

MESTRADO

Adriana da Silva Nogueira

**FILOSOFIA LOGO E LÓGICA DE
PROGRAMAÇÃO NO ENSINO SUPERIOR:
A TEORIA NA PRÁTICA**

2015



*Mestrado em Educação – Campus Centro I
Avenida Presidente Vargas 642, 22º andar – Centro
20071-001 Rio de Janeiro – RJ
Telefones: (21) 2206-9741 / 2206-9742*

ADRIANA DA SILVA NOGUEIRA

**FILOSOFIA LOGO E LÓGICA DE PROGRAMAÇÃO NO ENSINO
SUPERIOR:
A TEORIA NA PRÁTICA**

Dissertação apresentada à Universidade Estácio de Sá como requisito parcial para a obtenção do grau de Mestre em Educação.

Orientadora: Prof.^a Dr.^a Stella Maria de Azevedo Peixoto Pedrosa

**Linha de Pesquisa: Tecnologias da Informação e Comunicação nos Processos
Educativos**

**Rio de Janeiro
2015**

N778f

Nogueira, Adriana da Silva

Filosofia logo e lógica de programação no ensino superior : a teoria na prática. / Adriana da Silva Nogueira. – Rio de Janeiro, 2015.

105 f.

Monografia (Mestrado em Educação) –
Universidade Estácio de Sá, 2015.

1. Aprendizagem. 2. Lógica de programação. 3.
Filosofia Logo. I. Título.

CDD: 370

AGRADECIMENTOS

Em primeiro lugar, agradeço a Deus pelo dom da vida e por ser meu porto seguro em dias de maré revolta. Não foi fácil chegar até aqui, mas a certeza de que Ele estaria no controle de tudo me deu forças pra prosseguir.

Agradeço à minha família, em especial aos meus pais que me apoiaram nessa caminhada e me incentivaram a alçar voos mais altos. Agradeço pela paciência e carinho mesmo quando foi necessário privá-los da minha presença durante o tempo que me dediquei à esta pesquisa.

À Anderson de Assis Barros pelo seu apoio incondicional e por ter caminhado comigo até quando os pés não suportavam mais andar. Obrigada por não ter me deixado desistir.

Aos amigos, minha eterna gratidão pelo apoio e cuidado. Gostaria de agradecer, em especial, a duas pessoas que foram fundamentais. À Marta Salgueiro que mesmo há mais de quinhentos quilômetros de distância sempre esteve presente em todos os momentos. À Rafaela Lopes, que acompanhou cada momento me incentivando e me presenteando com sua amizade e carinho.

Ao corpo docente da UNESA que me apontou os caminhos para que eu chegasse até aqui.

Agradeço ao professor Alberto Tornaghi por ter estado presente no início da minha trajetória como pesquisadora e por ter sido mais que um mestre, um amigo de todas as horas.

Ao professor Alexandre Rosado por ter se dedicado à minha pesquisa sempre disposto a prestar o auxílio desejado.

À minha orientadora, professora Stella Pedrosa, por ter aceito o desafio de me orientar. Obrigada por todos os momentos intensos onde pudemos compartilhar nossos saberes. Obrigada por ouvir meus anseios e transformá-los em motivação para seguir em frente. Não tenho palavras para traduzir minha imensa gratidão.

Ao professor Márcio Lemgruber pelas discussões filosóficas em nossas aulas que me abriram os olhos e a mente para caminhar compassos mais firmes.

À professora Giselle Ferreira por ter me incentivado a não desistir e a prosseguir firme quando parecia ser impossível continuar minha trajetória.

Aos demais professores da Linha TICPE agradeço imensamente por todos os ensinamentos.

À Fundação Educacional Unificada Campograndense e à coordenação acadêmica por permitir realizar minha pesquisa e fornecer o apoio necessário à conclusão deste trabalho.

Aos meus alunos, alguns tão próximos que conheceram a trajetória desta pesquisa, agradeço por todo o apoio e carinho. Vocês fazem parte da minha história!

Aos colegas do curso, com os quais muitas vezes pude compartilhar as angústias e as alegrias, sempre caminhando de mãos dadas para alcançar nossos objetivos.

Por fim, à todos aqueles que contribuíram direta e indiretamente para a concretização deste trabalho, o meu muito obrigado.

“Os que se encantam com a prática sem a ciência são como os timoneiros que entram no navio sem timão nem bússola, nunca tendo certeza do seu destino”.

Leonardo da Vinci

RESUMO

Esta dissertação se desenvolveu dentro dos pressupostos metodológicos de natureza qualitativa. Foi realizada uma pesquisa-ação durante um semestre letivo que visava, com o uso de questionários e de testes diagnósticos, verificar se o processo de aprendizagem com base nos princípios da Filosofia LOGO contribui para a construção dos conhecimentos essenciais na área de Lógica de Programação. O estudo foi realizado junto a alunos do primeiro período dos cursos superiores de Bacharelado em Sistemas de Informação e Licenciatura em Computação em uma Faculdade da Zona Oeste da cidade do Rio de Janeiro. A Lógica de Programação e seus conceitos são pontos-chave em cursos ligados à computação. Como disciplina, seu conteúdo é ministrado no primeiro período. Seu aprendizado não é simples, pois depende das estruturas lógicas de pensamento construídas ao longo da vida. Também foi investigada a percepção dos alunos acerca do próprio processo de aprendizagem durante o período em que a disciplina foi ministrada. Observou-se que a Filosofia LOGO foi uma grande aliada, pois propiciou uma aprendizagem mais agradável, permitindo que os alunos, conscientes de seu ritmo de aprendizado, pudessem interagir com autonomia e, assim, gerenciar seu próprio processo de aprendizado. Ao término do trabalho são apresentadas algumas sugestões para ajustes na metodologia tradicional e na ementa proposta pela instituição. Com base na Filosofia LOGO, considera-se que o foco deva ser o desenvolvimento das estruturas lógicas de pensamento e não apenas no ensino de técnicas para a elaboração de programas computacionais.

Palavras-chave: Aprendizagem, Lógica de Programação, Filosofia LOGO

ABSTRACT

This research developed within the methodological assumptions of qualitative nature. An action research for a semester was held aimed at, with the use of questionnaires and diagnostic tests, verify that the learning process based on the principles of LOGO philosophy contributes to the construction of essential knowledge in Programming Logic area. The study was carried out among students of the first period of the courses of Bachelor of Information Systems and Graduation in Computer in a College of the West Zone of the city of Rio de Janeiro. The Logic Programming and its concepts are key points in courses related to computing. As a discipline, its content is taught in the first period. Your learning is not simple because it depends of logical structures of thought built over a lifetime. It was also investigated the perceptions of students about the learning process itself during the period when the discipline was given. It was observed that the LOGO Philosophy was a great ally because it gave a more pleasant learning, allowing students, aware of their learning pace, could interact with autonomy and thus manage their own learning process. At the end of the work are some suggestions for adjustments to the traditional method and the menu proposed by the institution. Based on the philosophy LOGO, it is considered that the focus should be the development of logical structures of thought and not just in teaching techniques for the development of computer programs.

Keywords: Learning, Logic Programming, LOGO Philosophy

LISTA DE TABELAS

Tabela 1: Ementa da disciplina Algoritmos e Programação I.....	30
Tabela 2: Principais comandos da Linguagem LOGO	47
Tabela 4: Análise do Aluno 1	73
Tabela 3: Análise do Aluno 5	74
Tabela 5: Análise do Aluno 7	74
Tabela 6: Análise do Aluno 9	75
Tabela 7: Análise do Aluno 10	76
Tabela 8: Análise do Aluno 16	77
Tabela 9: Análise do Aluno 17	77
Tabela 10: Análise do Aluno 21	78
Tabela 11: Análise do Aluno 23	78
Tabela 12: Análise do Aluno 26	79
Tabela 13: Análise do Aluno 28	80
Tabela 14: Análise do Aluno 31	80
Tabela 15: Análise do Aluno 33	81
Tabela 16: Análise do Aluno 3	82
Tabela 17: Análise do Aluno 13	83
Tabela 18: Análise do Aluno 18	84
Tabela 19: Análise do Aluno 19	84
Tabela 20: Análise do Aluno 20	85
Tabela 21: Análise do Aluno 22	86
Tabela 22: Análise do Aluno 25	86

LISTA DE FIGURAS

Figura 1: Representação em quatro fases do ciclo básico da investigação-ação.....	25
Figura 2: Tela inicial do Ambiente Super LOGO	47
Figura 3: Passos para criar um procedimento.....	48
Figura 4: Editor de procedimentos	49
Figura 5: Imagem produzida com o procedimento "castelo"	50
Figura 6: Resultado Final do Exercício	51
Figura 7: Tela inicial do SCRATCH	52
Figura 8: Construção de um quadrado.....	53
Figura 9: Agrupamento dos Comandos	53
Figura 10: Construção de um quadrado usando repetição.....	53
Figura 11: Tela Inicial do VisualG	56
Figura 12: Exercício com formas geométricas	62
Figura 13: Construindo um castelo.....	63
Figura 14: Resultado Final do Castelo	63
Figura 15: Tela do Jogo Megamania	68

LISTA DE GRÁFICOS

Gráfico 1 - Sexo do Público Alvo	27
Gráfico 2 - Faixa Etária do Público Alvo	27
Gráfico 3 - Você trabalha?	28
Gráfico 4 - Número de Horas Diárias Trabalhadas	29
Gráfico 5 - O LOGO contribuiu para as demais disciplinas que está cursando neste período?	65
Gráfico 6 - Você gostou de conhecer a Linguagem LOGO?	66
Gráfico 7 - O conteúdo do primeiro período aprimorou seu conhecimento anterior sobre programação?.....	67
Gráfico 8 - Categorias das Questões	70

SUMÁRIO

INTRODUÇÃO	13
CAPÍTULO 1	
A INSTITUIÇÃO E OS CURSOS	18
1.1. A Faculdade	18
1.2. Os Cursos	19
1.2.1. Licenciatura em Computação	19
1.2.2. Bacharelado em Sistemas de Informação.....	20
1.3. A disciplina	21
CAPÍTULO 2	
METODOLOGIA DA PESQUISA	24
2.1. Descrição dos Sujeitos da Pesquisa	26
2.2. Metodologia da Disciplina.....	29
CAPÍTULO 3	
A FILOSOFIA LOGO	34
3.1. O desenvolvimento da Inteligência segundo Piaget	34
3.2. O Construcionismo de Papert	36
3.3. Da Lógica de Programação à Filosofia LOGO.....	40
CAPÍTULO 4	
LINGUAGENS UTILIZADAS NA SALA DE AULA	45
4.1. A Linguagem LOGO	45
4.1.1. SuperLOGO	46
4.1.2. SCRATCH.....	52
4.2. Português Estruturado (Portugol)	54
4.2.1. VISUALG	55
CAPÍTULO 5	
RECORTES DO DIÁRIO DE CAMPO: O dia a dia das aulas.....	58
CAPÍTULO 6	
ANÁLISE DOS RESULTADOS	65
6.1. Elementos Marcantes	69
6.2. Análise dos Testes Diagnósticos.....	72
6.3. A Percepção do professor de Algoritmos e Programação II.....	87
CAPÍTULO 7	

CONSIDERAÇÕES FINAIS	89
REFERÊNCIAS	96
ANEXOS	98
Anexo 1 - Matriz Curricular Bacharelado em Sistemas de Informação	98
Anexo 2 - Matriz Curricular Bacharelado em Sistemas de Informação	100
Anexo 3 - Questionário 1	102
Anexo 4 - Questionário 2	103
Anexo 5 - Teste Diagnóstico 2	104
Anexo 6 - Teste Diagnóstico 2	105

INTRODUÇÃO

Minha trajetória com a computação começou aos 14 anos de idade, recém-saída de uma escola pública e ingressando em um curso técnico de processamento de dados de um colégio particular da zona oeste do Rio de Janeiro. Já no primeiro ano do ensino médio tive contato com lógica de programação e, embora tenha sido o início do meu aprendizado nesse novo território, as aulas eram demasiadamente engessadas e seguiam um padrão criado pela professora que lecionava tal disciplina.

Com dedicação aprendi as sequências de comandos e a estrutura rígida que não estava fundamentada em nenhuma bibliografia específica, porque se tratava da tradução de uma linguagem comumente utilizada como linguagem para ensino, o Pascal¹.

Não havia nenhuma relação do que era ensinado com a prática de programação, escrevíamos o código em nossos cadernos e recebíamos a correção, sempre em caneta vermelha. Num segundo momento passamos à linguagem Pascal, que pra nós se tratava de traduzir para o inglês os comandos já estudados.

O objetivo dos cursos técnicos era inserir os jovens profissionais no mercado, entretanto, eu e uma grande parte da turma sentíamos dificuldades em relacionar o que estávamos aprendendo com o que seria desenvolvido no mercado de trabalho.. Outra grande dificuldade foi no momento em que iniciamos o aprendizado com estruturas de repetição, pois tínhamos grandes dificuldades com a lógica em si. Como sempre fui apaixonada por matemática, lógica matemática e áreas afins, resolvi me aprofundar nos conhecimentos relacionados ao curso. Fiz alguns cursos livres de programação e, em um terceiro momento, começamos a trabalhar com uma linguagem visual que usava como base a linguagem Pascal e chamava-se Delphi.

Nesse período, comecei a ministrar aulas de informática em cursos livres, mas o conteúdo envolvia apenas sistemas operacionais e o pacote Office. Toda essa trajetória contribuiu para a escolha do curso superior, me fazendo optar por licenciatura em computação.

Esta etapa foi de suma importância. Embora o primeiro ano tenha sido considerado fácil para aqueles que, assim como eu, eram oriundos de cursos técnicos da área. Auxiliei vários colegas e percebi que muitos tinham receio de aprender e

¹ A linguagem de Programação Pascal recebeu este nome em homenagem ao matemático Blaise Pascal. Foi criada em 1970 por Niklaus Wirth, um suíço que desejava encorajar o uso de código estruturado. Segundo ele, Pascal foi criada simultaneamente para ensinar programação estruturada e para ser utilizada em sua fábrica de software.

acabavam se desmotivando. Esse medo de aprender é chamado por Papert de "matofobia" e, mais adiante, será abordado neste trabalho. A justificativa mais comum para as dificuldades era apontar o conteúdo como de difícil aprendizado e que não havia relação de teoria e prática. Essa relação dos alunos com as disciplinas dificultava ainda mais o aprendizado.

Minha trajetória como professora seguiu em turmas de ensino fundamental até que no último ano da faculdade comecei a lecionar para o curso técnico do colégio de aplicação da faculdade. Após a formatura, trabalhei como monitora de uma disciplina que contemplava o ensino de Delphi durante um ano, assim, eu lecionava para metade da turma e o professor titular para a outra metade.

No ano seguinte (2006) fui convidada pela coordenadora do curso a me tornar professora da instituição. Com a experiência de ministrar disciplinas de análise de sistemas e programação no ensino superior, comecei a perceber quais eram as lacunas existentes no processo de aprendizagem e as dificuldades mais pontuais alunos.

Anos depois, ao ingressar no mestrado em Educação da UNESA, na linha de pesquisa de Tecnologias, conheci o Professor Alberto Tornaghi que possuía uma significativa experiência com projetos envolvendo o LOGO, o que me levou a um novo contato com a linguagem. Minha experiência com esta linguagem havia sido apenas no primeiro segmento do ensino fundamental, mas ao perceber que se tratava de algo muito além de uma linguagem para criar códigos de programação, comecei a pensar nas possibilidades de usar sua aparentemente simples estrutura para reverter ou ao menos minimizar os problemas encontrados nas disciplinas que lecionava.

Ao refletir acerca da leitura do livro LOGO: Computadores e Educação de Seymour Papert, cheguei a conclusão que o que faltava no ensino das disciplinas básicas de programação era criar um ambiente favorável ao ensino onde o aluno pudesse "falar" a linguagem que estava aprendendo. Esse ambiente Papert (1988) denomina "matelândia".

A partir de então, começamos a discutir como a Filosofia LOGO poderia auxiliar no desenvolvimento das estruturas lógicas de pensamento de modo que resultasse em uma aprendizagem mais efetiva e que servisse de base para outras disciplinas. Assim nasceu esta pesquisa.

