemplo, invariantes de comportamentos do professor e aluno pertinentes ao contexto de aplicação da RP. O NAM (*Norm Analysis Method*) possibilita a especificação de padrões comportamentais dos agentes em sociedade que participam do sistema organizacional e a análise é realizada por meio da especificação na forma de normas.

Os conceitos fundamentais do MEASUR incluem agentes, *affordances* e normas. Para Liu (2000), agente é algo responsável por *affordances* e pode-se tomar como exemplo “aluno”, “professor” ou mesmo a instituição de ensino. O conceito de *affordance*, introduzido por Gibson (1968), pode ser utilizado para denominar a propriedade ou o comportamento de um objeto, sistema ou organismo combinada com as características de seu ambiente. O conceito de Gibson foi expandido por Stamper (1973) para incluir invariantes que são percebidas no mundo social. Uma “estrutura de seleção” é um exemplo de um *affordance* do agente “aluno”, que só é possível pelas habilidades do aluno com o “ambiente de programação” (que é outro *affordance*).

O conceito de normas define as responsabilidades dos agentes envolvidos em alguma tarefa, ou as condições sob as quais certas ações podem ou não ser realizadas.

Normas correspondem no nível social à ideia de *affordance* no nível individual. Esse conceito baseia-se na teoria de normas criada por Stamper e outros (2000), que inclui normas: perceptuais, cognitivas, avaliativas, denotativas e comportamentais. No processador proposto, estas normas resultam em requisitos chaves de aprendizagem a serem explorados nas práticas com alunos, assim como elementos¹ a serem analisados após as práticas. Demais ferramentas utilizadas (*e.g.*, Cebola Semiótica, lógica deôntica, e *framework* semiótico) são detalhadas em Liu (2000).

4. A Metodologia e Processo Proposto

Estudos preliminares [Zanetti e Bonacin 2014] exploraram a aplicação do *framework* semiótico e do NAM para a descrição de atividades de RP. Neste artigo, é proposto sistematizar o emprego do PAM, SAM e NAM no processo PDAP-RP (Processo de Desenvolvimento e Análise de Práticas com uso de Robótica Pedagógica). A Figura 1 apresenta uma visão geral do PDAP-RP, com linhas tracejadas que indicam fases distintas conforme descritas a seguir:

1. *Identificação do cenário e problemas* – Cenário são os elementos e ambientes necessários para a prática e problemas são as dificuldades na obtenção de um determinado objetivo. Nessa fase, o professor-pesquisador faz o levantamento inicial dos recursos existentes e quais são os objetivos. Ela é composta por três atividades: (1.1) *Delimitação do Cenário/Problema*: descrição do cenário e levantamento dos principais problemas; (1.2) *Análise de Stakeholders*: analisa os *stakeholders* (partes interessadas) envolvidos através da “cebola semiótica” [Liu, 2000]; e (1.3) *Uso de Framework Semiótico*: como guia para elicitação e análise de problemas e soluções das práticas usando RP. A fase visa classificar e organizar os problemas e soluções relacionadas ao domínio dos objetivos didáticos do nível físico ao nível social.
2. *Definição e modelagem de agentes, comportamentos e dependências ontológicas* – É realizada a análise e modelagem dos conceitos relacionados ao SAM. Em uma prática com RP os agentes (*e.g.*, alunos, professor) possuem *affordances* relacionados aos recursos de programação (*e.g.*, estruturas de repetição, seleção) e a interação com o robô (*e.g.*, reconhecimento de obstáculo, movimento). Esta fase inclui as seguintes atividades (inspiradas em Liu (2000)): (2.1) *Definição de Agentes e Affordances*: onde são listados candidatos a agentes e *affordances* a partir das descrições do problema; (2.2) *Agrupamento de Candidatos*: análise das unidades semânticas; (2.3) *Detalhamento de comportamentos e ações*: mapeamento as dependências existentes entre as unidades semânticas; e (2.4) *Diagramação de Ontologias*: criação de um modelo de representação gráfica para conceitos como: agentes (círculos), *affordances* (retângulos) e dependências (linhas lidas da esquerda para direita).
3. *Análise e modelagem de normas* – Primeiramente são modeladas as normas comportamentais que irão compor os quadros de descrição das tarefas a serem realizadas pelos alunos. As normas referem-se essencialmente às relações dos agentes com seus *affordances*. Grande parte dessas relações dizem respeito ao aluno, ao robô e à lógica de programação, incluindo aspectos relacionados aos recursos de

¹ “Elementos” no contexto de artigo são entendidos como objetos necessários podendo ser físicos (*e.g.*, arena, obstáculos) e conceituais (*e.g.*, objetivos, problemas).

hardware disponível, à comunicação e à linguagem usada entre aluno e robô. Esta fase inclui as seguintes atividades: (3.1) *Análise de Normas*: é a identificação, modelagem e análise das normas que guiam a prática pedagógica; (3.2) *Elaboração de quadro com normas comportamentais*: onde são produzidos quadros [Zanetti e Bonacin 2014] contendo os agentes envolvidos, os *affordances* e descrição em Lógica Deontica; (3.3) *Elaboração de quadro de respostas (demais normas)*: Uma vez definidas as normas comportamentais, é criado um quadro com os demais tipos de normas com os resumos das respostas dos alunos e suas impressões quanto aos aspectos perceptuais, cognitivos, denotativos e avaliativos. O objetivo é identificar se todas as ações foram realizadas, percebidas e compreendidas corretamente.

4. *Aplicação de práticas* – Nessa fase é aplicada a prática pedagógica em si. Para tanto, é definido qual o ambiente de aplicação. Essa fase engloba a avaliação e verificação dos resultados, que podem levar à revisão de outros pontos da metodologia e modelos em um processo iterativo. Esta fase inclui as atividades: (4.1) *Definição das práticas usando RP*: onde é definido o ambiente de aplicação da prática e quais elementos que o robô deverá possuir (*e.g.*, sensores e estrutura física); e (4.2) *Avaliação dos resultados*: o professor-pesquisador observa e determina, através dos pontos definidos nos quadros de normas, se as ações desenvolvidas pelos alunos e robô atenderam ao que era esperado. Questionários e entrevistas sintetizados através de quadros de respostas complementam a avaliação.

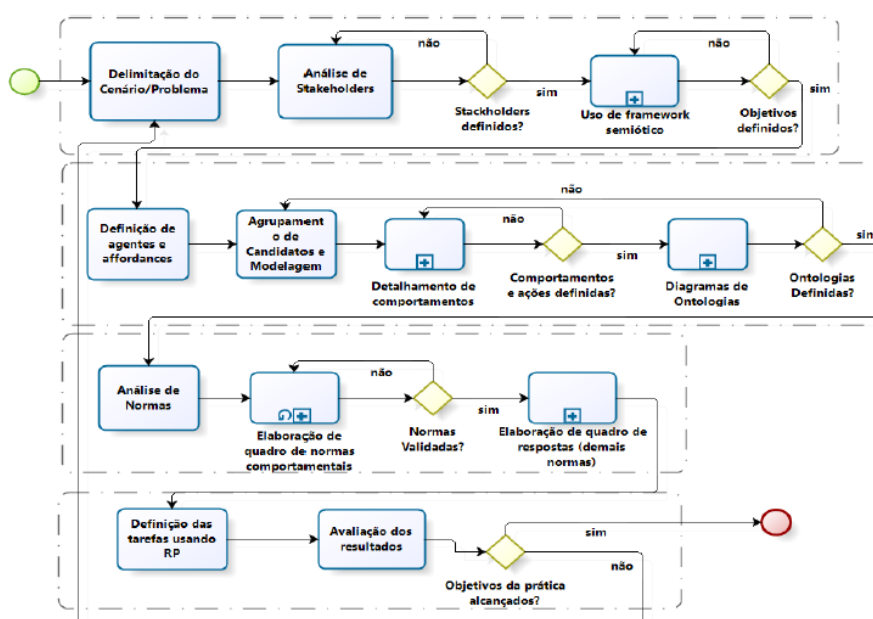


Figura 1. Visão geral do PDAP-RP

5. Aplicação e Resultados

A prática realizada teve como objetivo trabalhar dois conceitos de programação: (1) a estrutura de seleção (*if-else*) e a estrutura de repetição (*while*). Conforme ilustra a Figura 2, ela foi executada em uma arena na cor branca, sendo demarcada uma faixa na cor preta que indicava o percurso que o robô devia seguir. O robô contava com um sensor identificador de cor, para verificar se o mesmo se encontra sobre a faixa. Para coordenar

a movimentação do robô sobre o percurso foi necessário desenvolver uma programação com a seguinte verificação: “se encontrar branco, gire 30° graus para a esquerda até encontrar a faixa preta, caso contrário volte à posição inicial e gire 30° à direita até encontrar a faixa preta”. A subseção 5.1 apresenta a aplicação do método durante esta prática e a subseção 5.2 faz uma breve análise dos resultados obtidos.