Inicialmente pretendíamos trabalhar com o método clínico de Piaget, adaptado à coletividade da sala de aula para verificar como a utilização da Filosofia LOGO

influenciaria no aprendizado de lógica de programação e como ocorre o desenvolvimento do conhecimento lógico-matemático e de lógica de programação em si. Almeida Neto et al. (2006) afirmam que o método clínico consiste em entrevistas individuais e acompanhamento dos estudantes durante a resolução de um problema elaborado segundo os critérios de resolução que se deseja. O método foi aplicado por Piaget com elementos concretos. No caso de elementos abstratos, como uma linguagem de programação, seria necessário, ainda segundo Almeida Neto et al. (2006), realizar adaptações no mesmo.

Collares (2001, p.74) afirma que “o desafio que este método impõe, em princípio, ao fazer pedagógico é o de se aprender a observar e a ouvir a criança para construir, a partir do que se vê e ouve, hipóteses de trabalho que deem consistência ao planejamento e atendam às necessidades dos alunos”. Em sua adaptação ao contexto escolar, ele elaborou um modelo dinâmico não estereotipado, que tivesse autonomia e particularidades que permitissem que fosse aplicado coletivamente, onde a intervenção docente assume características diferenciadas. O fato de que a observação obedece aos períodos impostos pelo tempo de aula e que nem todas as soluções podem ser desenvolvidas em um tempo específico, pode gerar interrupções ou intervalos consideráveis que dificultaria tal processo.

Devido ao fato de termos feito um planejamento inicial com base em um número específico de alunos e considerando a disponibilidade integral do laboratório de informática e, também, por restrições impostas pelo próprio ambiente acadêmico, o planejamento teve que ser adaptado. Assim, não foi possível trabalhar com o método clínico já que este exigiria uma observação mais detalhada e uma seleção de participantes para acompanhamento e registro do desenvolvimento dos mesmos.

Após essa constatação, com supervisão da minha orientadora, considerando as reais possibilidades do ambiente acadêmico, o planejamento inicial foi, em parte, reestruturado.

Assim, a presente pesquisa buscou investigar o desenvolvimento de estruturas lógicas de pensamento e dos conceitos de lógica de programação através da inserção da Filosofia LOGO na disciplina Algoritmos e Programação I, no primeiro período dos cursos superiores de Bacharelado em Sistemas de Informação e Licenciatura em Computação de uma universidade privada da zona oeste do Rio de Janeiro, identificando os conceitos de lógica de programação que os alunos trazem quando

iniciam o curso, observando o desenvolvimento das estruturas lógicas de pensamento na interação com a linguagem LOGO e sua Filosofia e os processos de solução de problemas envolvendo lógica de programação.

Os sujeitos da pesquisa foram alunos do primeiro período que ainda não tinham tido contato, no curso, com linguagens de programação. A pesquisa e as atividades, descritas detalhadamente na metodologia, foram aplicadas pela pesquisadora, docente da disciplina em questão.

Acompanhamos o desenvolvimento dos sujeitos e sua evolução durante o primeiro período e verificamos como se deu o processo e como tal proposta contribuiu para a aprendizagem de lógica de programação. Sendo assim foi possível investigar, quais conceitos de lógica de programação os alunos já trazem desenvolvidos quando iniciam o curso, se ocorreu o desenvolvimento de estruturas lógicas de pensamento em decorrência da interação com a linguagem LOGO, qual a percepção dos alunos acerca da utilização da Filosofia LOGO e, por fim, esta contribuiu para o desenvolvimento das estruturas lógicas de pensamento dos alunos envolvidos.

A primeira parte da pesquisa foi desenvolvida durante as aulas que ocorreram entre março e junho de 2014 e foram realizados testes diagnósticos para verificar, de acordo com suas estruturas lógicas de pensamento, como eram capazes de solucionar o problema proposto no início e no final do período. No segundo período do curso, mais precisamente no mês de setembro, foi feita uma pesquisa com os alunos com o objetivo de verificar a percepção que tiveram acerca da vivência.

Espera-se que este trabalho contribua para os cursos de formação de programadores quer sejam a nível superior, médio ou cursos livres, além de um olhar atento à formação de professores e desenvolvimento de suas habilidades e metodologias para o ensino das disciplinas relacionadas ao desenvolvimento do raciocínio lógico.

O presente trabalho está estruturado da seguinte forma:

No primeiro capítulo apresentamos a instituição na qual realizamos esta pesquisa, bem como os cursos de Bacharelado em Sistemas de Informação e Licenciatura em Computação e a disciplina Algoritmos e Programação I.

O Capítulo 2 trata da metodologia de pesquisa utilizada para compor este trabalho, a metodologia aplicada na disciplina e a descrição dos sujeitos da pesquisa e o perfil da turma pesquisada.

O terceiro capítulo é uma revisão bibliográfica do construcionismo de Papert e da Filosofia LOGO e no capítulo seguinte apresentamos as linguagens trabalhadas em sala de aula, a saber: a Linguagem LOGO com o uso do SuperLOGO e o SCRATCH e o Português Estruturado.

Os recortes do diário de Campo com as atividades desenvolvidas aula a aula são retratados no Capítulo 5.

O Capítulo 6 demonstra a Análise dos resultados obtidos nesta pesquisa e, por fim, o Capítulo 7 expõe as considerações finais, conclusões e sugestões para trabalhos futuros.

CAPÍTULO 1

A INSTITUIÇÃO E OS CURSOS

1.1. A Faculdade

A criação da Faculdade de Filosofia de Campo Grande, localizada na Zona Oeste do Rio de Janeiro, era algo um tanto quanto inusitado e tido como um contrassenso, pois se tratava de uma área rural com poucas perspectivas de crescimento.

Com o objetivo de criar na região uma instituição que fosse especializada em formação de professores e que se tornasse uma referência, Arthur Miécimo da Silva fundou a Faculdade de Filosofia de Campo Grande em 1961.

Foi um percurso longo até que fosse considerada realmente uma referência na formação de professores e passasse a ser conhecida como "a casa do professor" pela sua preocupação em fornecer uma formação de qualidade e por ter cursos voltados apenas para essa finalidade.

Vale ressaltar que a faculdade pertence à mantenedora Fundação Educacional Unificada Campo-Grandense (FEUC) que atualmente possui ainda a UNATIC (Universidade da Terceira Idade), Colégio de Aplicação Emmanuel Leontsinis e o Colégio Magali. Trata-se de uma fundação privada, sem fins lucrativos, mantida basicamente pelos recursos obtidos com os serviços que presta. A administração da FEUC e de suas mantidas é exercida por intermédio da Superintendência, responsável pela gestão do pessoal, dos recursos materiais e financeiros.

Em 2001 foram incluídos em seu rol de cursos Bacharelado em Sistemas de Informação e Licenciatura em Computação, dando origem a uma nova unidade chamada Faculdade de Campo Grande. Em 2005, após a aprovação definitiva desses cursos, as duas unidades se integraram formando as Faculdades Integradas Campo-Grandenses. Ao passar dos anos suas atividades foram sendo ampliadas e, atualmente, as Faculdades Integradas Campo-Grandenses também oferecem cursos de pós-graduação "lato sensu" nas áreas dos cursos de graduação.

Além de receber alunos do município do Rio de Janeiro, em especial da zona oeste, onde está situada, a instituição atende a municípios vizinhos, entre os quais:

Itaguaí, Mangaratiba, Seropédica, Angra dos Reis, Paracambi e os da Baixada Fluminense.

1.2. Os Cursos

Iniciados no primeiro semestre do ano de 2001, sob o regime seriado anual, os cursos de Bacharelado em Sistemas de Informação e Licenciatura em Computação, atualmente denominado Licenciatura em Informática, formaram suas primeiras turmas em dezembro de 2004.

A estrutura curricular original apresentava carga horária total de 3200 horas-aula, integralizada em 4 anos. Com respaldo nas resoluções CNE/CP 1 e 2, de 18 e 19 de Fevereiro de 2002 (BRASIL, 2002), o curso de Licenciatura em Computação passou a ter uma duração de 3 anos, totalizando 3000 horas-aula, enquanto que o curso de Bacharelado em Sistemas de Informação manteve sua carga horária.

No primeiro semestre do ano de 2005, a comissão designada pelo MEC\INEP avaliou os cursos e emitiu um parecer favorável para fins de reconhecimento do curso, conforme Portaria nº 4.443, de 22 de Dezembro de 2005.

No ano de 2013, os cursos passaram por uma reestruturação das grades curriculares com alteração de disciplinas, através da reavaliação do mercado de trabalho e o perfil do egresso, o que trouxe mudanças significativas como apresentado a seguir.

1.2.1. Licenciatura em Computação

O Projeto Pedagógico do Curso - PPC, os objetivos do curso de Licenciatura em computação são:

- "Tratar as áreas da computação e da educação como áreas formativas de competências, inter-relacionando-as;
- Colaborar para o desenvolvimento e formação integral do educando;
- Atuar no planejamento, desenvolvimento e aplicações de atividades docentes e prática investigativa em computação e educação;
- Participar de equipes de desenvolvimento de softwares educativos ou de materiais educacionais, bem como aplicar e avaliar produtos já existentes no mercado;

- Planejar e executar currículos e programas de capacitação profissional, em organizações diversas, que empreguem a Informática como suporte e apoio educativo;
- Elaborar e participar de projetos na área de Educação a Distância ou atividades educativas com a mediação de TI e Comunicação;
- Organizar, planejar e administrar laboratórios de informática para fins educacionais.
- Desenvolver o uso educacional efetivo das ferramentas computacionais, de maneira integrada a problemas em outros domínios de conhecimentos;
- Compreender o significado da computação no contexto brasileiro e regional, sendo capaz de se posicionar de forma crítica em relação aos desafios atuais da educação contemporânea;
- Compreender a dinâmica da sociedade e seus problemas socioculturais, contribuindo, assim, para a evolução da mesma." (FEUC, 2013, p.7)

Como perfil do Curso de Licenciatura em informática prevê que o concluinte "deverá ser um profissional que incorpore competências, saberes e habilidades de criatividade e inovação, de cooperação e de trabalho em equipe, de gestão e tomada de decisões, de aquisição e produção de conhecimentos, de expressão e comunicação, não sendo somente reprodutor de conhecimentos já estabelecidos." (FEUC, 2013, p.7)

1.2.2. Bacharelado em Sistemas de Informação

Como consta no Projeto Pedagógico do Curso de Bacharelado em Sistemas de Informação "tem como objetivo formar profissionais competentes, empreendedores e éticos, capazes de projetar, implementar e gerenciar a infraestrutura de TI necessária para o aprimoramento dos processos de uma organização, abrangendo tanto a parte lógica quanto física dos projetos necessários." (FEUC, 2014, p.7)

O PPC define ainda o perfil do egresso como o profissional que "atua diretamente com a avaliação, desenvolvimento, aplicação e gestão de tecnologias de informação, como instrumento de aprimoramento de processos organizacionais, departamentais e/ou individuais.

O perfil do egresso de um curso deve ser constituído por um conjunto de habilidades trabalhadas durante o curso de graduação, que permitem o desenvolvimento

das competências exigidas para o bom desempenho das funções que este poderá exercer no mercado do trabalho."(FEUC, 2014, p.7)

1.3. A disciplina

A disciplina a que se presta esse estudo é denominada Algoritmos e Programação I. Segundo o PPC, "as disciplinas de Algoritmos e Programação visam conceituar a modelagem de algoritmos para os alunos, bem como apresentar aos mesmos estratégias e métodos para a solução de problemas e expressá-los por meio de algoritmos para, serem transcritos numa linguagem de programação de alto nível." (FEUC, 2014, p.20)

São 3 disciplinas em sequência, distribuídas em períodos consecutivos que pretendem atingir os objetivos listados acima. Apresentamos a ementa² atual da disciplina de Algoritmos e Programação I na tabela a seguir.

NOME DA DISCIPLINA: Algoritmos e Programação I	Código: 0685
	Carga Horária: 60
OBJETIVOS: Desenvolver a capacidade de analisar e resolver problemas de complexidade limitada. Projetar e avaliar soluções em linguagem algorítmica. Conhecer a sintaxe dos primeiros comandos em uma linguagem pseudocódigo.	
EMENTA: Algoritmos: definição, exemplos e construção prática. Resolução de problemas através de computadores. Técnicas de elaboração e verificação de algoritmos. Refinamentos Sucessivos “top-down”. Estrutura básica de dados: tipos primitivos, variáveis, constantes, identificador. Declaração de variáveis. Operações básicas: comandos de atribuição e operadores. Comandos de entrada e saída. Estruturas de Controle: Sequencial, Condicional e de Repetição. Conceitos de funções e procedimentos.	

Tabela 1: Ementa da disciplina Algoritmos e Programação I

Como conteúdo programático foram desenvolvidas as seguintes Unidades, sendo que as unidades 1 e 2 foram trabalhadas apenas superficialmente, como conteúdo expositivo, pois apesar de representar um conteúdo importante na área, não tem relação direta com programação de computadores:

² Apresentamos a ementa no Capítulo que trata da metodologia da Pesquisa, porém optamos por repeti-la já que neste item são abordados os demais elementos relacionados com a disciplina

I - Introdução aos Computadores Eletrônicos Digitais.

- 1.1 – Conceitos básicos de Computação
- 1.2 – Unidade de Entrada;
- 1.3 – Unidade de Saída;
- 1.4 – Memória Primária (Principal);
- 1.5 – Memória Secundária (Auxiliar);
- 1.6 - Unidade Central de Processamento – UCP

II – Etapas para Elaboração de Programas

- 2.1 – Estudo do Problema;
- 2.2 – Criação do Algoritmo;
- 2.3 – Teste da Solução I;
- 2.4 – Codificação da Solução;
- 2.5 – Edição do Programa;
- 2.6 – Teste da Solução II;
- 2.7 – Documentação da Solução

III - Elementos Básicos do Programa.

- 3.1 – Constantes;
- 3.2 – Variáveis;
- 3.3 – Operações;
- 3.4 – Funções pré-definidas;

IV – Principais Comandos.

- 4.1 – Comando de Atribuição;
- 4.2 – Comando de Entrada;
- 4.3 – Comando de Saída.

V – Estruturas de Controle do Fluxo de Execução dos Comandos

- 5.1 – A Estrutura Sequencia.
- 5.2 – A Condicional – Simples e Composta.
- 5.3 – A Estrutura Caso
- 5.4 – As Estruturas de Repetição
 - 5.4.1 – PARA
 - 5.4.2 – ENQUANTO
 - 5.4.3 – REPITA ATÉ

VI – Procedimentos e Funções

- 6.1 – Criação e chamada de procedimentos
- 6.2 - Criação e chamada de funções

As disciplinas de *Algoritmos e Programação* (I, II e III) apresentam os elementos essenciais para a construção das estruturas lógicas de pensamento necessárias para as demais disciplinas relacionadas à programação.

No capítulo 2 apresentamos a metodologia de pesquisa que nos permitiu realizar este trabalho.

CAPÍTULO 2

METODOLOGIA DA PESQUISA

A presente pesquisa teve caráter qualitativo. Alves-Mazzotti e Gewansznajder (2000, p. 132), destacam que neste tipo de pesquisa o pesquisador é considerado como o principal instrumento de investigação, além da “necessidade de contato direto e prolongado com o campo, para poder captar os significados dos comportamentos observados”. Esse processo de captação deve ser natural e não intrusivo, visto que o pesquisador leva em consideração o comportamento das pessoas no ambiente de estudo. (BODGAN; BIKLEN, 1999).

A metodologia utilizada para o desenvolvimento desta pesquisa foi pesquisa-ação. Tripp (2005, p. 446) salienta a importância de reconhecermos a pesquisa-ação "como um dos inúmeros tipos de investigação-ação", termo usado para "qualquer processo que siga um ciclo no qual se aprimora a prática pela oscilação sistemática entre agir no campo da prática e investigar a respeito dela." Assim, o pesquisador é responsável pelo planejamento, implementação, descrição e, avaliação de mudanças que possam aprimorar sua prática, o que possibilita um constante aprendizado, no que se refere tanto à sua prática quanto ao processo da investigação.

Rizzini (1999) menciona que, na literatura, a pesquisa baseada na participação pode ser nomeada de diferentes formas: pesquisa-ação, pesquisa participativa, pesquisa participante e pesquisa-ação participativa. Para essa autora, ela é "um método ou ainda, uma estratégia de pesquisa concebida a partir de problemas vividos pela população pesquisada com a participação dos grupos sociais diretamente envolvidos com a problemática e em todo o processo de conhecer e transformar a realidade" (RIZZINI, 1999, p. 39). Portanto, a pesquisa-ação tem como característica a construção da pesquisa, desde a concepção do problema até a criação de soluções alternativas por aqueles que vivenciam o problema.