Figura 2. Visão da arena e preparação dos alunos

5.1. Aplicação do Método PDAP-RP

Para a realização da prática foram selecionados seis alunos (duas alunas e quatro alunos) do curso Técnico em Informática integrado ao Ensino Médio, da Escola Técnica Rosa Perrone Scavone, localizada na cidade de Itatiba-SP. Todos os alunos têm idade entre quinze e dezessete anos, atualmente estão no 2º ano do curso, já cursaram disciplinas básicas de Lógica de Programação e Programação de Computadores e possuíam contato prévio com robótica. O grupo é formado por alunos em processo de aprendizagem de programação e ainda são afetados pelas dificuldades em disciplinas relacionadas à programação de computadores. A prática foi realizada em um dos laboratórios com duração de aproximadamente três horas.

A Tabela 1 apresenta a aplicação da Fase 1 do PDAP-RP, sendo composta pela delimitação do cenário/problema, análise dos principais *stackholders* e o uso de *framework* semiótico. Como mostra a Figura 3, na “Cebola Semiótica” (lado esquerdo) os problemas relacionados à prática e *stakeholders* estão categorizados de acordo com os níveis técnico (e.g., manuseio e comunicação), formal (e.g., procedimentos de ensino) e informal (e.g., cultura do ambiente). A Figura 3 também exhibe o uso do *framework* semiótico (lado direito), onde aspectos da plataforma tecnológica (e.g., recursos de hardware e software) e do sistema de informação humano (e.g., atuação profissional do aluno) estão categorizados nos seis níveis do *framework*.

Tabela 1. Atividades e resultados da Fase 1 do PDAP-RP

Fase 1: Identificação do cenário e problemas	
1.1. Delimitação do Cenário/Problema	Descrição do cenário listando os principais conceitos a serem trabalhados e a descrição dos objetivos a serem alcançados de maneira ampla.
1.2. Análise de Stackholders	Lista com a definição dos papéis dos <i>stackholders</i> professor e alunos. Nesse contexto, além de aplicar a prática, o professor tem o papel de facilitador e orientador. Os alunos têm como objetivo a resolução do problema proposto, identificação da solução e execução através da programação do robô. A Figura 3 (esquerda) apresenta a Cebola Semiótica utilizada na análise.
1.3. Uso de Framework Semiótico	Problemas e soluções classificadas de acordo com os níveis do <i>framework</i> . Como exemplo, no nível físico, foram apontados quais os recursos necessários ao robô (componentes estruturais, motores e sensores); e no pragmático, a listagem das principais soluções de programação necessárias que os alunos devem abstrair (e.g., estrutura de seleção, conceito de operadores lógicos, estruturas de repetição). O <i>framework</i> é apresentado na Figura 3 (direita).

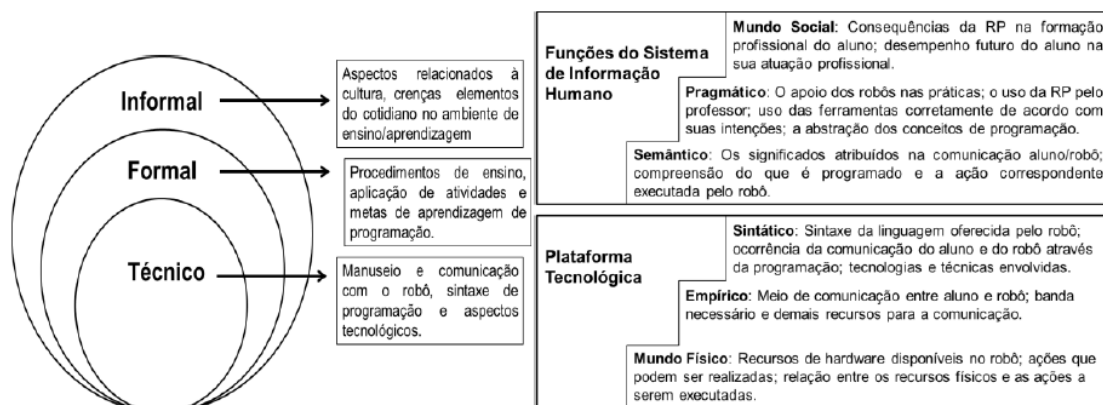


Figura 3. Cebola e Framework Semiótico

A Tabela 2 apresenta a Fase 2, cujo objetivo principal é analisar os agentes e seus *affordances*. A Figura 4 apresenta uma versão simplificada do diagrama de ontologia com agentes e papéis (e.g., professor e aluno de programação), *affordances* a serem trabalhados (e.g., identificar traçado e desviar de obstáculos) e dependências ontológicas (e.g., a programação do robô depende do robô e do aluno).

Tabela 2. Atividades e resultados da Fase 2 do PDAP-RP

Fase 2: Definição e modelagem de agentes, comportamentos e dependências ontológicas	
2.1. Definição de Agentes e <i>Affordances</i>	Foram definidos os agentes, incluindo “aluno” e “professor” como os principais responsáveis por vários <i>affordances</i> .
2.2. Agrupamento de Candidatos e modelagem de ontologias	Tabela listando os agentes e <i>affordances</i> encontrados na etapa anterior. Organizamos a tabela em duas colunas, sendo uma para as palavras candidatas a se tornarem <i>affordances</i> , e outra coluna com as classificações e informações que possam explicar suas funções.
2.3 Detalhamento de comportamentos e ações	Refinamento do artefato da etapa anterior, listando as dependências existentes entre as unidades semânticas, a partir de suas funções, comportamentos e ações.
2.4 Diagrama de Ontologia	Diagrama de Ontologia contendo as relações entre agentes e <i>affordances</i> (Figura 4).

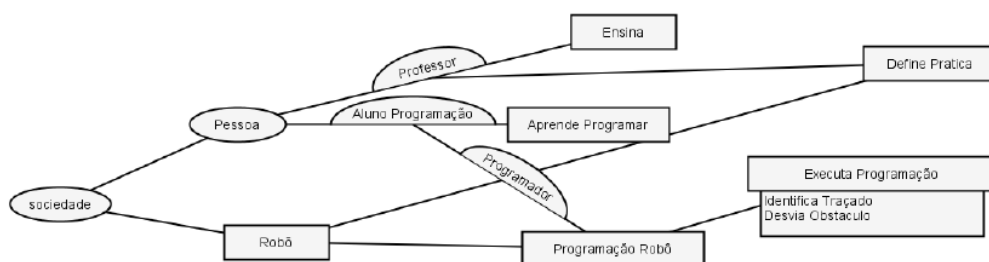


Figura 4. Diagrama de Ontologia

A Tabela 3 descreve a execução da Fase 3 incluindo a análise de normas e a elaboração de quadros descritivos das normas comportamentais, como no exemplo apresentado na Tabela 4. Esse exemplo detalha que o aluno é obrigado a descrever uma estrutura de seleção para o robô desviar de um obstáculo.