Baldissera (2012, p. 7), com base na proposta de Egg (apud Baldissera 1990), faz uma análise dos termos que compõe o nome pesquisa-ação:

Pesquisa ou investigação: é um procedimento reflexivo, sistemático, controlado e crítico que tem por finalidade estudar algum aspecto da realidade com o objetivo de ação prática;

Ação: significa ou indica que a forma de realizar o estudo já é um modo de intervenção e que o propósito da pesquisa está

orientado para a ação, sendo esta por sua vez fonte de conhecimento;

Participação: é uma atividade em cujo processo estão envolvidos os pesquisadores como os destinatários do projeto, que não são considerados objetos de pesquisa, mas sujeitos ativos que contribuem no conhecer e no transformar a realidade em que estão inseridos. [...] O modo de fazer o estudo, o conhecimento da realidade já é ação; ação de organização, de mobilização, sensibilização e de conscientização. A pesquisa-ação por ser participativa, supõe uma co-implicação no trabalho dos pesquisadores e das pessoas envolvidas no projeto onde se faz intercâmbio, socialização das experiências e conhecimentos teóricos e metodológicos da pesquisa.

Thiollent (1985, p.16) explica: “é necessário definir com precisão, qual ação, quais agentes, seus objetivos e obstáculos, qual exigência de conhecimento a ser produzido em função dos problemas encontrados na ação ou entre os atores da situação”. Trata-se de planejar e projetar o que será produzido a partir dos resultados que vão sendo obtidos durante toda a pesquisa.

Tripp (2005, p. 446) apresenta quatro fases do ciclo de investigação, conforme podemos observar na Figura 1. Tais fases podem ser divididas em: Ação, que inclui Planejar e Agir, e, investigação que inclui Descrever e Avaliar.

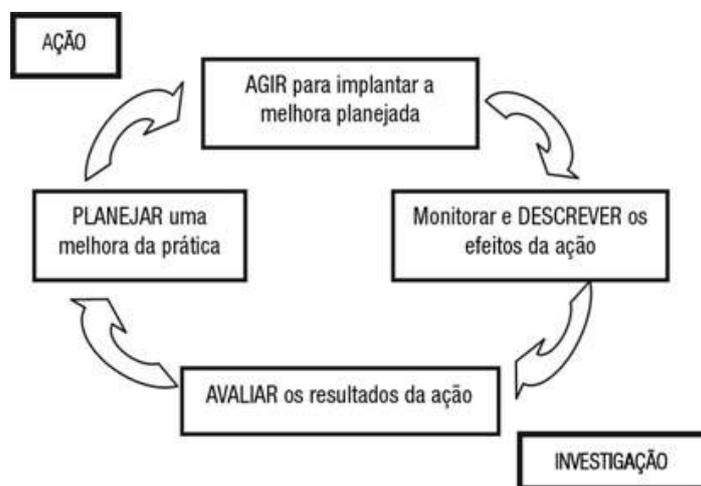


Figura 1: Representação em quatro fases do ciclo básico da investigação-ação.

Fonte: TRIPP, 2005, p. 446

A partir da prática da pesquisadora foi realizado um planejamento visando melhorá-la. O planejamento foi realizado, conforme descrito mais adiante, monitorado com objetivo de descrever os efeitos de tal transformação e nos permitiu avaliar os resultados da ação. Comprendemos que não é possível encerrar o ciclo ao final desta

pesquisa, mas que o constante aprimoramento das ações para melhorar o processo de ensino-aprendizagem é fundamental para atingirmos os objetivos propostos pelo ensino de algoritmos.

Para a realização desta pesquisa, foram acompanhados o desenvolvimento dos sujeitos e sua evolução durante o curso. Também verificou-se como se deu o processo e como tal proposta contribuiu para a aprendizagem de lógica de programação.

A pesquisa foi desenvolvida durante as aulas da disciplina denominada *Algoritmos e Programação I*, ministrada no turno da noite, pela pesquisadora em questão.

O cenário para o qual foi feito o planejamento foi para uma turma com aproximadamente 30 alunos, que deveria ser alocada em um laboratório com capacidade semelhante aos alunos da turma, tendo cada computador os programas SCRATCH e SuperLOGO instalados, onde as aulas semanais seriam semanalmente ministradas. O cenário real foi uma turma com aproximadamente 50 alunos, alocados em um laboratório, porém só poderíamos utilizar o espaço quinzenalmente, além de termos disponível neste espaço apenas 20 máquinas. Cerca de 5 ou 6 alunos traziam seus notebooks, mas devido ao espaço físico alguns não tinham sequer contato com o computador.

2.1. Descrição dos Sujeitos da Pesquisa

O público alvo desta pesquisa são os 34 alunos que no primeiro período dos cursos de bacharelado em Sistemas de Informação e Licenciatura em Computação tiveram contato com a Filosofia LOGO e que permaneceram no segundo período. Para complementar as informações obtidas no primeiro período, foi aplicado um questionário (Anexo 4) quando esses alunos cursavam o segundo período, ou seja, após a mudança da metodologia de ensino.

Para realizar a análise dos dados, foram tabuladas todas as respostas previamente organizadas uma planilha, gerados gráficos com as questões objetivas e separadas as respostas discursivas em categorias não exclusivas. Tal divisão nos permitiu uma melhor análise e compreensão do que os alunos disseram.

Dos alunos que participaram da pesquisa e permaneceram no curso, 30 são do sexo masculino e apenas 4 do sexo feminino, conforme apresentado no Gráfico 1. Esse perfil é tradicional em cursos da área de tecnologia.

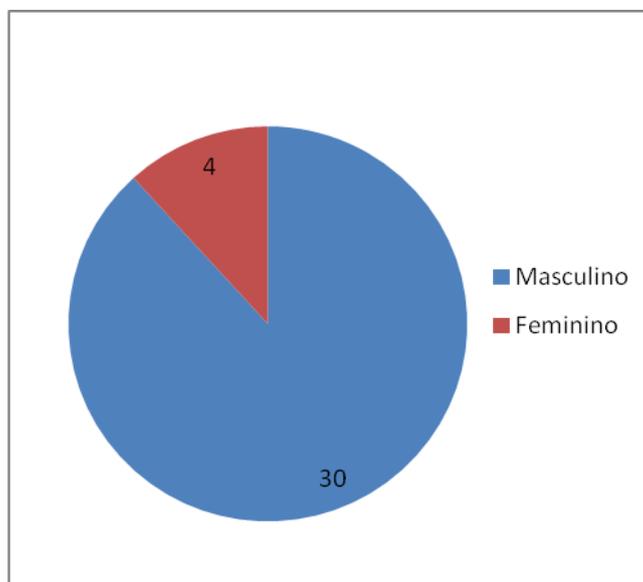


Gráfico 1 - Sexo do Público Alvo

Em geral, destaca-se um público jovem nos cursos de computação, visto que muitos ingressam nestes cursos imediatamente após concluírem o ensino médio ou pouco tempo após. Tais cursos são considerados atrativos para o mercado de trabalho, já que as empresas de tecnologia tem ganhado cada vez mais espaço. De fato, podemos observar no gráfico 2, mais da metade dos alunos possuem entre 21 e 30 anos.

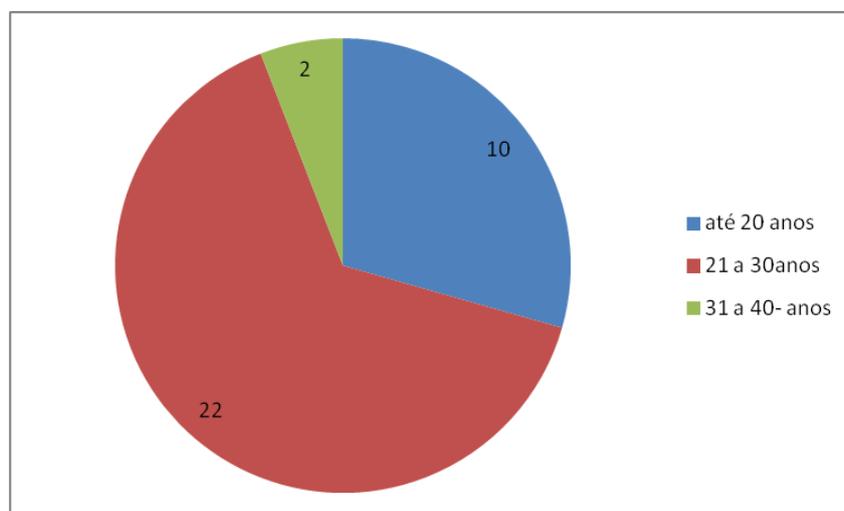


Gráfico 2 - Faixa Etária do Público Alvo

Como se trata de um curso noturno, buscamos informações sobre a residência e a atividade laboral destes alunos, sobre a região onde trabalham e, ainda, onde residem para que tivéssemos noção do seu deslocamento diário.

Verificamos que a maioria (dezenove) reside em Campo Grande, bairro onde está localizada a instituição de ensino, apenas 3 residem em cidades vizinhas, a saber, Itaguaí, Nova Iguaçu e Angra dos Reis) e os demais em outros bairros da zona oeste do Rio de Janeiro. Dos residentes no bairro de Campo Grande, 6 trabalham no mesmo bairro. Os alunos que residem em Itaguaí e Angra dos Reis trabalham em sua própria cidade, ficando o deslocamento maior para o itinerário do trabalho até a faculdade.

Cinco dos alunos afirmam que suas atividades não estão centralizadas em apenas um bairro, mas em toda a cidade do Rio de Janeiro. Estes são residentes de bairros da Zona Oeste, localizados próximo à instituição.

Conforme apresentado no gráfico 3, aproximadamente 29 alunos trabalham e apenas 5 não possuem atividades laborais. Menos da metade trabalha no próprio bairro de Campo Grande.

Ainda sobre as atividades laborais, dos 17 que trabalham com computação apenas um é do sexo feminino. A área de atuação desses alunos é suporte e infraestrutura, nenhum dos 34 alunos atua na área de desenvolvimento de software.

Onze declararam que não trabalham com nenhuma atividade relacionada à computação e uma aluna atua como professora de informática educativa no ensino fundamental.

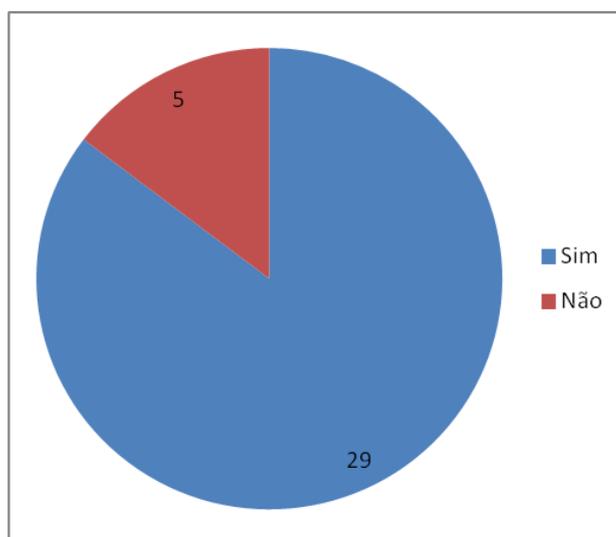


Gráfico 3 - Você trabalha?

A média de horas trabalhadas por esses alunos está acima das 8 horas e meia diárias. Como mostra o gráfico 4, mais da metade dos alunos que declararam trabalhar, cumpre uma jornada de 8 horas, sendo que 12 cumprem de 9 a 12 horas diárias.

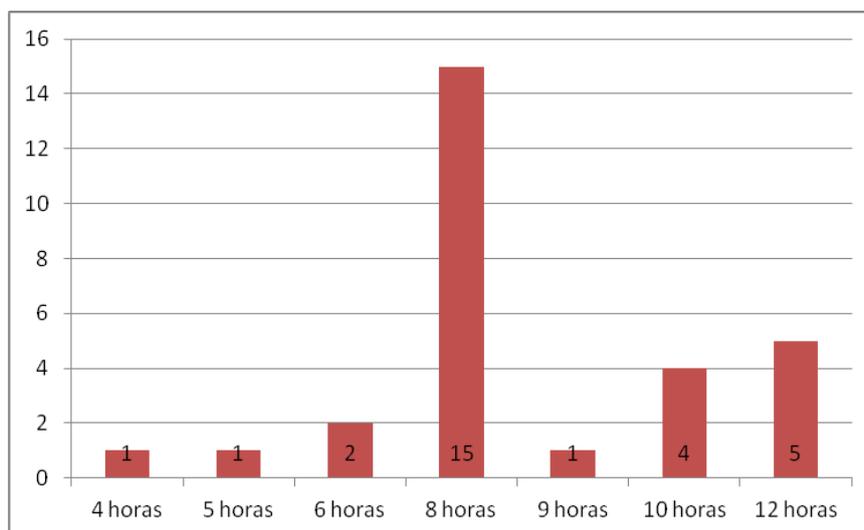


Gráfico 4 - Número de Horas Diárias Trabalhadas

2.2. Metodologia da Disciplina

Uma das constatações feitas pela pesquisadora - enquanto professora de diferentes disciplinas de programação - era a dificuldade dos alunos em aprender e manter os conhecimentos adquiridos, já que se trata de conceitos fundamentais para a vida profissional e acadêmica do aluno.

Com 4 tempos semanais³, as aulas eram expositivas e a prática na maioria das aulas do curso eram "no papel", ou seja, os alunos aprendem conceitos de lógica de programação e conceitos de programação de computadores, praticando no caderno, ou seja, sem ver o funcionamento. Somente nas últimas aulas que lhes era dado alguma prática. Sendo assim, possuíam apostilas com o conteúdo a ser lecionado e listas de exercícios que eram discutidas e trabalhadas/corrigidas pelo professor em sala de aula.

Como exposto anteriormente, a turma foi submetida a uma pesquisa-ação, o que acarretou uma revisão da metodologia da disciplina com a introdução de linguagens de

³ O turno da noite apresenta 40 minutos por tempo de aula, já o turno da manhã 50 minutos.

programação não comerciais – o SuperLOGO e o SCRATCH⁴ – que foram idealizadas especialmente para a aprendizagem e o desenvolvimento de estruturas lógicas de pensamento.

A tabela 1 apresenta a ementa da disciplina desenvolvida pela coordenação em conjunto com o corpo docente dos cursos, não havendo qualquer intervenção ou diferenciação para essa pesquisa. Cabe ressaltar que as ementas não são fechadas, ficando a cargo da instituição defini-las a partir das diretrizes para os cursos superiores.

NOME DA DISCIPLINA: Algoritmos e Programação I
OBJETIVOS: Desenvolver a capacidade de analisar e resolver problemas de complexidade limitada. Projetar e avaliar soluções em linguagem algorítmica. Conhecer a sintaxe dos primeiros comandos em uma linguagem pseudocódigo.
EMENTA: Algoritmos: definição, exemplos e construção prática. Resolução de problemas através de computadores. Técnicas de elaboração e verificação de algoritmos. Refinamentos Sucessivos “top-down”. Estrutura básica de dados: tipos primitivos, variáveis, constantes, identificador. Declaração de variáveis. Operações básicas: comandos de atribuição e operadores. Comandos de entrada e saída. Estruturas de Controle: Sequencial, Condicional e de Repetição. Conceitos de funções e procedimentos.

Tabela 1: Ementa da disciplina Algoritmos e Programação I

Com base no exposto, criamos uma lista de competências a serem desenvolvidas durante a realização da pesquisa.

- Compreender o conceito de variável.
- Conhecer os tipos de dados existentes e manipuláveis através de linguagens de programação.
- Usar comandos de leitura e impressão (comandos de impressão representam comandos que permitem escrever na tela textos informativos ou resultados do processamento.
- Manipular variáveis (realizar cálculos, atribuir valores).
- Conhecer e usar as estruturas condicional ou estruturas de decisão.
- Compreender o que é uma proposição
- Usar os conectivos lógicos E e OU e os operadores relacionais
- Conhecer e aplicar as estruturas de repetição PARA, ENQUANTO e FAÇA...ENQUANTO na solução de problemas.

⁴ O Capítulo 4 apresenta das linguagens utilizadas em sala de aula.

- Conhecer e usar matrizes unidimensionais e bidimensionais.

Conforme dito anteriormente, trabalhamos com uma turma de aproximadamente 50 alunos, alocada em um laboratório com capacidade reduzida e que só poderia ser usado quinzenalmente. Este fato fez com que alguns alunos não tivessem acesso à prática durante as aulas

No primeiro contato com a turma foi feita uma diagnose a partir de perguntas orais sobre o nível de conhecimento que os alunos possuíam. Verificou-se tratar de uma turma heterogênea, pois alguns alunos haviam cursado ensino médio técnico na área ou áreas relacionadas, enquanto outros haviam feito cursos livres e outros, ainda, não possuíam qualquer conhecimento.

Na aula seguinte, foi aplicado um primeiro questionário (Ver anexo 1), onde os alunos puderam expor mais claramente essas informações, que foram confrontadas com o primeiro teste diagnóstico, a fim de verificar se o que foi relatado condizia com a capacidade de estruturação de soluções que eles possuíam.

O primeiro teste diagnóstico buscou reconhecer a capacidade inicial de estruturação de soluções e verificar quais estruturas lógicas de pensamento os alunos já traziam e quais conseguiam aplicar na solução do problema proposto. Foi solicitado aos alunos que descrevessem a solução de forma simples, sem o uso de nenhuma pseudolinguagem⁵. O teste trazia questões simples relacionadas ao cotidiano, sendo que a primeira questão exigia uma solução simples e a segunda demandava um pouco mais de estruturação lógica para sua resolução.

No final do período, os alunos foram submetidos a outro teste, com questões equivalentes ao do primeiro, porém com maior grau de complexidade e que visaram medir sua evolução diante da metodologia aplicada.

Durante as aulas trabalhamos com a Filosofia LOGO e com atividades específicas objetivando inicialmente promover a ambientação na ferramenta de programação e (SuperLOGO). Em seguida, foram propostos desafios que exigiam o uso dos conceitos de lógica descritos a seguir e que permitiam avaliar a forma de estruturação da solução. As soluções do primeiro teste foram comparadas com as do segundo a fim de investigar se houve desenvolvimento e ampliação das estruturas. Os

⁵ Pseudolinguagem é considerada em programação uma linguagem fictícia e intermediária entre a linguagem humana e a linguagem de programação.

desafios foram elaborados para estimular os alunos a buscarem a resolução de problemas quer individualmente, quer com a ajuda dos colegas, estando conscientes do seu ritmo de aprendizado e de sua evolução.

Os principais conceitos de lógica de programação trabalhados foram: sequência lógica; conceitos de variáveis e constantes; operadores relacionais lógicos e aritméticos; estruturas de seleção e repetição; e funções/procedimentos.

A turma conheceu a proposta de trabalho no primeiro dia, mostrando-se interessada e participativa. Por ser uma turma de 1º período, os integrantes não se conheciam, mas pelo próprio perfil que apresentavam, interagiram muito bem. Foi feita a sugestão de que criassem um grupo no Facebook para troca de experiências e de informações entre eles. O grupo criado não pertenceria à disciplina, mas à turma, que interagiu sem a interferência da pesquisadora que apenas observava e, quando necessário, compartilhava o material das aulas. Esse espaço se configurou como um ambiente de compartilhamento e interação, busca e troca de soluções. Desde no início das aulas, enquanto tratávamos de conceitos teóricos sobre computação, eles já interagiam entre si.

A proposta inicial da pesquisa era apenas utilizar os programas que tivessem por base a linguagem LOGO, mas ao apresentar o conteúdo e o programa da disciplina, os alunos tomaram conhecimento de que estudariam o conteúdo em pseudocódigo/português estruturado, o que nada mais é que uma linguagem fictícia que permite a programação/construção de programas sem obedecer à estrutura rígida de uma linguagem de programação. Tal linguagem é utilizada para que seja focada a lógica de programação e não uma linguagem propriamente dita.

Parte da turma possuía conhecimento de tal conteúdo e do próprio programa que começaram a utilizar, o VisualG, que permite executar o código escrito em pseudocódigo. Sendo assim, aqueles que não conheciam a ferramenta passaram a conhecê-la por intermédio e iniciativa dos próprios alunos o que nos fez incorporá-la às aulas.

Nossa metodologia de trabalho estava apoiada na Filosofia LOGO que, em síntese, visa apresentar o ambiente de programação como um “lugar” onde o aluno estará no controle da aprendizagem, e “ensinará o computador a pensar”. Desse modo, ele “embarca numa exploração de como ele mesmo pensa” (PAPERT, 1988, p. 35), experiência e reflexão que muitos que chegaram à fase adulta não tiveram durante seus

estudos. Portanto, todas as aulas buscaram criar um ambiente de descoberta e aprendizado onde o aluno pôde assumir o controle de seu ritmo de aprendizagem e trabalho, mesmo quando o foco eram outras linguagens.

Vale ressaltar que desde o primeiro momento não enfatizamos a linguagem LOGO, inclusive porque não poderíamos alterar a ementa do curso e uma parte das soluções geradas a partir dos conteúdos ministrados não poderiam ser testados diretamente.

Após a realização da pesquisa, já com os alunos cursando o segundo semestre de seus respectivos cursos, aplicamos um novo questionário com o objetivo de verificar a percepção individual sobre a experiência que tiveram no primeiro período. Tal questionário possuía questões abertas e fechadas, nos permitindo uma visão mais ampla de como os alunos se perceberam antes e após a experiência.

O professor da disciplina seguinte Algoritmos e Programação II também foi entrevistado para que pudesse relatar sua percepção acerca das características apresentadas pela turma após esta experiência.

No próximo capítulo será aprofundada a relação da Lógica de Programação com a Filosofia LOGO.

CAPÍTULO 3

A FILOSOFIA LOGO

3.1. O desenvolvimento da Inteligência segundo Piaget

Segundo Piaget (1972), ao agir, o sujeito desenvolve continuada e progressivamente sua inteligência. O “[...] conhecimento não procede, em suas origens, nem de um sujeito consciente de si mesmo, nem de objetos já constituídos (do ponto de vista do sujeito) que a ele se imporiam. O conhecimento resultaria de interações que se produzem a meio caminho entre os dois [...]” (PIAGET, 1972, p.14).

Piaget separa o processo cognitivo inteligente em: aprendizagem e desenvolvimento. Para o autor a aprendizagem refere-se à aquisição de uma conclusão ou solução de um problema que é particular ao aprendiz e que foi aprendida em função da experiência, seja ela sistemática ou não, enquanto que o desenvolvimento é “uma equilibração progressiva, uma passagem contínua de um estado menor de equilíbrio para um estado de equilíbrio superior” (PIAGET, 1998, p. 13), ou seja, o desenvolvimento seria a aprendizagem de fato, sendo o aprendiz o responsável pela construção dos conhecimentos (MACEDO, 1994).

Por meio das ações, o conhecimento se interioriza e se transforma progressivamente. Tais processos são denominados assimilação e acomodação, e são definidos como pressupostos para a construção do conhecimento. (PIAGET, 1972)

A assimilação está relacionada à ação do sujeito sobre um objeto, através da qual ele incorpora novas experiências ou informações às já existentes. A acomodação é um processo através do qual o sujeito modifica suas estratégias de ação, ideias e conceitos, em função das novas informações/experiências adquiridas, gerando, portanto, novas estruturas cognitivas. Entre a assimilação e a acomodação há o processo de equilibração, caracterizado como adaptação, constituindo-se como um dinamismo do desenvolvimento cognitivo. Tais processos são inconscientes e ocorrem em diferentes níveis, seja pelo aprendizado de um assunto novo ou a evolução da aprendizagem no que se refere à passagem de um conhecimento mais simples para um mais complexo.

Quando falamos de desenvolvimento e processo de construção do conhecimento, não podemos deixar de abordar os estágios propostos por Piaget. Ele apresenta seis

estágios ou períodos de desenvolvimento (PIAGET, 1998, p.15). Os três primeiros estágios estão compreendidos entre o nascimento e até, no máximo o segundo ano de vida, denominado período de lactância, momento no qual a criança ainda não desenvolveu a linguagem e o pensamento. O primeiro seria chamado por ele de estágio dos "reflexos ou mecanismos hereditários", onde a criança apresenta os primeiros instintos e as primeiras emoções. No segundo, é possível notar os primeiros hábitos motores, a organização das percepções e a diferenciação dos sentimentos. No terceiro, é desenvolvida a inteligência sensório-motora ou prática, que antecede a linguagem, a criança já é capaz de demonstrar o que deseja, mas ainda não é capaz de nomear coisas; nesta etapa também é possível demonstrar afetividade. A inteligência é prática e noções de espaço e tempo são construídas pela ação.

O quarto estágio está compreendido entre os dois e sete anos e corresponde a segunda parte do que chamamos "primeira infância". Neste estágio a criança tem o que o autor chama de "inteligência intuitiva", é egocêntrica e não compreende nada sem haja alguma explicação (fase dos "porquês"), além disso, nas relações sociais é a fase de submissão ao adulto.

No quinto estágio, que compreende dos sete até onze ou doze anos, começa o desenvolvimento da lógica e dos sentimentos morais e sociais de cooperação; nesse momento a criança já é capaz de realizar abstração, mas ainda depende do mundo concreto para chegar a esta.

No sexto estágio, que compreende o período da adolescência até a fase adulta, já é possível realizar abstração, desenvolver e testar hipóteses na resolução de problemas e, ao final desta, já é possível aplicar o raciocínio lógico na resolução de qualquer problema. Esse é também um momento de formação da personalidade e inserção afetiva e intelectual na sociedade adulta. Na presente dissertação nos atemos a esse sexto e último estágio.

Com base no apresentado, supõe-se que os adultos possuam as estruturas das operações intelectuais abstratas que levam ao uso da lógica desenvolvidos. Entretanto, cabe ressaltar, que embora espere-se que cada fase ocorra de acordo com a idade do indivíduo, percebe-se que nem sempre isso acontece nos períodos previstos. Portanto, não é possível afirmar que o indivíduo ao chegar à idade adulta tenha todas as suas estruturas desenvolvidas, visto que alguns desenvolvem algumas destas que outros jamais chegarão a desenvolver integralmente.

Cabe ainda observar, que os estudos de Piaget fundamentam o construtivismo, cuja perspectiva não separa o processo de construção do conhecimento, do processo de constituição e desenvolvimento do sujeito na sua interação com o mundo.

Esse sujeito se define como tal a partir do momento em que se constitui junto com o objeto do conhecimento, que não é apenas, nem necessariamente, físico. Dessa forma, falar em construção do conhecimento significa falar ao mesmo tempo em construção do sujeito que conhece e do objeto a ser conhecido.

O termo construtivismo faz referência à relação entre a estrutura e o processo, assim sendo, a construção do conhecimento inclui tanto a construção do sujeito quanto a do objeto a ser conhecido, aparecendo, ambos, “como resultado de um processo permanente de construção” (Coll, 1987, p.186), o que permite que as estruturas sejam construídas pela relação estabelecida entre eles, sendo a interação mediada pela ação. “Conhecer não consiste, com efeito, em copiar o real, mas em agir sobre ele e transformá-lo” (Piaget,1973, p.15)

3.2. O Construcionismo de Papert

Papert e Piaget trabalharam juntos entre 1958 a 1963, momento em que Papert se interessou pelo construtivismo de Piaget que, posteriormente, foi utilizado como embasamento para o construcionismo.

Papert (1988, p.188) conceitua aprendizagem da mesma forma que Piaget no que se refere “a uma aprendizagem espontânea, natural” se e somente se houver interação com o ambiente e com o outro, o que fundamenta teoricamente investigações relacionadas às práticas pedagógicas embasadas no construtivismo.

Assim como Piaget (1982), Papert (2008) menciona que o conhecimento não pode ser transmitido ou transferido, mesmo quando parece que isso está sendo feito. As informações não podem ser incorporadas nas estruturas mentais já “digeridas”. Se fosse possível observar os processos cerebrais, veríamos que o interlocutor está criando uma versão pessoal do que lhe é transmitido.

Papert (2008, p. 146), considera três grandes estágios, a lactância, que seria a primeira parte da primeira infância e que ele chama fase pré-lógica, a segunda parte da

primeira infância, onde há lógica mas de forma concreta, e, por fim, a adolescência "onde o pensamento é dirigido e disciplinado por princípios de lógica (...)".

Ele dedicou seu trabalho à crianças no segundo grande estágio, o das operações concretas e, como ele cita, "Em termos de lógica, a criança trará para uma determinada situação um modo de pensamento que poderá ser muito diferente do modo usual (...)" (PAPERT, 2008, p. 147). O trabalho a ser realizado nesse estágio é o de "desenvolver as entidades mentais relevantes e dar-lhes conexões para que as distinções tornem-se significativas." Ao mencionar a questão da distinção está se referindo ao conteúdo verbal e não verbal.

Através do construcionismo somos convidados a "examinar mais de perto a construção mental" e o papel da escola nesse contexto (PAPERT, 2008, p.137), enfatizando a possibilidade de se aprender de forma diferente, não tanto como a de *ensinar* de forma diferente. Esta diferenciação é importante, pois embora considere a importância do ensino, ele prioriza a aprendizagem.

Papert (2008) aborda o construcionismo como uma filosofia que não põe em cheque o valor da instrução, mas tem-se como meta "a maior aprendizagem a partir do mínimo de ensino" (PAPERT, 2008, p.134) e afirma que a atitude construcionista é indispensável pelo seu caráter minimalista. Com base no exposto, o professor deve dar ao aluno as ferramentas necessárias para que ele construa o conhecimento e se sinta impulsionado a buscar as respostas que necessita. Pretende-se, assim, oferecer atividades agradáveis, que partam de algo concreto, a partir de experiências do *mundo real*, para que se aprenda *fazendo* de modo melhor do que anteriormente.

O foco do trabalho de Papert está na *aprendizagem*. Ele considera que não se deve disponibilizar informações ou conteúdos para o aluno, mas criar situações favoráveis à construção do conhecimento. Assim, a aprendizagem significa o processo de construção do conhecimento (processo) com o desenvolvimento de um produto pelo aluno, com o intermédio do professor (PAPERT, 1988).

Ele apresenta, então, o computador como uma máquina capaz de transformar a aprendizagem pela forma como aprendemos a utilizá-la, mencionando que é possível que uma criança torne-se um "programador", considerando-se que a programação nada mais é do que a comunicação com o computador por meio de uma linguagem que ele é capaz de compreender (PAPERT, 1988, p.18).

Papert (1988, p.59) sugere, ainda, a utilização dos computadores para romper as barreiras entre as áreas humanas e científicas, plantando sementes que gerariam “uma cultura epistemológica menos dissociada [...]”; ele menciona o fato de crescermos em uma sociedade onde se acredita que existam pessoas espertas e outras menos capazes, onde “tudo é preparado para que as crianças atribuam suas primeiras experiências de aprendizagem desagradáveis ou malsucedidas à sua própria inabilidade” (PAPERT, 1988, p. 63). Essas ideias vão se consolidando a cada fracasso e fazem com que os alunos acreditem que são mais ou menos capazes de aprender determinado conteúdo.

O uso do computador como uma “máquina de ensinar”, ou seja, replicando os modelos tradicionais de ensino é denominado como instrucionismo. Habitualmente o instrucionismo é compreendido como a utilização do computador no processo educacional como suporte, reforço ou complementação do trabalho que é feito pelo professor em sala de aula (Valente,1993).

Na concepção do autor o instrucionismo é a "crença de que o caminho para uma melhor aprendizagem deve ser o aperfeiçoamento da instrução" (PAPERT, 2008, p.134). Sua ideia parte do princípio de que para que tenhamos uma educação melhor, a instrução precisa ser aperfeiçoada. Assim, o uso de computadores no ensino deve ter como meta que os computadores *façam a instrução*.

Percebemos, assim, que através do construcionismo, Papert se propõe a inserir as teorias de construção do conhecimento propostas por Piaget em um ambiente computacional. A programação de computadores permite o processo de assimilação, onde o aluno conhecerá conceitos inerentes à linguagem de programação e, em seguida, o momento de acomodação, por meio de revisão de estratégias, solução de problemas que poderiam ser reelaboradas em níveis de dificuldade e compreensão diferentes, passando do mais simples ao mais complexo.

As cinco dimensões base do Construcionismo são assim apresentadas por Rosa (2004, p.46 e 47):

Dimensão Pragmática constitui um caráter prático, o aprendiz nota que está aprendendo algo que ele pode usar e que realmente possui utilidade. Além disso, percebe que a utilização do produto construído não demorará a ocorrer.

A **Dimensão Sintônica**, a segunda dentre as cinco, é a que apresenta a característica que permite que o produto construído seja personalizado, ou seja, o projeto e o aprendiz permanecem em sintonia desde a elaboração até a finalização do mesmo. Isso favorece a conceitualização por parte do aluno, que desenvolve algo com que ele se identifica, ou seja, um projeto pessoal.