Tabela 3. Atividades e resultados da Fase 3 do PDAP-RP

Fase 3: Análise e modelagem de normas	
3.1. Análise de Normas	Descrição informal dos comportamentos dos agentes nos níveis pragmático e social.
3.1.1. Elaboração de quadro com normas comportamentais	Desenvolvimento de quadros que descrevem as normas comportamentais, contendo especificação sobre quais agentes estão envolvidos no cenário, quais são os <i>affordances</i> e o detalhamento dessa norma através do uso da Lógica Deontica para definir regras e ações, conforme Tabela 4.
3.1.2. Elaboração de	Tabela qualificando as diferentes normas. Nela consta questões relacionadas às diferentes normas

<i>quadro de respostas</i>	organizadas na seguinte maneira: Normas Perceptuais: O resultado da tarefa foi facilmente percebido? Teve alguma dificuldade em perceber os efeitos dos conceitos de programação? Normas Cognitivas: Houve dificuldades na compreensão da tarefa? Conseguiu compreender plenamente os efeitos dos conceitos no robô? Normas Avaliativas: Os resultados obtidos foram satisfatórios? Como avalia os efeitos obtidos? Normas Denotativas: Os conceitos de programação foram escolhidos adequadamente de acordo com a categoria/tipo de tarefa?
----------------------------	--

Tabela 4. Exemplo de quadro de normas comportamentais

<i>Norma</i>	<i>Desviar de um obstáculo na arena</i>
<i>Agentes</i>	Aluno (no papel de programador)
<i>Affordances</i>	Robô (<i>affordance</i> da Sociedade); Programação Robô (<i>affordance</i> aluno com o Robô); Desvia Obstáculo (<i>affordance</i> aluno por meio da programação do robô)
<i>Detalhamento da norma</i>	sempre que robô identificar um obstáculo na arena; se é preciso desviar desse obstáculo; então, o aluno; é obrigado a; utilizar uma estrutura de seleção para determinar o desvio

A Tabela 5 apresenta a Fase 4, com as atividades relacionadas à aplicação das práticas. Os resultados são descritos e comentados na próxima subseção.

Tabela 5. Atividades e resultados da Fase 4 do PDAP-RP

Fase 4: Aplicação de Práticas	
4.1. Definição das tarefas usando	Descrição da arena e tarefa conforme destacado anteriormente.
4.2. Avaliação dos resultados	Tabulação das respostas dos alunos a partir das normas perceptuais, cognitivas, avaliativas e denotativas (a próxima subseção descreve respostas às normas perceptuais e cognitivas).

5.2. Análise dos Resultados

Para analisar os resultados obtidos na prática foram elaboradas quatro questões relacionadas às normas perceptuais e cognitivas: (1) “O resultado da tarefa feita foi facilmente percebido?”, (2) “Teve alguma dificuldade em perceber os efeitos dos conceitos de programação?”, (3) “Houve dificuldades na compreensão da tarefa?”, e (4) “Conseguiu compreender plenamente os efeitos dos conceitos no robô?”.

A Tabela 6 sintetiza as respostas dos alunos às questões apresentadas. Ao analisar as respostas é possível identificar que houve uma percepção consistente quando considerado a compreensão dos conceitos de programação e o efeito positivo na utilização do robô para isso.

Tabela 6. Exemplo de quadro de respostas

	O resultado da tarefa foi facilmente percebido?	Teve alguma dificuldade em perceber os efeitos dos conceitos de programação?	Houve dificuldades na compreensão da tarefa?	Conseguiu compreender plenamente os efeitos dos conceitos no robô?
Aluno 1	Sim, foi fácil.	Não houve dificuldade em relação a usar os conceitos de programação.	Não, entender o que o robô deveria realizar durante sua execução não foi difícil.	Sim, ficaram bastante claros.
Aluno 2	Sim, o objetivo foi facilmente compreendido.	Era possível identificar quais ferramentas precisavam ser utilizadas.	Não.	Houve algumas variações na resposta do robô a certos comandos, mas basicamente ele correspondia ao que era esperado.
Aluno 3	Sim, foi facilmente percebido.	De certa forma sim, mas por causa do ambiente de programação que não trabalhava diretamente com o código.	Não.	Sim, embora devido a erros do robô algumas instruções eram executadas de forma que não correspondiam ao que era programado.

Aluno 4	<i>Sim.</i>	<i>Após os primeiros testes foi mais fácil identificar.</i>	<i>Não.</i>	<i>Quase sempre. Só tive problemas quando o robô respondia de maneira estranha.</i>
Aluno 5	<i>Foi fácil perceber.</i>	<i>Os efeitos da programação no robô eram claros pra mim.</i>	<i>Não tive dificuldades.</i>	<i>Sim, consegui ver as ações dele de maneira correta.</i>
Aluno 6	<i>Sim.</i>	<i>Consegui perceber quais eram os conceitos necessários e usá-los.</i>	<i>Eu não tive dificuldades em compreender.</i>	<i>Sim, consegui ver as suas ações corretamente. Em alguns momentos ele não respondia corretamente, mas acredito que não era erro da programação.</i>

De acordo com as respostas da Tabela 6, o robô conseguiu proporcionar um ambiente concreto amenizando a barreira da abstração. O retorno positivo pelos alunos também demonstra indicativos da viabilidade da metodologia na análise tanto dos aspectos comportamentais, quanto dos cognitivos e perceptuais. Uma análise das respostas também demonstra indícios iniciais de melhoria na aprendizagem, apontando uma compreensão positiva dos efeitos dos conceitos de programação junto ao robô.

6. Considerações Finais

O PDAP-RP mostrou que o processo tem potencial para o desenvolvimento e o fornecimento de instrumentos de avaliação de práticas com RP. O diferencial deste processo é a utilização SO na formatação de práticas, assim como a avaliação de resultados através de uma perspectiva orientada a normas.

Vários aspectos que envolvem professores e alunos podem ser descritos e analisados por meio dessa perspectiva. Levando-se em conta que a utilização da RP para o ensino de programação de computadores não é algo trivial, é essencial ter meios para auxiliar nesse processo. O uso da metodologia proporciona uma visão na qual o ambiente de aprendizagem se apresenta como um sistema de signos, o qual pode ser analisado através de seus aspectos semânticos, pragmáticos e sociais descritos em *affordances* e normas. A compreensão desse sistema auxilia na identificação de quais são as etapas fundamentais para a construção de práticas que possam contemplar toda estrutura didática, da concepção à avaliação. Além disso, a metodologia contribui potencializando a uso da RP, indo além dos aspectos motivacionais, proporcionando a possibilidade de estabelecer atividades organizadas que podem ser revistas para uma melhoria contínua. A busca pela sistematização da criação de práticas possibilita identificar pontos específicos no qual houve erros durante sua elaboração.

É possível destacar algumas limitações da metodologia e processo, por exemplo, em situações (embora não frequentes) com respostas inesperadas ou inconsistentes em ações do robô, sem que a programação fosse responsável por tais erros. Outra limitação chave é que ainda não foi possível obter as respostas no nível de granularidade necessária a uma avaliação estatística quantitativa dos resultados da prática.

Entretanto, os resultados preliminares obtidos na prática mostram o potencial do PDAP-RP trazer benefícios no estudo de práticas de RP. As etapas possibilitam um encadeamento de trabalho que favorece a identificação dos objetivos pedagógicos no contexto de programação de computadores, responsabilidades dos alunos e professor envolvidos, definição de metas de desempenho e recursos de avaliação satisfatórios em relação às habilidades técnicas e cognitivas desenvolvidas pelo aluno. Como trabalho futuro, espera-se um refinamento maior da metodologia e processo, objetivando melhorar os indicativos de desempenho do aluno. Para tanto, já estão sendo realizados

experimentos com grupos distintos de alunos (iniciante e com experiência), bem como a aplicação com um número maior de participantes.