A terceira dimensão é a **Sintática**, a qual nos remete à questão do aprendiz facilmente poder usar os elementos do ambiente educacional, progredindo gradualmente nesse ambiente, de acordo com suas necessidades e seu próprio desenvolvimento cognitivo. O aluno manipula os elementos, na forma ideal, sem precisar previamente de outros conhecimentos para isso. No caso de um *software*, a interface deve ser de fácil manipulação.

A **Dimensão Semântica** é o quarto princípio apresentado. Trata de uma característica que o ambiente de aprendizagem deve apresentar no que se refere às situações reais do mundo, ou mesmo concretas (que façam sentido ao aluno) e que tenham relação com os conceitos a serem construídos. É de grande importância que o aluno consiga atribuir significado ao que está construindo, da mesma forma que possibilite a descoberta de novas conjecturas e não apenas trabalhe com formalismos e signos que o levem a uma abstração, que não possui qualquer nexos com sua realidade.

Finalmente, a última dimensão é a **Dimensão Social**, que engloba a integração do desenvolvimento da atividade com as relações pessoais e cultura do aprendiz. Realmente, desenvolver um ambiente no qual o aprendiz busca relações sociais e que, conseqüentemente, revele que tal envolvimento humano, durante uma atividade, facilita muito a produção de significado para o aluno, em relação ao que ele esteja desenvolvendo, é fato de grande importância para a aprendizagem desse.

Ao se pensar em programação de computadores tais dimensões são identificadas claramente já que partimos de um problema com aplicação real, quer faça parte do cotidiano do aluno ou não. Assim, ao estruturar uma solução, ele verifica sua utilidade dentro do escopo do que está aprendendo ou do que deseja aprender, torna tal atividade um projeto pessoal, seleciona os elementos ou busca por elementos que possam compor sua solução e a constrói de forma significativa buscando, na maioria dos casos, buscando interação entre os colegas da turma para compartilhar seus resultados ou buscar por elementos que complementem sua solução.

Quando se deseja que o sujeito seja ativo no processo de construção do conhecimento, ele deve participar de todo o processo de forma consciente e interagindo seja com a máquina, com seus pares ou com o próprio conteúdo que deseja assimilar. Caso contrário, cria-se apenas um mero reprodutor. Como afirma Piaget (1978, p. 176):

[...] fazer é compreender em ação uma dada situação em grau suficiente para atingir os fins propostos, e compreender é conseguir dominar, em pensamento, as mesmas situações até poder resolver os problemas por elas levantadas, em relação ao porque e ao como das ligações constatadas e, por outro lado, utilizadas na ação.

Tal afirmação nos faz perceber que a resolução de uma atividade, ou seja, a conclusão de algo proposto não significa que tenha havido compreensão. Valente (1999, p. 39) afirma que “Não será o fazer, o chegar a uma resposta, mas a interação com o que

está sendo feito, de modo a permitir as transformações dos esquemas mentais”. No que se refere a programação de computadores, se o aluno for capaz de dominar o conteúdo ministrado, aplicá-lo na resolução de problemas, refletir sobre os resultados encontrados e, ainda, encontrar caminhos para melhorar tais soluções, provavelmente terá alcançado a compreensão.

Foi com base nesses conceitos que Papert desenvolveu a Filosofia LOGO, termo que, do grego, significa "pensamento, raciocínio e discurso", ou também, "razão, cálculo e linguagem”.

3.3. Da Lógica de Programação à Filosofia LOGO

Para alunos dos cursos superiores de Informática⁶, dominar os conceitos de lógica de programação é fundamental para o bom aproveitamento de todo o curso. Já no primeiro período do curso, estes alunos tem contato com uma disciplina que pode ter várias nomenclaturas, entre elas, o próprio nome do conteúdo, ou seja, Lógica de Programação. Tal disciplina exige alto grau de abstração e capacidade interpretativa, o que faz com que seja considerada de difícil aprendizado. Após terem-na cursado com êxito, os alunos são considerados aptos para aprender as linguagens de programação propostas pelo curso/universidade.

Com base nos conceitos de lógica de programação, pode-se desenvolver soluções computacionais em forma de programas. Essas soluções são conhecidas como algoritmos, são textos com instruções ordenadas e que serão decodificadas por um programa específico (GUIMARÃES; LAGES, 1994) – um compilador, interpretador, tradutor ou ambos – para uma linguagem inteligível ao computador.

A construção de algoritmos depende das estruturas de raciocínio lógico que são desenvolvidas ao longo da educação básica e de toda a vida do indivíduo. Portanto, quanto mais desenvolvidas tais estruturas estiverem, mais facilidade haverá na construção de algoritmos.

Uma série de fatores pode interferir negativamente no processo de aprendizagem de programação, e são apontados por Gomes (2008):

⁶ Segundo o Ministério da Educação - MEC a área de Ciências Exatas e da Terra, tem como uma de suas subáreas a Informática e, dentro dela, diversas nomenclaturas para os cursos superiores, entre elas: Ciências da Computação, Engenharia da Computação e Sistemas de Informação.

- a) método de ensino: o ensino não pode ser personalizado devido às restrições temporais, além disso, conceitos dinâmicos são apresentados de forma estática e há uma preocupação maior em ensinar a linguagem e sua sintaxe do que trabalhar a resolução de problemas;
- b) método de estudo: a prática não é suficiente e os alunos não estudam o necessário para aprenderem a programar;
- c) habilidades e atitudes dos alunos: para programar é necessário resolver problemas e alguns alunos não apresentam as habilidades necessárias para a sua resolução, em função de déficits de conhecimentos lógicos e matemáticos, falta de conhecimento das estruturas sintáticas e gramaticais das linguagens o que acarreta dificuldades em detectar erros sintáticos;
- d) natureza da programação: para programar é necessário alto nível de abstração e, durante a construção do programa, o programador tem que se preocupar tanto com a sintaxe quanto com a construção do algoritmo;
- e) aspectos psicológicos: falta de motivação; aprendizado num período de transição em sua vida, onde o aluno encontra-se num momento de instabilidade e, ainda, no início do curso.

Acreditamos que um dos fatores que mais interferem negativamente é o fato da não personalização do ensino, pois alguns alunos já trazem experiências com linguagem de programação enquanto outros não possuem qualquer conhecimento. Isso resulta em desinteresse de forma que o grupo, neste primeiro momento não é homogêneo.

Cabe ao professor perceber os elementos críticos do processo de aprendizagem de sua turma e promover recursos que facilitem a compreensão dos assuntos ministrados.

Lima e Leal (2013, p.15) mencionam que

a didatização, o cuidado com as dificuldades de aprendizagem, a contextualização dos conteúdos, a adequação da linguagem utilizada nas exposições, o preparo técnico e pedagógico e a vivência prática na área de programação por parte dos docentes são características que foram entendidas como necessárias para uma melhor mediação da aprendizagem.

O que se conclui das análises anteriormente mencionadas é a necessidade de uma abordagem específica no que se refere ao ensino de linguagens de programação, devido não só à dificuldade de aprendizado, mas também pelas suas características. Quando dizemos “abordagem específica”, fazemos referência à necessidade de trabalhar tais conteúdos de acordo com uma metodologia que favoreça a aprendizagem. Assim é necessário criar um ambiente propício para tal, onde seja possível respeitar o tempo de assimilação de cada aluno e, ainda, as estruturas lógicas de pensamento que estes já

trazem e as que ainda necessitam ser construídas para que cada aluno esteja apto a desenvolver algoritmos sejam eles de nível básico ou avançado.

A Filosofia LOGO, concebida por Papert (1988, p.35), visa apresentar o ambiente de programação como um “lugar” onde o aluno estará no controle da aprendizagem, e “ensinará o computador a pensar”. Assim, ele “embarca numa exploração de como ele mesmo pensa”, experiência e reflexão, que muitos que chegaram à fase adulta não tiveram durante seus estudos. O ambiente desenvolvido por ele propõe que o aluno possa se utilizar da tecnologia para se motivar e criar condições de desenvolvimento autônomo e autoral. Neste ambiente, o aluno poderá criar suas próprias atividades e, ao fazê-lo, desenvolver suas estruturas lógicas de pensamento.

Papert (1988) conceitua o termo “matofobia” como o medo de aprender. A palavra “matofobia” leva a duas associações: uma delas é o conhecido medo da matemática, que muitas vezes tem a intensidade de uma verdadeira fobia. A outra vem do radical “mathe”. Em grego significa “aprender” de maneira geral.

Com base nisso, Papert (1988) afirma a importância da contextualização através do que ele chama de “matelândia”, que consiste em criar um ambiente propício para a aprendizagem de qualquer conteúdo que se deseja aprender; é tornar a área de conhecimento, a matemática, por exemplo, como “vocabulário natural”. Usa-se as demais habilidades do aluno para favorecer o aprendizado, ainda que tais habilidades sejam de outra área específica. “O conceito de mobilizar os vários interesses e facilidades [...] para promover todos os domínios da atividade intelectual é uma resposta à sugestão de que diferentes aptidões podem refletir diferenças reais no desenvolvimento do cérebro” (PAPERT, 1988, p. 67). Ele apresenta a Filosofia LOGO como um dos caminhos para construção dessa “matelândia”.

Santanchè e Teixeira (1999) afirmam que a Filosofia LOGO é baseada nos princípios construtivistas de Piaget, onde o aluno/aprendiz é encarado como um construtor ativo de seu conhecimento. Nessa perspectiva, o computador e sua programação são meios com os quais se constroem artefatos de conhecimento, sendo este o mediador para a reflexão sobre como o conhecimento é construído usando a ideia de que é o aluno quem ensina o computador.

Para que esta pudesse ser utilizada, foi desenvolvida uma linguagem que possui o mesmo nome e que representa, na realidade, uma família de linguagens. Seus criadores foram Wallace Feurzeig, Daniel Bobrow e Seymour Papert. Desenvolvida no

laboratório de inteligência artificial do MIT, já em suas primeiras versões, o LOGO permitia controlar os movimentos de um pequeno robô, usando os conceitos de inteligência artificial. A partir desses estudos, Papert considerou a importância da concepção de uma linguagem cujo foco fosse a aprendizagem e não o ensino e que contribuísse para o desenvolvimento da inteligência.

O LOGO não é uma linguagem tradicional de programação do ponto de vista técnico, pois foi concebido com o objetivo de provocar o desenvolvimento das estruturas de pensamento de crianças a partir do concreto, do movimento de seu próprio corpo. A partir da interação com o concreto, era possível manipular um robô que possuía movimentos simples e foi chamado de "tartaruga" devido à sua forma, tornando-se, posteriormente, a marca registrada da linguagem.

Como no período em que foi criada já havia terminais gráficos de baixo custo, a tartaruga mudou-se para a tela do computador, onde pode mover-se de forma mais rápida e precisa, sendo possível comandar apenas os movimentos na tela, movendo-se, por exemplo, para frente ou para trás, girar para a direita ou esquerda, assumir diferentes formas ou riscar a tela produzindo desenhos ao se mover, entre outros. Hoje, LOGO é o nome genérico de uma coleção ou família de linguagens de programação criadas sob a mesma perspectiva. Há algumas diferenças em cada "membro" desta família, as mudanças desejadas podem ser gerenciadas pelos seus usuários e compartilhadas com a comunidade de programadores, originando assim um novo membro desta família.

Papert (1988) apresenta a linguagem LOGO como um caminho para a construção de um ambiente propício à aprendizagem e com base na sua concepção filosófica, conhecida como Filosofia LOGO⁷, que sugere que atividades desenvolvidas com computador e LOGO levam à aprendizagem em decorrência de desafios/problemas que o aluno precisa solucionar.

Durante o período de operação da mesma, o aluno tem a experiência de ensinar comandos ao computador, escrevendo assim, sobre a linguagem existente uma nova linguagem com base em sua melhor assimilação dos comandos preexistentes. Tais procedimentos permitem ao aluno desenvolver um novo comando com uma sequência de instruções já existentes. Ao dar uma instrução desconhecida pela linguagem, esta rapidamente dá um feedback, atribuindo para si o não conhecimento de tal elemento. A

⁷ Abordaremos a Filosofia LOGO em um capítulo específico.

resposta dada é: “Não aprendi...”, o que faz com que o aluno se perceba como aquele que ensina ao computador e, ainda, não considere o erro algo ruim, mas uma possibilidade de novos aprendizados.

Nessa abordagem, ao contrário do que tradicionalmente se observa, embora com um papel relevante, o professor não é um elemento central, pois ele e os alunos exercem constantes interações no processo de ensino e aprendizagem.

Baseando-se no exposto, decidimos apresentar a adequação da metodologia da disciplina ministrada no primeiro período dos cursos de bacharelado em sistemas de informação e licenciatura em computação para favorecimento do desenvolvimento das estruturas lógicas de pensamento, conforme descrito na metodologia e no capítulo onde é feita a análise dos dados coletados em campo.

No próximo capítulo apresentamos as linguagens de programação utilizadas nesta pesquisa e suas características para o aprendizado.

CAPÍTULO 4

LINGUAGENS UTILIZADAS NA SALA DE AULA

4.1. A Linguagem LOGO

Como dito anteriormente, a linguagem de programação LOGO foi projetada como um instrumento de aprendizagem, em consonância com a Filosofia que leva seu nome, usando-se a ideia de criação de um ambiente para o aprendizado de um determinado conteúdo ou área. Uma das facilidades desta linguagem é o fato de o programador poder criar comandos com nomes familiares e exclusivos. Durante muito tempo, especialmente no Brasil, foi considerada erroneamente como linguagem para crianças. Entretanto o LOGO é uma linguagem tão poderosa quanto as demais, pois possui comandos que permitem criar uma infinidade de algoritmos e atividades e, além disso, é de fácil aprendizado.

O termo LOGO vem do grego "*logos*" que significa "palavra", uma referência aos comandos da linguagem. O LOGO representou uma mudança de paradigma sobre o uso do computador na educação, por possibilitar uma exploração da programação de computadores de maneira lúdica e interessante.

Tendo sido criado no final dos anos 60, somente na década seguinte o LOGO começou a ser usado nas escolas dos Estados Unidos. No Brasil, as pesquisas começaram na década de 80 e a linguagem foi muito bem aceita por pesquisadores e professores do ensino fundamental. Na década de 90, o interesse pela linguagem LOGO. Atualmente, esta linguagem praticamente não é usada, a não ser em alguns projetos com robótica educacional, devido à facilidade de se manipular interfaces robóticas através dela. Um dos fatores que influenciaram para a diminuição do seu uso foi a popularização de interfaces gráficas, editores de imagens e programas multimídia. De fato os softwares educacionais com suas interfaces e multimídia tornaram-se muito mais atraentes do que o próprio LOGO e a ideia equivocada de usá-lo apenas como "software educativo", ao invés de uma linguagem para desenvolvimento das estruturas lógicas de pensamento, levou ao abandono.

O LOGO é uma linguagem interativa e interpretada. Quando dizemos que uma linguagem é interativa, relacionamos à capacidade da interface interagir, ou seja, fornecer feedback para o usuário. Já ao dizer que é interpretada, relacionamos ao fato de que cada comando, ou seja, cada instrução dada ao computador pode ser executada isoladamente, sem a necessidade de construir um programa completo. Sendo assim, o LOGO "interpreta" cada comando, traduzindo para a linguagem de máquina e o executa.

4.1.1. SuperLOGO

Consideramos como ambiente LOGO os programas que foram criados a partir da Filosofia LOGO e que seguem as mesmas características. A maioria dos ambientes divide a tela do computador em pelo menos duas regiões: a janela de comandos e a janela de execução (ver figura 1). Na janela de execução a tartaruga se move, realizando as instruções comandadas. O modo como estas áreas estão dispostas na tela depende da versão e do ambiente que estamos usando, o que faz também com que a linguagem varie.

Para nossa pesquisa usamos inicialmente o SuperLOGO, um LOGO adaptado para o português pelo Núcleo de Informática Aplicada à Educação da UNICAMP (NIED)⁸ da UNICAMP e que é distribuído gratuitamente pela internet.

Como é possível ver na figura 2, a área de comandos fica localizada na parte inferior da tela, onde são digitados os comandos na caixa de texto (área branca) e visualizar acima, na área cinza, que registra os comandos digitados, ainda que, por erro de digitação ou nomenclatura, não seja possível executar. Se houver erros, o próprio programa se encarrega de notificá-lo nesta área.