7. Referências

- Benitti, F. B. V. (2012). Exploring the educational potential of robotics in schools: A systematic review. *Computers & Education*, 58(3), 978-988.
- D'Abreu, J. V. (2002). Integração de Dispositivos Mecatrônicos para Ensino-Aprendizagem de Conceitos na Área de Automação. Tese de Doutorado em Engenharia Mecânica, Faculdade de Engenharia Mecânica, Universidade Estadual de Campinas, Campinas-SP.
- Gibson, J. J. (1968). *The Ecological Approach to Visual Perception*. Houghton Mifflin Company, Boston, Massachusetts, 127-143.
- Gomes, A., Henriques, J., e Mendes, A. J. (2008). Uma proposta para ajudar alunos com dificuldades na aprendizagem inicial de programação de computadores. *Educação, Formação & Tecnologias*.
- Gomes, A., Areias, C., Henriques, J., e Mendes, A. J. (2008). Aprendizagem de programação de computadores: dificuldades e ferramentas de suporte. *Revista Portuguesa de Pedagogia*, v. 42, n. 2.
- Liu, K. (2000). *Semiotics in Information System Engineering*. Cambridge: Cambridge University Press.
- Papert, S. (1994). *A máquina das crianças: repensando a escola na era da informática*. Porto Alegre: Artmed.
- Soares, R. F., e Borges, M. A. F. (2011). Robótica: aprendizado em informática de forma lúdica. In *WEI-XXXI Congresso da Sociedade Brasileira de Computação, Natal-RN: SBC*.
- Stamper, R. K. (1973). *Information in Business and Administrative Systems*, Hoboken: John Wiley & Sons.
- Stamper, R., Liu, K., Hafkamp, M., & Ades, Y. (2000). Understanding the roles of signs and norms in organizations-a semiotic approach to information systems design. *Behaviour & Information Technology*, 19(1), 15-27.
- Valente, J. A. (1993). Por quê o computador na educação. *Computadores e Conhecimento: repensando a educação*. Campinas: Gráfica da UNICAMP.
- Zanetti, H.A.P.; Bonacin, R.. Uso de semiótica e análise de normas em práticas de ensino de Programação de Computadores utilizando Robótica Pedagógica. *Revista Eletrônica de Tecnologia e Cultura*, v. 1, n. 1, 2014.
- Zilli, S. R. (2004). *A robótica educacional no ensino fundamental: Perspectivas e prática*. Dissertação de Mestrado em Engenharia, Universidade Federal de Santa Catarina, Centro Tecnológico, Florianópolis-SC.

ANEXO IV – IMAGENS DAS OFICINAS



ANEXO V – QUESTIONÁRIOS APLICADOS AOS ALUNOS

Questionário 1

Questionário sobre práticas com Robótica Pedagógica

***Obrigatório**

Coloque as iniciais do seu nome, separados por ponto (.), sem espaços. *

Por exemplo: Jose da Silva Souza = J.S.S.

Qual é sua idade? *

Qual é seu sexo? *

☐ Masculino

☐ Feminino

Está em qual ano do seu curso? *

☐ 1º ano

☐ 2º ano

☐ 3º ano

1. Houve dificuldades na compreensão da tarefa?

2. O objetivo da tarefa foi facilmente percebido?

3. Teve alguma dificuldade em perceber os efeitos dos conceitos de programação?

4. Conseguiu compreender plenamente os efeitos dos conceitos de programação no robô?

5. Houve dificuldades em entender as respostas (feedback) do robô para as ações programadas?

6. Os resultados obtidos foram satisfatórios para seu aprendizado?

7. A prática ajudou na aquisição de novos conceitos de programação ou ajudou a concretizar ainda mais seu conhecimento?

8. Os conceitos de programação foram escolhidos adequadamente de acordo com a categoria/tipo de tarefa?

Enviar

Nunca envie senhas em Formulários Google.

Powered by
 Google Forms

Este conteúdo não foi criado nem aprovado pelo Google.
[Denunciar abuso](#) - [Termos de Serviço](#) - [Termos Adicionais](#)

Questionário 2 - Práticas com Robótica Pedagógica

Para as respostas desse questionário, assumo a seguintes escala:

- 1 - Discordo totalmente
- 2 - Discordo parcialmente
- 3 - Indiferente (Não concordo, nem discordo)
- 4 - Concordo parcialmente
- 5 - Concordo totalmente

***Obrigatório**

Coloque as iniciais do seu nome, separados por ponto (.), sem espaços. *

Por exemplo: Jose da Silva Souza = J.S.S.

Qual é sua idade?

Qual é seu sexo? *

- ☐ Masculino
- ☐ Feminino

Está em qual ano do curso? *

- ☐ 1º ano
- ☐ 2º ano
- ☐ 3º ano

1. A Robótica Pedagógica é um instrumento interessante para ensino de programação. *

1 2 3 4 5

discordo totalmente ☐ ☐ ☐ ☐ ☐ concordo totalmente

2. A comunicação com o robô ocorreu sem problemas ou dificuldades. *

1 2 3 4 5

discordo totalmente ☐ ☐ ☐ ☐ ☐ concordo totalmente

3. Os recursos (sensores, motores, etc) do robô foram percebidos de maneira clara. *

1 2 3 4 5

discordo totalmente ☐ ☐ ☐ ☐ ☐ concordo totalmente

4. As respostas dadas pelo robô à programação executada foram claras e coerentes.

1 2 3 4 5

discordo totalmente ☐ ☐ ☐ ☐ ☐ concordo totalmente

5. A utilização da Robótica Pedagógica auxiliou na aprendizagem ou aumentou seu conhecimento nos conceitos de programação. *

1 2 3 4 5

discordo totalmente ☐ ☐ ☐ ☐ ☐ concordo totalmente

6. A forma de avaliação da prática era bastante clara e coerente. *

1 2 3 4 5

discordo totalmente ☐ ☐ ☐ ☐ ☐ concordo totalmente

7. Não há problemas/dificuldades em aplicar os conceitos de Lógica de Programação na Programação Visual por blocos. *

1 2 3 4 5

discordo totalmente ☐ ☐ ☐ ☐ ☐ concordo totalmente

8. Não houve problema em compreender os objetivos da prática proposta. *

1 2 3 4 5

discordo totalmente ☐ ☐ ☐ ☐ ☐ concordo totalmente

9. Não houve dificuldade em elaborar a solução da prática proposta. *

1 2 3 4 5

discordo totalmente ☐ ☐ ☐ ☐ ☐ concordo totalmente

10. Não houve dificuldades em aplicar os conceitos de Lógica de Programação (estruturas de decisão, estruturas de repetição e modularização) na solução. *