No centro da tela principal, pode ser observada uma tartaruga que é nomeada pelo próprio programa como "tat". Como um robô, tat tem movimentos pré-definidos, embora, como mencionado, seja possível criar novos comandos. Os comandos da linguagem são simples e diretos. Por exemplo: se deseja que apareça um texto escrito na área de comandos basta usar o comando escreva e entre colchetes o texto que deseja que

⁸ Site do Núcleo de Informática Aplicada à Educação da UNICAMP: <http://www.nied.unicamp.br>

seja reproduzido, ex.: escreva [LOGO]. Ao pressionar *Enter* tal mensagem será reproduzida.

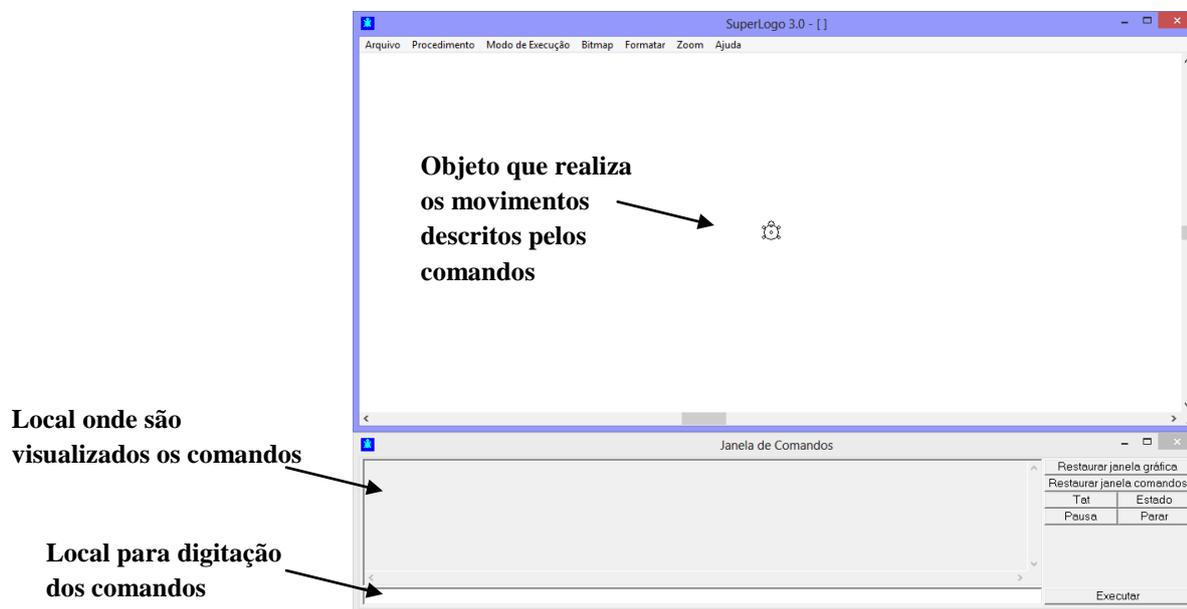


Figura 2: Tela inicial do Ambiente Super LOGO

Apresentamos a seguir uma tabela com os principais comandos da linguagem e seus respectivos significados.

Comando	Significado
pf	Para frente
pt	Para trás
pd	Para direita (valor do ângulo medido em graus)
pe	Para esquerda (valor do ângulo medido em graus)
ul	Use lápis
un	Use nada
ljc	Limpe janela de comandos
at	Apareça tartaruga
dt	Desapareça tartaruga
tat	Limpa a tela e coloca a tartaruga na posição inicial
repita	Permite executar várias vezes os comandos especificados
escreva	Permite escrever um texto na tela

Tabela 2: Principais comandos da Linguagem LOGO

Ao digitar *pf 50* ou *parafrente 50*, está comandando a tartaruga para mover-se para frente 50 pontos/passos na tela. Os comandos *paraesquerda* (*pe*) ou *paradireita* (*pd*) giram a tartaruga de acordo com o desejado. A tartaruga deixa um rastro por onde

passa, como se estivesse usando um lápis, caso você deseje mover a tartaruga sem riscar use o comando *usenada* (un) e para retornar *uselapis* (ul). Assim, é possível combinar comandos e produzir movimentos que criem desenhos na tela.

Para evitar a digitação de comandos várias vezes, usamos o comando *repita*. Por exemplo: Para desenhar um quadrado, poderiam ser dadas as seguintes instruções:

pf 50 pe 90

pf 50 pe 90

pf 50 pe 90

pf 50 pe 90

Entretanto é mais simples usar o comando *repita*,

repita 4 [pf 50 pe 90]

Na realidade, o que define como uma pessoa irá criar um quadrado é sua capacidade de abstração e a utilização do caminho que ele decide como mais fácil. É fundamental deixar que cada um construa suas tarefas e mostrar que o resultado final pode ser obtido com mais ou menos esforço computacional, mas que ainda assim, o importante no processo é o resultado obtido.

Um elemento muito importante da linguagem LOGO é a possibilidade de criar novos procedimentos, que nada mais são do que um grupo de instruções que produzem algum resultado. Por exemplo, voltando à ideia da construção de um quadrado. Ao digitar a palavra *quadrado* na área de comandos, se não houver nenhum comando/procedimento com este nome, obteremos a seguinte mensagem na caixa de entrada: "AINDA NÃO APRENDI QUADRADO", o que significa que a linguagem não reconhece o comando ou o procedimento.

Há no ambiente, um local específico para criar um procedimento, conforme apresentado na Figura 3. Bastando para isso, clicar no menu *Procedimento*, opção *Novo*.

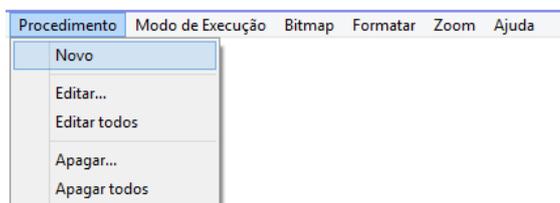


Figura 3: Passos para criar um procedimento

Para se definir um procedimento usamos o comando *aprenda*. Sendo assim, para criar o procedimento que desenhará o quadrado, deve-se digitar:

aprenda quadrado
repita 4 [pf 50 pe 90]
fim

Veja, na figura 4, a tela do editor de procedimentos.

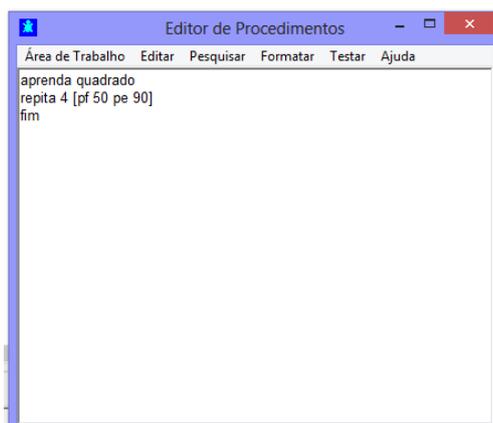


Figura 4: Editor de procedimentos

Ao clicar na opção atualizar do menu Área de Trabalho e fechar o editor, o procedimento já está criado, podendo ainda ser editado ou apagado. Estando o procedimento criado, basta digitar *quadrado* e a tartaruga desenhará o quadrado na tela.

Um dos exemplos utilizados durante as aulas ministradas à turma pesquisada foi o procedimento castelo.⁹(SANTOS, 2006, p.7)

Exemplo: Construindo um castelo

aprenda castelo
tat
pd 90
un pt 150 ul pf 300
pe 90 pf 160 pe 90
repita 2 [pf 10 pe 90 pf 10 pd 90 pf 10 pd 90 pf 10 pe 90]
pf 10 pe 90
pf 160 pt 120 pd 90
repita 9 [pf 10 pd 90 pf 10 pe 90 pf 10 pe 90 pf 10 pd 90]
pf 10 pd 90 pf 10 pe 90 pf 10

⁹ Este exercício foi extraído da apostila de SuperLOGO da Universidade Estadual Paulista

```

pe 90 pf 130 pt 160 pd 90
repita 2 [ pf 10 pe 90 pf 10 pd 90 pf 10 pd 90 pf 10 pe 90]
pf 10 pe 90
pf 160
un mudexy -140 90
ul pd 180 repita 4 [ pf 30 pd 90]
un mudexy 110 90
ul repita 4 [ pf 30 pd 90]
un mudexy -40 0
ul pf 60 pd 90 pf 80 pd 90 pf 60
un pt 30
pd 90 pf 50 pd 90
fim

```

Além dos comandos já apresentados anteriormente, foi utilizado o *mudexy* que posiciona a tartaruga no ponto determinado, tendo a tela como um plano cartesiano e sendo x e y como os eixos correspondentes.

O resultado obtido pode ser observado na figura 5.

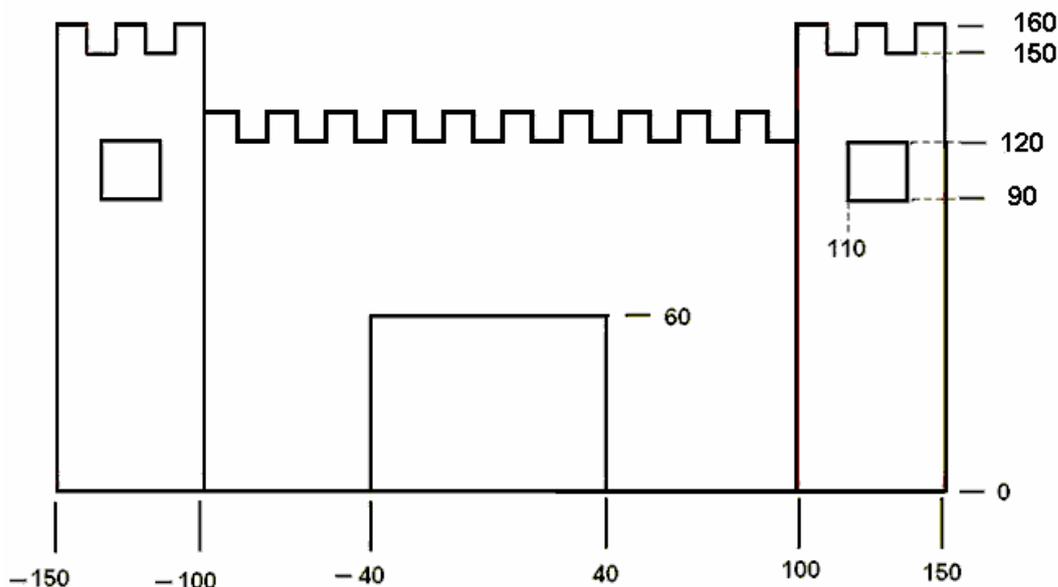


Figura 5: Imagem produzida com o procedimento "castelo"

Para pintar o castelo foi criado outro procedimento nomeado *corcastelo*. O objetivo deste procedimento era posicionar a tartaruga nos locais onde desejava-se pintar através do comando *mudexy* e alterar a cor através do comando *mudecp* e *pinte*. *Mudecp* permite alterar a cor do pincel para a cor selecionada, cor essa pertencente a uma tabela da própria linguagem e que pode ser referenciada através de números. *Pinte* é o comando responsável por pintar a área onde a tartaruga está posicionada. Foi usado ainda o comando espere para que durante a execução pudesse ser visto o castelo sendo

colorido parte a parte com uma pausa de 3 segundos. O código abaixo apresenta o procedimento *corcastelo*.

aprenda corcastelo

```

un
  mudexy -130 110 pd 180 mudecp 4 pinte pd 180 espere 30
  mudexy 130 110 pd 180 mudecp 4 pinte pd 180 espere 30
  mudexy -65 20 pd 180 mudecp 10 pinte pd 180 espere 30
  mudexy 130 20 pd 180 mudecp 14 pinte pd 180 espere 30
  mudexy -130 20 pd 180 mudecp 14 pinte pd 180 espere 30
  mudexy 0 20 pd 180 mudecp 7 pinte pd 180 espere 30
fim

```

Como apresentado na figura 6, temos o castelo pintado com as cores definidas.

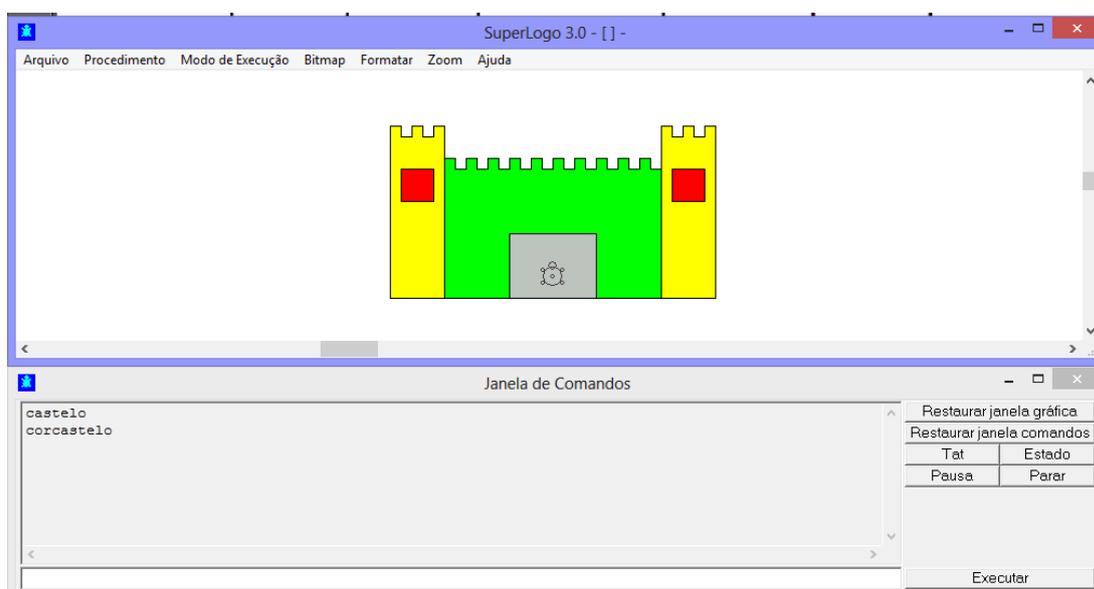


Figura 6: Resultado Final do Exercício.

Os comandos e exemplos anteriores representam uma breve apresentação do que é possível desenvolver com a linguagem LOGO, pois através dos comandos apresentados podem ser criadas infinitas possibilidades.

4.1.2. SCRATCH

Conforme apresentado anteriormente, existe uma família de ambientes que utilizam a linguagem LOGO. O SCRATCH é um desses ambientes que usa os mesmos comandos, porém sua apresentação é bem diferente do SuperLOGO.

O SCRATCH é um ambiente desenvolvido pelo grupo Lifelong Kindergarten no Media Lab do MIT. Assim como os demais ambientes, ele é fornecido gratuitamente. Foi projetado especialmente para usuários entre 8 e 16 anos e funciona como um jogo de quebra-cabeça, onde os comandos podem ser arrastados e encaixados para produzir os efeitos desejados. A figura 7 apresenta a tela inicial do SCRATCH.

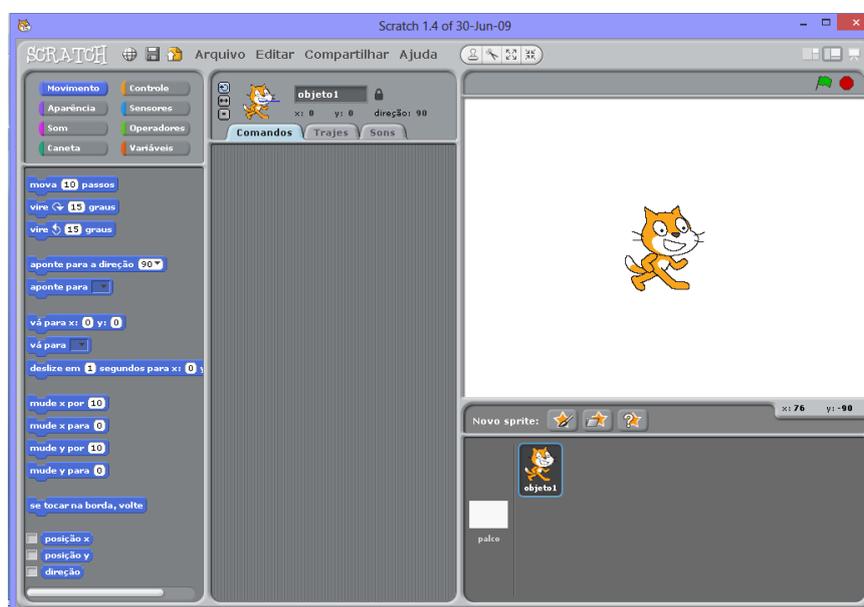


Figura 7: Tela inicial do Scratch

Neste ambiente, os comandos são apresentados em forma de peças que ficam posicionados do lado esquerdo. Para montar seu programa, o usuário deve arrastá-los para a área de comandos. A figura 8 mostra como seria o bloco de comandos para criar um quadrado.