1 2 3 4 5

discordo totalmente ☐ ☐ ☐ ☐ ☐ concordo plenamente

Enviar

ANEXO VI – QUADROS DE RESPOSTAS

QUESTIONARIO 1 - GRUPO DE CONTROLE

Aluno	Questão 1	Questão 2	Questão 3	Questão 4	Questão 5	Questão 6	Questão 7	Questão 8
T.F	Não, as explicações foram bem claras e de fácil compreensão.	Sim, foram facilmente percebido	Em algumas coisas, mas foram superadas.	Sim, consegui!	Em algumas coisas...	Sim, muitas coisas me ajudaram para melhor desempenho para o meu aprendizado.	Sim, ajudou em ambas as partes.	Sim, foram de acordo!
L.R.S	Não, O conteúdo foi bem explicado, da maneira que consegui entender o que foi passado.	Sim. Estávamos recebendo um treinamento.	Um pouco.Pelo fato dos conceitos de programação serem mais complicados. Isso na minha opinião.	Sim, foi de fácil compreensão.	Não, consegui entender as respostas para as ações programadas.	Sim. Ampliei meu conhecimento nessa área e dessa forma os resultados obtidos foram satisfatórios.	Os dois. A prática ajudou na aquisição de novos conceitos de programação e me ajudou a conceituar ainda mais seu conhecimento.	Sim. Foram bem ensinados, e conseguimos entender.
R.P.E	Não, pois para mim ficou claro a compreensão quanto a tarefa.	Sim, foi claramente entendida o objetivo.	Não, pois com atenção deu para perceber os efeitos causados.	Mais ou menos, as vezes surgia uma pouco de dificuldades.	Um pouco, em algumas vezes o robô não atendia ao programa feito.	Nem tanto, ajudou sim em meu aprendizado mas acho que não o bastante a ponto de dizer que me satisfaz.	Sim, ajudou bastante em algumas dúvidas que foram surgindo aos poucos.	Sim, foram ensinados adequadamente de acordo com o tipo da tarefa passado pela tutora
G.G.C.C.	Não, para mim não foi difícil de compreender pois esta tudo ligado a programação de computador.	sim, desde o primeiro encontro o professor relatou o objetivo de estarmos recebendo o curso.	não, os conceitos da robótica esta completamente ligado aos conceitos de programação.	sim, fazendo um paralelo com programação. Até ficou mais claro testando no robô, do que os conceitos que eu usava em programação.	não, pois logo que que ele executava as ações eu já visualizava os comandos que eu tinha de programação.	sim, tirei as pequenas dúvidas que eu tinha e me ajudou não matérias novas de programação.	na minha opinião a pratica me ajudou nas duas coisas.	sim, foi de muito bom senso do professor ter escolhido dessa forma pois me ajudou a entender o resultado final da programação.
L.R.O	No começo sim, pois não consegui entender direito como fazer e o que deveria ser feito, mas depois eu consegui entender direito.	Não, foi difícil de perceber, pois tem coisas muito variadas para fazer e não foi explicado como tinha que ser feito.	Tive, pois tem muitos comandos e eu não consegui entender o que cada um fazia.	Na primeira aula eu não entendi nada, pois foi apenas teórico, na segunda aula por ser pratica eu consegui entender tudo o que usamos.	Não, quando eu entendi alguma coisa eu consegui perceber claramente o que eu entendi.	Sim, eu consegui entender códigos que eu não sabia.	Sim, eu consegui entender Switch Case.	Foram, eles usaram apenas códigos que já sabíamos usar.
T.R.P.	Não, pois foi bem explicado a matéria.	Sim, pois as instruções dadas foram claras.	Um pouco, pois na hora de por em pratica era difícil, porém não era impossível, e após um tempo achemos a resolução.	Para fazer a programação foi um pouco complicado no começo, mas depois deu para compreender.	Sim, no começo, mas depois de um tempo deu para compreender e perceber que o erro estava na programação.	Sim, pois eu consegui aprender e executar um programa no LEGO.	Ela ajudou a conceituar mais meu conhecimento, pois havia alguns assuntos de programação que eu não entendia totalmente, e com o LEGO eu entendi.	Sim, quando precisávamos usar sensores, ela dava tarefas com sensores.
F.A.P.A.	Inicialmente houve dificuldades para o entendimento da função que cada bloco de comando exercia sobre o robô. Porém, após ver o funcionamento do robô, o conteúdo foi fixado mais facilmente.	Sim. Os objetivos de cada tarefa foi facilmente percebido, de forma que pudéssemos desenvolver a lógica para que o robô executasse as tarefas de forma correta e precisa.	Os conceitos de programação já aprendidos anteriormente foram de grande importância para a programação no robô, já que alguns conceitos já tinham sido apresentados anteriormente nas diversas matérias do curso de Informática.	Em relação aos conceitos aprendidos durante o treinamento, houve certa facilidade em sua compreensão. Porém, fica claro como poderíamos ter explorado de forma mais profunda alguns outros conceitos que foram apresentados apenas de forma superficial.	Não. As respostas dadas pelo robô esclarecem a programação "abstrata" realizada no computador, mesmo que o robô não responda com cem por cento de exatidão aos comandos.	Sim, foram satisfatórios. Já que o conhecimento é apenas acumulado, podemos aplicar conceitos já aprendidos na robótica e vice-versa.	A pratica no treinamento de robótica, mesmo que em pequena quantidade, ajudou a conceituar o conhecimento. Tornando os conceitos menos abstratos.	Claramente. Em cada tarefa pedida, podemos aplicar diferentes conceitos de programação, como "looping", "switch-case" entre outros...
A.A	Um pouco, pois houve uma pequena dificuldade em selecionar os motores que deveriam ser utilizados.	Sim, o propósito de cada ação/tarefa foi compreendido.	Não, cada efeito foi bem colocado e compreendido.	Apenas dificuldade na escolha dos motores de partida.	Não, os erros de utilização de comandos foram corrigidos e melhorados.	Sim, o desenvolvimento de cada programa ajudou a encher e entender tarefas distintas e a compreender os objetivos propostos.	Sim, o desenvolvimento de tarefas e objetivos foram melhor compreendidos.	Sim, cada tarefa foi realizada adequadamente.