Figura 8: Construção de um quadrado.

Ao optar por usar a repetição, basta selecioná-la na área de comando, pois como as peças estão prontas para serem encaixadas, os comandos foram agrupados por suas características. Nesse caso, a repetição está no grupo *Controle* (ver figura 9).



Figura 9: Agrupamento dos Comandos

A construção do quadrado usando os comandos de repetição ficaria como apresentado na figura 10.



Figura 10: Construção de um quadrado usando repetição

Para a execução de um comando ou sequência de comandos, basta clicar no bloco da área específica. Como podemos ver, a tartaruga que ficava posicionada no centro da tela foi substituída por um gato, que também pode ter o formato desejado pelo usuário, bastando para isso alterar o "traje", que podem ser personalizados ou inseridos

da galeria do próprio programa. Existem trajes de animais, objetos, desenhos de pessoas, entre outros.

Sons também podem ser inseridos no programa de maneira simples usando a aba com este nome, como mostra a figura 10. No SCRATCH não é possível criar procedimentos, visto que os comandos são agrupados e posicionados em uma área visível. No site do projeto¹⁰, qualquer pessoa ou instituição pode compartilhar seus projetos. É possível optar por baixar a ferramenta ou criar seus projetos online e, ainda, baixar projetos prontos compartilhados por usuários de todo o mundo e modificá-los.

4.2. Português Estruturado (Portugol)

O primeiro passo para o aprendizado de programação em cursos e universidade, geralmente, é aprender o que conhecemos como pseudolinguagem, que nada mais é que uma linguagem fictícia com uma estrutura não rígida criada apenas para ensinar os conceitos de programação sem que se esteja preso à alguma linguagem.

Usa-se a linguagem natural, ou seja, comandos escritos em língua portuguesa para que não seja necessário que o aluno decore os comandos, mas se preocupe exclusivamente com a lógica de programação e a estruturação da solução de problemas. Tal linguagem pode ser chamada Português estruturado ou Portugol.

O "português estruturado possui elementos (comandos) que possibilitam uma representação linear, de um processo ou sequencia de passos, mostrando a transição de informações entre os comandos, dados de entrada e saída." (PIVA et al., 2011, p.43)

Leite et al. (2013, p. 638) menciona que, o processo de aprendizado feito diretamente em uma linguagem de programação formal, pode torná-lo complexo e desestimulante. Isso ocorre especialmente pelo fato dos comandos estarem em outra língua, especificamente o inglês. Através de comandos mais claros, espera-se que o aluno aprenda de forma mais rápida e eficaz, pois pode testar seus programas e visualizar os resultados de forma imediata e direta.

A partir do momento que o aluno domina os princípios básicos da programação de computadores que foram trabalhados com o português estruturado, este é

¹⁰ <http://scratch.mit.edu/>

considerado apto a aprender qualquer linguagem de programação. Isso justifica que esse conteúdo seja ministrado no momento inicial do aprendizado.

4.2.1. VISUALG

Inicialmente, pretendíamos utilizar apenas programas que tivessem por base a linguagem LOGO, mas no momento em que os alunos tomaram conhecimento do conteúdo programático da disciplina e que estudariam pseudocódigo/português estruturado por iniciativa própria iniciaram a utilização do software VisualG. Os alunos que já conheciam programação tinham experiência com esta ferramenta e encorajaram os outros alunos a utilizá-la também. Esta ferramenta acabou integrando a maioria das aulas, permitindo que os alunos executassem os códigos gerados e por ser simples seu uso.

Por não podermos alterar a ementa do curso, embora tenhamos trabalhado com a linguagem LOGO, desde o início sabíamos que ela não poderia ser nosso foco principal na disciplina. Parte dos códigos gerados a partir dos conteúdos não poderiam ser testados diretamente com esta linguagem, pois exigem um nível de complexidade mais alto na elaboração dos algoritmos. Assim, deve ser ressaltado que durante todo o trabalho desenvolvido na disciplina, independente da linguagem em estudo, nossa ênfase foi a Filosofia LOGO, que, criando um ambiente favorável ao aprendizado do conteúdo proposto pela disciplina.

A ideia de se desenvolver um programa como o VisualG¹¹ nasceu da necessidade de um professor¹² da disciplina de algoritmos em ter uma ferramenta que auxiliasse os alunos a testarem seus exercícios nas aulas de programação, para "rodar" os programas construídos anteriormente no papel. Esse programa foi pioneiro. Antes dele não existiam ferramentas para essa finalidade;

Na prática, percebemos que desenvolver programas no papel é um obstáculo para o aprendizado de técnicas de elaboração de algoritmos, mas também submeter iniciantes ao rigor de linguagens de programação pode ser um tanto arriscado para o processo de aprendizado.

¹¹ Página Oficial: <http://www.apoioinformatica.inf.br/produtos/visualg>

¹² O software foi desenvolvido pelo professor Cláudio Morgado de Souza que atua como programador/analista e professor universitário no Rio de Janeiro.

Pensando nisso, foi desenvolvida uma ferramenta capaz de simular os comandos em português, como *leia* e *escreva*, permitir a verificação dos valores das variáveis, o acompanhamento continuado da execução de um algoritmo (o que é de relevante valor didático), ainda suportar um modo simples de depuração¹³.

Portanto, o VisualG é um programa que edita, interpreta e executa algoritmos como outros programas para computadores, possuindo recursos que facilitam o aprendizado das técnicas de programação: simulação da "tela" do computador, visualização de variáveis, *breakpoints*¹⁴, entre outros. Tem como base uma linguagem semelhante à do *Portugol* que é largamente conhecida e utilizada em cursos, tanto em nível técnico quanto superior, de introdução à programação.

O VisualG é um programa gratuito e não requer configurações específicas para que seja instalado, podendo ser usado em qualquer computador e, por isso tem sido usado em larga escala por alunos e instituições. Na figura 11, apresentamos a tela principal do VisualG.

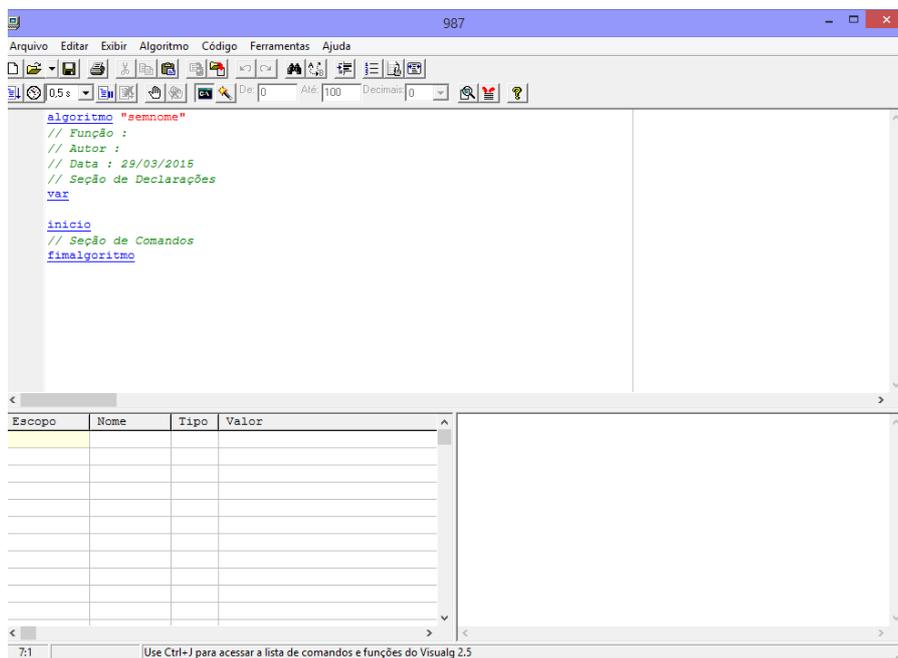


Figura 11: Tela Inicial do VisualG

A importância do uso de compiladores ou programas que permitam ao aluno testar as soluções construídas é que além de estimulante, agiliza o processo. alunos

¹³ O processo de depuração consiste em encontrar e reduzir erros/problemas num programa. Esses erros incluem aqueles que impedem o programa de ser executado e aqueles que produzem um resultado inesperado (exceção).

¹⁴ Breakpoints são pontos de parada durante a execução de um programa, ou seja, ao inserir um breakpoint em um programa e executá-lo em seguida, o programa irá interromper sua execução no ponto exato onde este elemento foi inserido.

engajados e estimulados, se comprometem com seu aprendizado e com as tarefas que necessita realizar. Nisso contribui a proposta da Filosofia LOGO que visa, essencialmente, tornar a sala de aula um ambiente contextualizado, onde o aluno possa falar a língua do conteúdo que está aprendendo, ou seja, tornar possível a construção de uma matelândia.

Acreditamos que o uso das linguagens e das ferramentas disponibilizadas contribuiu para a criação dessa matelândia, ou seja, desse ambiente propício ao aprendizado de programação de computadores e reduziu o receio de aprender ou de cometer erros na elaboração das soluções, visto que o erro é o caminho para novas descobertas.

CAPÍTULO 5

RECORTES DO DIÁRIO DE CAMPO: O dia a dia das aulas

Conforme descrito anteriormente, a presente pesquisa foi realizada em uma turma de primeiro período que unia alunos dos cursos de Bacharelado em Sistemas de Informação e Licenciatura em Computação. Inicialmente trabalhei com cerca de 60 alunos.

As aulas ocorriam sempre às terças-feiras e dispúnhamos de quatro tempos de 40 minutos cada.

A primeira aula ocorreu no dia 18/02/2014. Este foi o primeiro contato com a turma, que conheceu a disciplina e a ementa a ser trabalhada e a proposta de trabalho desta pesquisa. Houve um momento para apresentação individual, onde cada aluno pode, além de dizer seu nome, se tinha algum conhecimento/experiência com programação e em qual nível.

Propus a criação de um grupo no Facebook onde pudéssemos compartilhar especialmente conteúdos da disciplina. Neste espaço, durante todo o semestre os alunos compartilharam suas soluções, trocaram experiências, o que permitiu uma maior interação entre eles.

Nesta aula não houve exposição do conteúdo da disciplina apenas do que seria trabalhado durante o semestre, explicação de que passaríamos por uma transformação na metodologia e que trabalharíamos de forma alternativa à tradicional.

No dia 25/02/2014 iniciei o conteúdo previsto com um breve histórico do computador e seu funcionamento básico. Discutimos sobre os conceitos básicos de computação (Hardware, Software, Dispositivos de Entrada/Saída, Armazenamento de Dados, Processamento). Sentimos a necessidade de formalizar a entrevista realizada na primeira aula, com um breve questionário, que possuía as seguintes questões:

- Você tem algum conhecimento de programação?
- Quais as linguagens que você conhece? Diga o nome da(s) linguagem(s):
- Em que linguagem já programou? Como aprendeu? Ainda que tenha aprendido sozinho, essa informação é importante para nós.
- Qual recurso você usa para tirar as dúvidas que possui? Participa de alguma comunidade? Tem contato com outros programadores?
- O que você já desenvolveu?

Na semana seguinte não houve encontro com a turma por ser quarta-feira de cinzas.

No dia 11/03 os alunos conheceram as definições dos principais elementos que trabalharíamos em sala de aula. São eles: Lógica de Programação, Algoritmos, Pseudocódigo. Foi possível discutirmos sobre os tipos de Linguagem de Programação existentes, bem como suas características e diferenciais no mercado, além do processo de construção de algoritmos, que deve ter início, meio e fim, ou seja, entrada, processamento e saída.

Neste encontro, os alunos realizaram o primeiro teste que chamamos de Teste Diagnóstico (Ver Apêndice 4). Eles foram incentivados a elaborar a sequência lógica para solucionar os problemas propostos independente de regras de linguagem ou códigos predefinidos. Falamos sobre a importância da realização do teste e que o mesmo deveria ser elaborado não como uma avaliação, mas como um instrumento onde poderiam apresentar o que conheciam ou não sobre o conteúdo.

No dia 18/03/2014 após retomarmos a questão do processo de elaboração de algoritmos, realizamos uma atividade em turma. A atividade consistia em descrever o passo a passo para programar um robô para pintar uma parede.

Apresentei a proposta da atividade através do seguinte enunciado: *"Suponha que você tem uma parede e, de frente para ela, posicionado ao centro da mesma, um robô, ao lado dele temos uma lata de tinta e um pincel. Como devo proceder para ensinar o robô a pintar a parede?"*

Inicialmente os alunos consideraram a proposta como algo simples. Abordei a necessidade de estabelecer uma lista de ações para ensinar o robô a então pintar a parede. Os alunos reconheceram que dentro de um contexto que parecia simples, eles teriam várias formas de abordar o problema e, ainda, vários pontos de vista e formas de resolvê-lo.

Notei que alguns alunos se destacaram e não cooperaram com a atividade, ficaram apenas observando. Neste momento, não foi possível avaliar se eles não compreenderam a proposta ou se não sabiam como responder.

A maioria da turma expôs seu ponto de vista e sua possibilidade de solução. Houve momentos de troca intensa e debate entre eles do que estava faltando ou da possibilidade de conclusão. Os passos eram registrados no quadro e discutidos de forma a verificar se estavam na sequência lógica correta, se representavam uma ou mais

instruções ou se haveria repetição em algum momento. Eles sentiram a necessidade de realizar testes antes de alguns passos, para verificar, por exemplo, se o pincel estava sem tinta, onde terminava os limites da parede, entre outros.

Pelo curto tempo não conseguimos concluir a atividade 100%. Ao final apresentei novas possibilidades e coisas que deveriam ser pensadas quando se trata de programar o robô para realizar tarefas. Uma programação de uma atividade não pode ser feita da mesma forma que nós realizaríamos, pois o tempo todo realizamos tarefas de maneira quase que automática, ou seja, sem refletir sobre o que estamos de fato fazendo.

A turma demonstrou-se desejando avançar tanto no conteúdo quanto no que se refere à programação. Podemos compreender que esta atividade foi um momento que, além de intensa troca, permitiu aos alunos compreenderem como funciona o processo de programação de computadores. Na aula seguinte, ao propor desafios mais simples que poderiam ser resolvidos individualmente, a maioria resolveu de maneira rápida e correta. Após cada desafio, reservávamos um momento para discussão e, poucos ajustes tiveram que ser feitos às soluções.

Neste momento ainda não estávamos usando nenhuma linguagem ou ferramenta para a programação, as soluções eram esboçadas em português, na linguagem mais próxima possível à língua falada e escrita.

A participação do grupo e a interação entre eles foram fundamentais para a realização da atividade. De uma forma geral, eles interagiam muito bem entre eles em todas as atividades e estavam dispostos a participar de tudo que lhes era proposto. Dessa forma, procurei aproveitar esse momento para realizar atividades capazes de mantê-los engajados e produzindo sempre.

No dia 25/03/2015 apresentei algumas regras para construção de algoritmos e defini alguns comandos a fim de demonstrar como funciona numa linguagem de programação o fato de termos comandos pré-definidos com funções específicas e podermos construir nossas soluções com base nesses comandos.

Usando como base o exemplo da aula anterior "Robô Pintor" foram explorados uma série de desafios abordando as características de um algoritmo, sua sequência lógica e focando na resolução de problemas em si. Foram feitos exercícios onde os alunos tiveram que criar sequências lógicas para elaborar soluções para problemas propostos.

No dia 01/04/2014, apresentei o Português Estruturado e alguns conceitos básicos de Programação como: a estrutura do programa, tipo de dados, variáveis e constantes e comandos de leitura e impressão. Assim, os alunos puderam conhecer a estrutura básica de um programa em Português Estruturado e realizar um exercício simples. Neste momento os alunos propuseram o uso do programa VisualG que foi incorporado às aulas.

Visto que a aula do dia 15/04/2014 estava destinada à aplicação de avaliação, no dia 08/04/2014 propus uma listagem com diversos desafios de níveis de dificuldade diferentes, que foi realizado pelos alunos em grupos de 3 ou 4. O objetivo da solução de tais desafios em grupo foi propiciar um período de discussão durante a busca de soluções.