QUESTIONARIO 1 - GRUPO 1

Aluno	Questão 1	Questão 2	Questão 3	Questão 4	Questão 5	Questão 6	Questão 7	Questão 8
A.S.T.	Não, foi fácil compreender como meier no programa, pois tinha estruturas parecidas com as outras linguagens de programação.	Sim, pois houve uma explicação relativamente boa, além de uma interação com o robô, o que torna mais atrativo para que possamos programar.	Não muitas, pois as estruturas de programação eram bem parecidas com as outras linguagens C, mesmo sendo em blocos.	Sim, pois com a interação do robô, fica muito mais fácil de ver os resultados obtidos com sua programação imediatamente, o que traz certa satisfação para si mesmo.	Não, pois ao olhar as ações executadas realizadas pelo robô e ver o programa, a visualização do erro é relativamente fácil, pelo modo em blocos que se programa o robô.	Sim, pois diferente das linguagens de programação, o robô envolve muitos outros fatores, como gravidade, obstáculos, caminhos, etc, o que torna a programação um pouco mais difícil, mas legal de aprender.	Sim, porque eu ainda não havia muita experiência com programação em blocos, o que ajudou muito para eu poder usar outros programas também.	Sim, foi facilmente descoberto qual conceito era adequado para cada tarefa.
G.B.C	No início sim por ser uma coisa nova, mais depois que notei que era igual a aula de programação foi fácil.	Sim, por ser feita com o intuito de facilitar na compreensão da aula de programação e para trazer um modo curioso para os alunos entender a linguagem C.	Não, pois usa os mesmos códigos da linguagem C.	Sim, com a ajuda da aula de programação ficou mais fácil entender os códigos e as sequências que eles tinham fazer.	No início sim, mas depois ficou mais fácil de fazer.	Sim, pois a partir do robô entendi melhor o conceito, tive mais animo de continuar no curso e gostei muito de ver como funciona um robô e seu comandos.	A aula de robótica ajudou a concretizar ainda mais meus conhecimentos em programação.	Sim, pois a partir dos conhecimentos que já tínhamos de programação ficou mais fácil de fazer a programação do robô.
W.P	Não, a tarefa é muito fácil de entender.	Sim, pois a programação em robôs é pouco complexa.	Não, são conceitos parecidos com programação em linguagem C.	Sim, os efeitos eram claramente percebidos principalmente na hora de estar o programa.	Não, pois ele respondia ao que se pedia em cada programa.	Sim, foram satisfatórios pois a cada erro descobria o que deveria ser feito.	A prática ajudou a concretizar ainda mais meu conhecimento.	Sim, pois se adapta de acordo com as funções escolhidas.
L.M.E.S	Por ser feito em pequenos grupos a compreensão foi facilmente percebida e compreendida, além de já saber um pouco de programação o que ajudou muito a entender o que foi passado, posso dizer que foi realmente muito bem apresentado aos alunos.	Sim, depois de apresentadas as funções, o objetivo a ser alcançado foi o mais fácil de entender.	O próprio nome do conceito já deixa claro o que é e para que serve, sendo raras as dificuldades de compreensão e uso de sua programação.	No robô foi mais difícil pois o que foi passado foi o aprendizado com base nas funções a serem usadas na CBR, ou seja, só nos foi ensinado o que é usado na CBR.	Não, o robô responde muito bem ao propósito, onde os erros são por parte do programador, pois tem também um recurso q avisa a maior parte dos erros.	Sim, tudo o que foi proposto e ensinado foi recebido e aprendido por mim, sabendo executar com exito as funções propostas, claro sendo as que foram ensinadas.	A base ensinada foi muito parecido com os conceitos estudados na matéria de programação, onde ajudou minha parte a compreender o que eu estava fazendo com ainda mais novo e sentido.	Sim, foi passado para nós os conceitos a ser aplicado no robô, sendo os mesmos conceitos já aprendidos em sala, estando de acordo com o que já sabíamos foi sim adequado a nós.
L.A.T.	Não, o programa possui um uso extremamente simples, e de fácil compreensão.	Sim, o principal objetivo, que foi dar uma instrução para o robô seguir uma linha, foi executado sem nenhum problema.	Não muito, pois o laço de repetição, mais de dificuldade, mais foi compreendido um pouco depois, sem dificuldade.	Sim, os conceitos principais de programação com laço de repetição e tomada de decisões foram aplicados no robô sem problemas.	Um pouco, pois tinha algumas vezes, que por pequenitas falhas no programa, o robô estava executando outra coisa.	Sim, pois vendo os comandos sendo executados na prática, foi mais fácil de ser compreendido sua real função.	Ajudou a concretizar meus conhecimentos, principalmente em laço de repetição, o logo me ajudou a entender melhor.	Sim, foram usados os comandos necessários para a tarefa, como a seguir uma trajetória, que usamos laço de repetição para seguir o preto e tomada de decisão caso saísse da linha.
V.L.	Na compreensão da tarefa não houve dificuldades, pois já entendíamos o que era para ser feito e a proposta era bem clara.	Sim, foi facilmente percebido pois o objetivo proposto era bem claro.	Um pouco, encontrei mais dificuldade a esse respeito pois ainda tinha dúvidas nos conceitos da programação.	A princípio não, pois ainda tinha certas dúvidas em relação a programação, consequentemente não pude visualizar claramente todos os efeitos da programação no robô.	Não, quando o robô executava os comandos dados conseguir entender claramente, a dificuldade mesmo foi apenas em determinados momentos da programação.	Sim, pude perceber que a programação no robô e exatamente igual a da matéria de lógica de programação porém no robô os comandos são visualizados mais facilmente por estarem em "campos" e não termos que monitor a linha de comando inteira.	Me ajudou na aquisição de novos conceitos, pois antes de começar com o projeto de robótica ainda encontrava dificuldades na compreensão dos comandos e conceitos para realizar a programação.	Sim de acordo com os conhecimentos que eu tinha dos comandos creio que foram bem adaptados e adequados para o robô, pois mantinha os mesmos princípios da programação em si so que mais simplificados e fáceis de ser compreender.

QUESTIONARIO 1 - GRUPO 1 (continuação)

Aluno	Questão 1	Questão 2	Questão 3	Questão 4	Questão 5	Questão 6	Questão 7	Questão 8
B.P.D.	O projeto LEGO foi de grande auxílio para com os alunos pois nós compreendemos melhor a lógica da programação a princípio foi um pouco estranho pois o robô era programado através não da linha, mas de blocos. Após um certo período ficou mais fácil compreender a programação, portanto posso afirmar sem dificuldades que não houve dificuldade com relação ao projeto.	Sim, nós pudemos perceber claramente que a intenção do projeto LEGO era nos auxiliar a entender e compreender a programação, apesar de ela ser feita em blocos foi de grande ajuda pois os conceitos das estruturas de programação do LEGO.	Não, os conceitos de lógica de programação ensinados a nós no decorrer das aulas já nos foi de grande ajuda pois tornou mais fácil a compreensão das estruturas de programação do LEGO.	Sim, porém devo admitir que se nós pudéssemos ter mais aulas de LEGO seria de maior auxílio, nós vimos apenas alguns conceitos básicos, se nossa escola tivesse mais LEGOs talvez nós pudéssemos ter o LEGO como uma matéria e não apenas uma semana de aula sobre o LEGO, o que nos daria a chance de aprender mais sobre ele.	Não exatamemente a nossa dificuldade foi com relação a como fazer o robô girar exatamente 90° nós não conseguimos toda a vez que tentávamos ele ia ou mais pra frente ou mais pra trás, foi a única dificuldade o restante, os motores e as engrenagens eram mais simples.	Sim sem vestígio de dúvida, ele nos mostrou o que nós podemos fazer com a programação além é claro de ilustrar nosso aprendizado. O que nos gera um grande conhecimento de programação e nos dá uma base para conhecer outros métodos de programação.	Sim, ele beneficiou muito nosso conhecimento de programação pois durante o projeto nós tivemos de pensar e planejar a programação ver se um laço de repetição ajudaria, ou então se deveria usar if else, o que foi de grande ajuda.	Sim, cada tipo de tarefa tinha a intenção clara de nos ajudar em um tipo de programação, e eram tarefas que avançavam no nível de dificuldade o que nos dava uma maior ajuda para compreender a programação, por exemplo quando aprendemos a mexer nos motores, nós tivemos de fazer ele andar reto, depois fazer ele andar e virar e por último fazer ele andar, girar 90° andar mais um pouco e depois parar e o mesmo pode se dizer das outras atividades. Elas forma muito bem elaboradas, e nós processamos cada atividade muito bem, sempre fazendo com que o robô cumprisse o objetivo proposto.

QUESTIONARIO 1 - GRUPO 2

Aluno	Questão 1	Questão 2	Questão 3	Questão 4	Questão 5	Questão 6	Questão 7	Questão 8
L.C.F.	Não houve dificuldades na compreensão das tarefas, apenas a explicação da tarefa foi dada.	Sim, pois a tarefa foi explicada e o objetivo foi claro.	Não, pois a tarefa foi explicada e o objetivo foi claro.	Sim, pois a tarefa foi explicada e o objetivo foi claro.	Não, pois a tarefa foi explicada e o objetivo foi claro.	Sim, pois a tarefa foi explicada e o objetivo foi claro.	Sim, pois a tarefa foi explicada e o objetivo foi claro.	Sim, pois a tarefa foi explicada e o objetivo foi claro.
E.B.P.S	Não, as tarefas foram bem explicadas, apenas a explicação da tarefa foi dada.	Sim, pois a tarefa foi explicada e o objetivo foi claro.	Não, pois a tarefa foi explicada e o objetivo foi claro.	Sim, pois a tarefa foi explicada e o objetivo foi claro.	Sim, pois a tarefa foi explicada e o objetivo foi claro.	Sim, pois a tarefa foi explicada e o objetivo foi claro.	Sim, pois a tarefa foi explicada e o objetivo foi claro.	Sim, pois a tarefa foi explicada e o objetivo foi claro.
S.A.C.J.	Em alguns casos, pois não houve uma explicação clara.	Sim, pois a tarefa foi explicada e o objetivo foi claro.	Não, pois a tarefa foi explicada e o objetivo foi claro.	Não, pois a tarefa foi explicada e o objetivo foi claro.	Sim, pois a tarefa foi explicada e o objetivo foi claro.	Sim, pois a tarefa foi explicada e o objetivo foi claro.	Sim, pois a tarefa foi explicada e o objetivo foi claro.	Sim, pois a tarefa foi explicada e o objetivo foi claro.
R.J.C.F.	Não, a maioria das tarefas foram bem explicadas pelos professores e organizadores.	Sim, pois a tarefa foi explicada e o objetivo foi claro.	Não, pois a tarefa foi explicada e o objetivo foi claro.	Sim, pois a tarefa foi explicada e o objetivo foi claro.	Sim, pois a tarefa foi explicada e o objetivo foi claro.	Sim, pois a tarefa foi explicada e o objetivo foi claro.	Sim, pois a tarefa foi explicada e o objetivo foi claro.	Sim, pois a tarefa foi explicada e o objetivo foi claro.
F.M.T	Na maioria dos casos não houve problemas de compreensão.	Sim, pois a tarefa foi explicada e o objetivo foi claro.	Não, pois a tarefa foi explicada e o objetivo foi claro.	Sim, pois a tarefa foi explicada e o objetivo foi claro.	Sim, pois a tarefa foi explicada e o objetivo foi claro.	Sim, pois a tarefa foi explicada e o objetivo foi claro.	Sim, pois a tarefa foi explicada e o objetivo foi claro.	Sim, pois a tarefa foi explicada e o objetivo foi claro.
G.C.	Não, como foi bem explicado, não houve dificuldades de compreensão.	Sim, pois a tarefa foi explicada e o objetivo foi claro.	Não, pois a tarefa foi explicada e o objetivo foi claro.	Sim, pois a tarefa foi explicada e o objetivo foi claro.	Sim, pois a tarefa foi explicada e o objetivo foi claro.	Sim, pois a tarefa foi explicada e o objetivo foi claro.	Sim, pois a tarefa foi explicada e o objetivo foi claro.	Sim, pois a tarefa foi explicada e o objetivo foi claro.
W.Z.A	Não, a compreensão da tarefa foi dada, apenas a explicação da tarefa foi dada.	Sim, pois a tarefa foi explicada e o objetivo foi claro.	Não, pois a tarefa foi explicada e o objetivo foi claro.	Sim, pois a tarefa foi explicada e o objetivo foi claro.	Sim, pois a tarefa foi explicada e o objetivo foi claro.	Sim, pois a tarefa foi explicada e o objetivo foi claro.	Sim, pois a tarefa foi explicada e o objetivo foi claro.	Sim, pois a tarefa foi explicada e o objetivo foi claro.
C.F.R	Sim, pois no começo da tarefa, a explicação da tarefa foi dada.	Sim, pois a tarefa foi explicada e o objetivo foi claro.	Não, pois a tarefa foi explicada e o objetivo foi claro.	Sim, pois a tarefa foi explicada e o objetivo foi claro.	Sim, pois a tarefa foi explicada e o objetivo foi claro.	Sim, pois a tarefa foi explicada e o objetivo foi claro.	Sim, pois a tarefa foi explicada e o objetivo foi claro.	Sim, pois a tarefa foi explicada e o objetivo foi claro.
C.V.G.C.S	Não houve dificuldades na compreensão da tarefa, apenas a explicação da tarefa foi dada.	Sim, pois a tarefa foi explicada e o objetivo foi claro.	Não, pois a tarefa foi explicada e o objetivo foi claro.	Sim, pois a tarefa foi explicada e o objetivo foi claro.	Sim, pois a tarefa foi explicada e o objetivo foi claro.	Sim, pois a tarefa foi explicada e o objetivo foi claro.	Sim, pois a tarefa foi explicada e o objetivo foi claro.	Sim, pois a tarefa foi explicada e o objetivo foi claro.
P.W.C.	Sim, em alguns momentos as tarefas pareciam um pouco confusas, em relação ao trajeto a ser realizado.	Sim, pois a tarefa foi explicada e o objetivo foi claro.	Não, pois a tarefa foi explicada e o objetivo foi claro.	Sim, pois a tarefa foi explicada e o objetivo foi claro.	Sim, pois a tarefa foi explicada e o objetivo foi claro.	Sim, pois a tarefa foi explicada e o objetivo foi claro.	Sim, pois a tarefa foi explicada e o objetivo foi claro.	Sim, pois a tarefa foi explicada e o objetivo foi claro.
V.A.C	Não, as tarefas foram bem explicadas e todas as dúvidas foram respondidas.	Sim, pois a tarefa foi explicada e o objetivo foi claro.	Não, pois a tarefa foi explicada e o objetivo foi claro.	Sim, pois a tarefa foi explicada e o objetivo foi claro.	Sim, pois a tarefa foi explicada e o objetivo foi claro.	Sim, pois a tarefa foi explicada e o objetivo foi claro.	Sim, pois a tarefa foi explicada e o objetivo foi claro.	Sim, pois a tarefa foi explicada e o objetivo foi claro.

QUESTIONARIO 1 - GRUPO 2 (continuação)

Aluno	Questão 1	Questão 2	Questão 3	Questão 4	Questão 5	Questão 6	Questão 7	Questão 8
J.M.P.F	Não, todas as tarefas foram explicadas e dúvidas foram solucionadas.	Sim, algumas das tarefas eram bastante complexas, mas todas de fácil interpretação.	Não, a linguagem de programação para o Lego é muito simples, não há nada de grande complexidade na programação, era basicamente o que já havíamos visto nas aulas de L.P.	Sim, já que todos os conceitos usados para a programação dos robôs já haviam sido ensinados nas aulas L.P.	Os problemas que os robôs apresentavam, na maioria das vezes, eram falhas mecânicas.	Os conceitos foram na verdade aprimorados.	Sim, com os robôs, é mais fácil ver o resultado da programação, pois o plano físico facilita a visualização.	Sim, mas adaptações devem ser feitas na programação para que os robôs apresentem isso no real (não no virtual).
R.G.C.J	Sim, as tarefas passadas foram compreendidas sem alguma dificuldade.	Sim, os objetivos foram de fáceis compreensão.	Não, os conceitos de programação aprendidos anteriormente foram facilmente compreendidos.	Sim, foi possível compreender totalmente os efeitos no robô.	Não, a lógica utilizada foram entendidas totalmente pelo robô.	Sim, foi de grande satisfação ver os resultados obtidos.	Sim, após o aprendizado, muitas coisas foram desconhecidas foram compreendidas.	Sim, os conceitos usados foram adequados a cada tarefa.
J.G.S.G	Um pouco, pois em alguns casos nem todos os pontos de avaliação eram claros.	Sim, mas poderia ser mais simples, deixando os pontos avaliados as vezes não muito claros.	Não, pois os comandos em linguagens comuns e as ferramentas comuns são semelhantes.	Sim, pois são bem claros e objetivos.	As vezes, já que ocorriam alguns bugs e problemas de calibração.	Sim, muito, pois a lógica do robô é igual a do computador, com comandos semelhantes.	Sim, pois com ele podemos ver a execução e lógica de uma forma mais simples e prática.	Sim
W.M.C.	Algumas propostas não tiveram explicações coerentes.	A maior parte foi de fácil percepção.	Não, foi de fácil percepção os conceitos lógicos.	Sim, foi fácil utilizar os conceitos aprendidos nas aulas de Lógica de Programação.	Na maior parte das vezes o feedback do robô foi coerente com a programação realizada.	Certo que sim, foi mais fácil ilustrar os conceitos lógicos e com isso melhorando o entendimento geral sobre programação.	Ajudou em ambos os casos tanto para novos conceitos e para concretizar os já adquiridos.	Na maior parte das vezes foi escolhido adequadamente
J.M.P.S	Não, nenhuma. As tarefas eram sempre um objetivo final claro.	Sim, foi parcialmente fácil e não houve problemas.	Não, os efeitos da programação são diretamente colocados na prática da robótica.	Sim, os efeitos aparecem em todos os momentos durante a programação.	Não, com as boas programações as quais aprendemos foi fácil perceber os efeitos.	Sim, os resultados foram demasiados grandes.	Com certeza sim! A prática ajuda no pensamento e faz com que os conhecimentos fiquem mais ativos e a programação se torne mais prazerosa.	Sim. Sempre foram. Foi sempre coerente com o aprendizado.
C.I.A	Não, recebemos uma folha bem detalhada de como realizar a tarefa, e antes houve uma explicação boa e qualquer dúvida sobre a tarefa que surgiram durante o desenvolvimento foram respondidas.	Sim, todo material recebido era bem detalhado e especificando os objetivos.	Não, ao compreender a tarefa, facilmente pode-se perceber quais conceitos deveriam ser utilizados.	A maioria dos conceitos de programação foram implementados e compreendidos facilmente.	Algumas vezes, pois muitas vezes que o robô deveria realizar eram influenciadas pela bateria desta, o que dificultava no momento de criar um programa exato para realizar as tarefas.	Sim, foram satisfatórios e agregou conhecimento na área de robótica.	Ajudou a concretizar ainda mais meus conhecimentos obtidos em sala de aula.	Sim, os conceitos foram escolhidos adequadamente para cada tipo de tarefa.
J.G.P.	Não, pois foi claramente percebido o que deveria ser feito	Sim, foi claramente explicado	Não, devido a familiaridade	Sim, pois a programação foi consideravelmente simples devido a ser programação em blocos	Em alguns casos, pois em um certo robô, tivemos o problema de o sensor de cor estar identificando uma cor em lugar que ela não existia	Sim, ajudou a aprimorar meu conhecimento	Sim, ajudou a concretizar o conhecimento, mais especificamente em estruturas condicionais	Sim, foram todos totalmente de acordo
B.H.T	Não houve dificuldade na compreensão da tarefa, pois foi bem esclarecido os objetivos da tarefa.	Sim, foi facilmente percebido, os objetivos eram bem claros.	Não, os objetivos junto com a tarefa eram bem claros, submetendo na facilidade de compreender os conceitos de programação.	Sim, os robôs eram simples de ser manipulados, como também, fornecendo uma compreensão bem sucedida dos seus conceitos.	Não houve, as ações eram consideravelmente claras para a compreensão.	Foram sim, houve um grande auxílio, foram resultados muito gratificantes, pois obtive uma grande carga de aprendizado para fazer relações entre o robô e as ações programadas, como também os conceitos de programação.	Ajudou a concretizar ainda mais meu conhecimento sobre conceito de programação, como assim dito, teve grande auxílio.	Foram bem escolhidos, as relações estabelecidas eram bem sucedidas, levando a perceber a escolha ideal diante do conceito e do tipo de tarefa.

QUESTIONARIO 1 - GRUPO 2 (continuação)

Aluno	Questão 1	Questão 2	Questão 3	Questão 4	Questão 5	Questão 6	Questão 7	Questão 8
A. G. R. S.	Não houve dificuldade, mas é necessário trabalhar melhor técnicas relacionadas a lógica de programação. Creio que a forma que é trabalhada com os equipamentos LEGO não são necessariamente a melhor forma de praticar e elaborar lógicas e compreender as tarefas, creio que a utilização de outros Kits de desenvolvimento tenha um maior efeito sobre os alunos.	Sim, os objetivos foram apresentados de modo fácil.	Sim, no início foi difícil a compreensão pela falta de precisão dos Kits LEGO.	Sim, foi possível a compreensão dos conceitos de programação mesmo em ocasiões que os eletos não foram os esperados.	Sim, as ações programadas nls equipamentos utilizados não tinham uma resposta fiel aos comandos pré-programados, devido a imprecisão dos sensores nele presentes.	Não, gostaria de ter utilizados Kits que pudessem melhorar realmente meus conceitos tanto de lógica de programação como a programação em si, o Kits utilizado não ofereceu os resultados que eu esperava.	Ajudo a concretizar ainda mais meus conhecimentos, já que não vi nada totalmente diferente do que já havia visto anteriormente.	Nem sempre, diversas vezes a utilização das melhores lógicas não eram possíveis devido a falta de recursos proporcionado pelo Kit utilizado.
M.A.J.S.	Não, para mim a tarefa ficou clara e o nível de dificuldade exigido foi médio.	Sim, o objetivo da tarefa foi facilmente percebido para que a mesma pudesse ser realizada.	Sim, a única dificuldade apresentada se deve ao modo de como se programava utilizando os blocos; sendo que a utilização de códigos seria para mim mais entendível.	Sim, os blocos utilizados no programa eram claros em relação a quais ações seriam executadas no robô.	Sim, por motivos desconhecidos certas vezes um mesmo robô executava um mesmo código de diferentes maneiras, o que dificultava a programação do mesmo.	Sim, os resultados obtidos foram satisfatórios e me trouxeram um conhecimento numa área antes não tão conhecida por mim.	Ambos: ajudou na aquisição de novos conceitos de programação ligados a robôs e também ajudou a concretizar outros conceitos de programação já conhecidos por mim os utilizando também nos robôs.	Sim, os conceitos foram escolhidos adequadamente de forma a realizar a tarefa, porém o ambiente de programação era desfavorável a aprendizagem da programação para o LEGO.
J.J	Não. Não Houve nenhuma dificuldade, pois tudo foi extremamente repassado e nos ensinados para nos preparar pra tarefa que foi dada.	Sim, pois ele foi claramente a nós explicados antecipadamente, facilitando bastante.	Não, como já foi ressaltado as dificuldades não houve algumas dificuldades nos conceitos de programação ou de entender as tarefas.	O necessário, pois é claro que não aprendemos tudo sobre, mais o necessário pra matar a curiosidade e enxergar que não é nada tão longe da realidade como era minha mentalidade anteriormente.	Não, pois durante a nossa aprendizagem foi testado cada comando cada bloco o que fazia qual era o necessário para cada ocasião facilitando então quando foi feita a junção de conceitos.	Extremamente. Pois é algo que poucas pessoas tem expectativa de encontrar tal coisa em seu aprendizado, e por mim que fosse já seria satisfatório, mais pela maneira qual foi feita o tempo estendido.	Claro pois como qualquer programação, você tem de exercitar a sua lógica e aperfeiçoar-la mais logo ajudou na minha evolução.	Sim, em algumas horas não tinha algo padronizado para o tipo de programação o que nos libertava e nos deixava usar as diferentes lógicas o que por uma parte é ótima mais de certa forma acaba prejudicando os de menos lógica.

ANEXO VII – QUADROS DE AVALIAÇÃO DAS NORMAS

Alunos	Desviar de obstáculo	Seguir traçado	Encontrar traçado	Identificar faixa	Identificar falha no traçado
Grupo de Controle					
T.F	3	3	2	3	1
L.R.S	3	2	2	2	3
R.P.E	2	3	2	3	2
G.G.C.C.	2	3	3	3	1
I.R.O	3	2	3	2	2
T.R.P.	2	2	2	3	2
F.A.P.A.	2	3	1	2	3
A.A	3	2	2	2	2
Grupo 1					
C.A.S.R.	3	3	2	3	2
A.S.I	3	2	2	3	3
W.P	3	3	3	3	2
G.B.C	3	3	3	3	2
L.M.E.S	3	3	2	3	3
L.A.T.	3	3	1	3	2
V.L.	3	3	2	3	2
B.P.D.	3	2	2	3	2
Grupo 2					
L.C.F.	3	3	2	3	2
E.B.P.S	3	2	2	3	2
S.A.C.J.	3	2	3	3	3
R.J.C.F.	3	2	2	3	3
F.M.T	3	3	2	3	2
G.C.	3	3	2	3	2
W.Z.A	3	3	3	3	2
C.F.R	3	3	3	3	2
C.V.G.C.S	3	3	2	2	3
P.W.C.	2	2	1	3	2
V.A.C	3	3	2	3	3
J.M.P.F	2	3	1	3	1
R.G.C.J	3	2	1	3	3
J.G.S.G	3	2	2	3	3
W.M.C.	3	3	2	3	2
J.M.P.S	2	1	2	3	3
C.I.A	3	3	3	3	2
J.G.P.	3	2	2	3	2
B.H.T	3	3	3	3	1
A. G. R. S.	3	3	2	3	2
M.A.J.S.	3	2	2	3	1
J.J	3	2	2	2	2
L.M.G	3	2	2	3	2
C.R.M.S	3	3	3	3	2