Todos os alunos presentes conseguiram elaborar as soluções e, durante a aula pude perceber que muitos se dispunham a auxiliar os demais.

A aula de 15/04/2014 estava destinada a avaliação. Pela estrutura da disciplina e pelas exigências da própria instituição não foi possível reformular a avaliação, assim foi necessário elaborar uma prova teórica contendo questões a serem resolvidas no papel. Como na primeira avaliação é possível compor nota com outras atividades e flexibilizá-la, optei por realizá-la em duplas.

No dia 22/04/15 não houve aulas na instituição, pois foi definido no calendário como recesso de feriado, já que no dia anterior foi feriado de Tiradentes (21/04) e no dia seguinte feriado de São Jorge (23/04), este último não consta na lista de feriados nacionais, sendo um feriado da cidade do Rio de Janeiro.

No dia 29/04/2015 trabalhei as estruturas condicionais *Se* e *Caso*. Após a apresentação das mesmas foram propostos desafios que incluíssem tais estruturas que já haviam sido trabalhadas no momento em que solucionamos o problema do "Robô Pintor", porém de maneira não formal.

Dia 06/05/2014 foi destinado à apresentação das estruturas de repetição *Para*, *Repita* e *Faça...Enquanto*. A questão da repetição de blocos de comando também havia sido enfatizada no problema do "Robô Pintor". Foram propostos problemas de grau mais complexo que os anteriores envolvendo todas as estruturas aprendidas até então.

No dia 13/05 introduzi o SuperLOGO. Como os alunos já conheciam as estruturas que seriam usadas na linguagem LOGO, apresentei esta ferramenta para que trabalhássemos a Lógica de Programação.

Assim eles conheceram a estrutura da linguagem, os comandos básicos e foram propostos desafios que utilizavam operações de entrada de dados, estruturas condicionais, de repetição. Os primeiros desafios consistiam em criar na tela as seguintes figuras. (SANTOS, 2006, p.5)

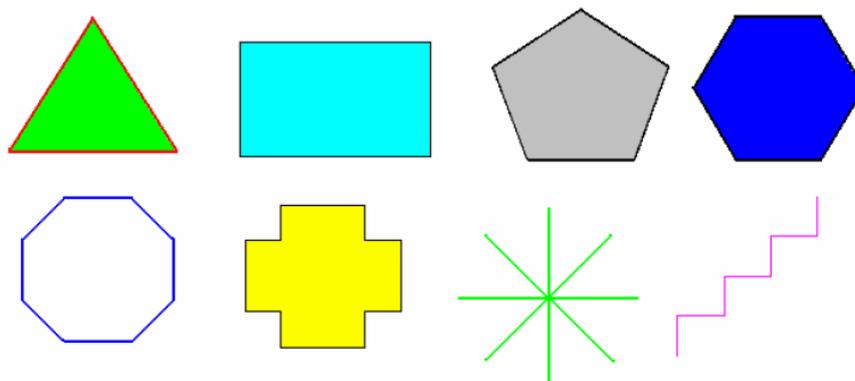


Figura 12: Exercício com formas geométricas

Os alunos descobriram durante a execução das atividades, novas possibilidades, através da tentativa e erro e através da busca de soluções pela criação de novos itens. Na tentativa, novas figuras surgiram e eles apresentaram suas descobertas.

A turma manteve-se interessada nas descobertas que faziam.

Após esse momento inicial, foi sugerida a criação de um Castelo como apresentado na figura abaixo (SANTOS, 2006, p.7). Como se trata de um exercício bastante complexo, propus o desafio e após algum tempo, apresentei a solução para que pudessem comparar e finalizar o exercício. Alguns alunos usaram a solução como consulta para correção, mas tentaram elaborar cada passo testando e verificando suas hipóteses. Aproveitei o momento para relacionar os comandos da Linguagem aos comandos aprendidos em português e focar o uso do comando *aprenda* que se traduziria, nas aulas seguintes em procedimentos e funções do Português Estruturado.

As figuras 13 e 14 apresentam os resultados que deveriam ser produzidos na tela após a proposta desta atividade.

Exemplo: Construindo um castelo

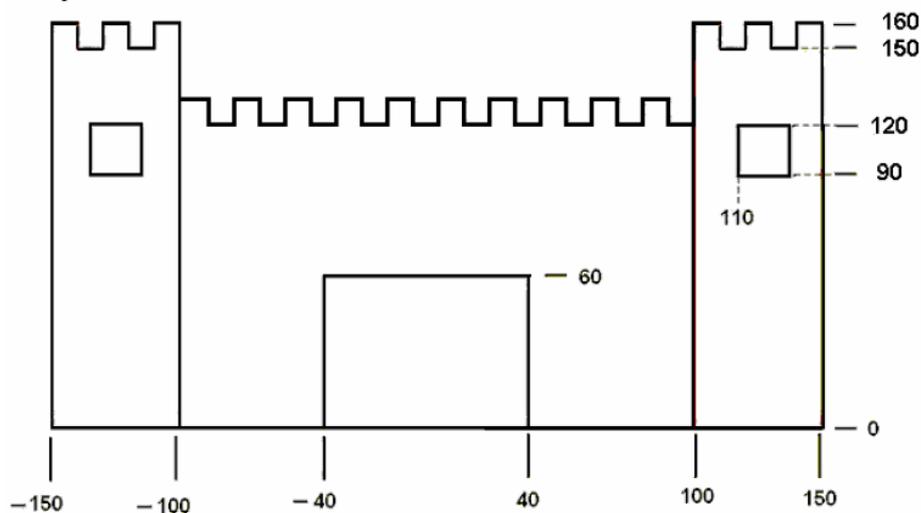


Figura 13: Construindo um castelo

Resultado Final

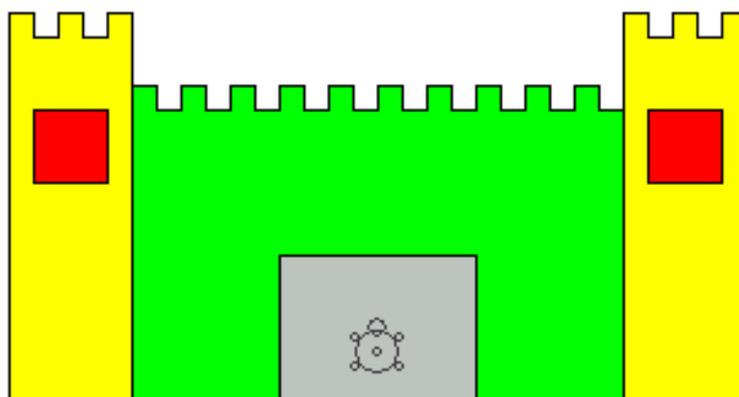


Figura 14: Resultado Final do Castelo

No dia 20/05/2014, pretendia continuar trabalhando com o LOGO, porém não foi possível usarmos o laboratório, assim, destinamos a aula a solucionar alguns problemas complexos e tivemos que retomar ao Português Estruturado.

No dia 27/05/2014 houve uma greve de ônibus no município e compareceu à aula cerca de 30% da turma, sendo assim, só foram trabalhados exercícios e foi um momento de esclarecimento de dúvidas.

Na semana do dia 03/06/2015 não foi possível novamente utilizarmos o laboratório. Essas 3 aulas que ocorreram sem a possibilidade de dar continuidade ao trabalho com o LOGO, fizeram com que alguns alunos não assimilassem tal linguagem e não a percebesse como uma ferramenta útil ao seu aprendizado.

Neste encontro foi possível aprofundar o conceito de procedimentos e funções discutidas com o SuperLOGO e, na segunda parte da aula aplicar o segundo teste diagnóstico, que apresentamos na íntegra no Apêndice.

O dia 10/06/2014 acabou representando nossa última aula do semestre, já que as aulas foram suspensas no dia 17/06 devido aos jogos da Copa do Mundo. Por conta disso, resolvi aproveitar o tempo para enfatizar a linguagem LOGO através do SCRATCH.

Após apresentar suas características principais e as possibilidades da ferramenta, propus que os alunos desenvolvessem aplicações que poderiam ser jogos ou animações. Durante o período de aulas eles desenvolveram algumas aplicações simples e um dos usou seu tempo livre para criar jogos mais complexos que foram compartilhados no grupo do Facebook mesmo após o término das aulas.

O dia 24/06 foi reservado para revisão geral do conteúdo e esclarecimento de dúvidas, já que pelo calendário oficial da instituição, as semanas seguintes estariam reservadas para a Segunda Avaliação, Segunda Chamada e Prova Final respectivamente.

CAPÍTULO 6

ANÁLISE DOS RESULTADOS

A partir do questionário 2, que pode ser verificado no Anexo 4, foi possível compreender a percepção dos alunos acerca da utilização da Linguagem LOGO e do aprendizado de Lógica de Programação.

Sobre o fato de se eles consideram que o LOGO contribuiu para as demais disciplinas cursadas após o primeiro período, 20 responderam que sim, conforme mostra o gráfico 5. Percebemos, após algumas análises, que isso não tem relação com uma vivência positiva ou negativa, apenas com a opinião pessoal de cada um.

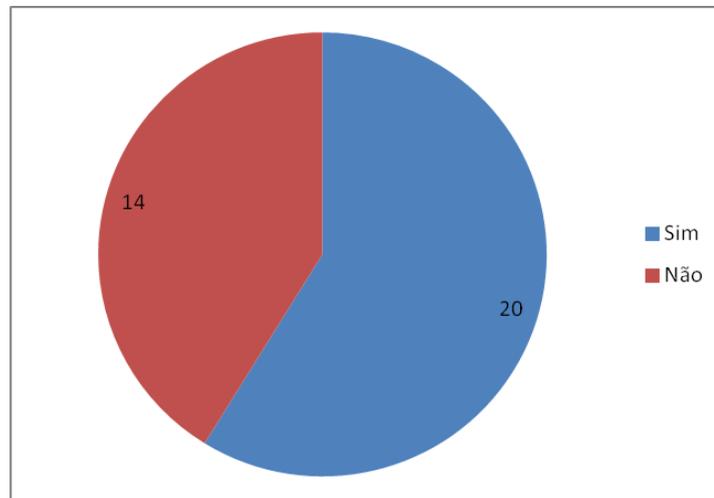


Gráfico 5 - O LOGO contribuiu para as demais disciplinas que está cursando neste período?

Dos 34 alunos entrevistados, 7 não gostaram de ter conhecido Linguagem LOGO na disciplina Algoritmos e Programação I, como mostra o gráfico 6, tal relato não teve nenhuma relação com a Linguagem em si, mas com o fato de acharem simples demais e, por isso, desnecessária, sem aplicação no meio acadêmico e, segundo o Aluno 16, "*retardou as técnicas de programação*". Isso não é uma realidade já que a Linguagem nos permite trabalhar com todas as estruturas de lógica de programação trabalhadas em Português estruturado, porém de uma forma menos formal e visando desenvolver a estrutura de raciocínio lógico. Deve-se ressaltar que esse aluno, ao iniciar o curso, já possuía conhecimentos anteriores embora não fossem avançados.

Verificamos que dos alunos que relataram não gostar, apenas o Aluno 21 não possuía qualquer conhecimento anterior relacionada à programação de computadores e 6 consideram que o LOGO não contribuiu de forma significativa para as disciplinas

cursadas no segundo período. O Aluno 29 foi o único que, apesar de não ter relacionado LOGO à uma vivência positiva, considera sua contribuição para outras disciplinas.

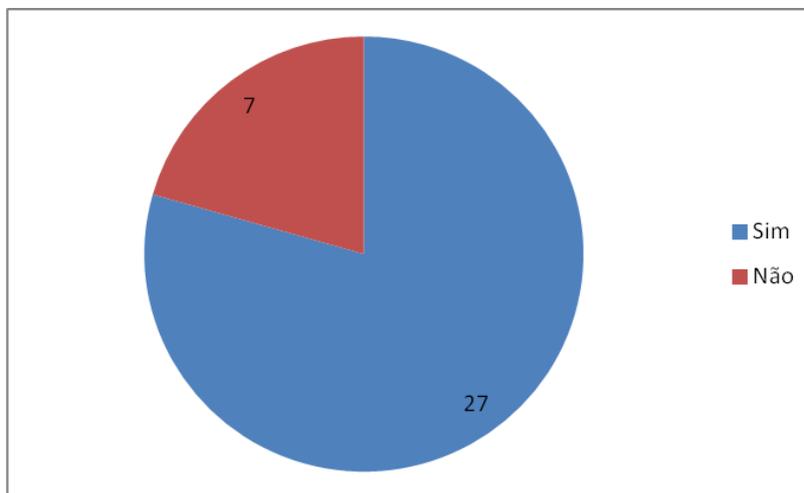


Gráfico 6 - Você gostou de conhecer a Linguagem LOGO?

O Aluno 14 relata não ter se identificado com "*a forma de programação*" do LOGO e excluiu o LOGO dos elementos marcantes de sua vivência, porém menciona seu desejo em ter uma experiência/conhecimento maior sobre programação durante o curso. Este aluno trabalha com infraestrutura de redes e já realizou cursos anteriores.

Para os alunos, em geral, o primeiro período aprimorou o conhecimento anterior sobre programação, apenas 4 disseram que manteve-se o mesmo. Três desses alunos cursaram ensino médio técnico em informática ou cursos livres na área de programação. Todos trabalham na área de suporte e infraestrutura.

Dos entrevistados, 13 informaram que seu conhecimento foi aprimorado através do uso da Filosofia LOGO. Independente da experiência anterior, todos relataram que a experiência no primeiro período foi boa para rever e ampliar conceitos já conhecidos. O Aluno 12 relata seu desejo em realizar exercícios mais complexos.

Dezessete alunos afirmaram não ter nenhum conhecimento anterior. O relato da maioria foi positivo, mas alguns alunos não tiveram uma boa vivência. O Aluno 3, afirma não ter participado de muitas aulas, mas avalia o método de ensino como "*construtivo*". O Aluno 11 e o Aluno 26 mencionam que o conteúdo ministrado não supriu suas expectativas. Aqui não se referem exclusivamente ao LOGO, mas ao conteúdo da ementa da disciplina como um todo.

O Aluno 24 avalia sua experiência como difícil, afirmando que foi o primeiro contato com programação, mas a considera muito importante.

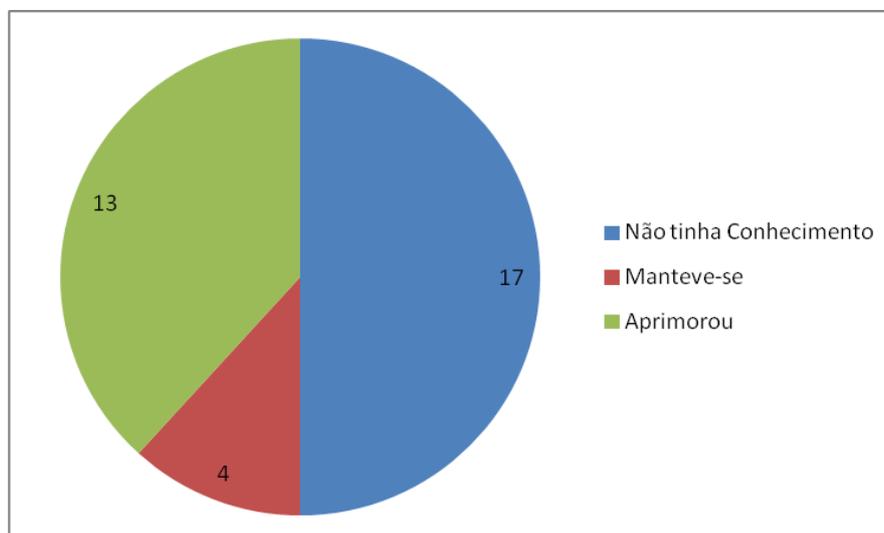


Gráfico 7 - O conteúdo do primeiro período aprimorou seu conhecimento anterior sobre programação?

Percebemos assim, que o público alvo é relativamente homogêneo no que se refere aos fatores principais mencionados como: sexo, idade, local de moradia, percepção da experiência, ou seja, a maioria é do sexo masculino, reside no mesmo bairro da instituição e acredita que a experiência contribuiu para seu aprendizado, tendo-a como positiva. Alguns que ressaltaram pontos negativos estavam relacionados a conhecimentos prévios e a crença de que não perceberam elementos motivadores que trouxessem novidade para eles.

Um dos alunos, o Aluno 23, já conhecia a Linguagem LOGO na versão dos anos 80 como recurso para integrar as crianças na sala de aula. Ele destaca como marcante a utilização do software SCRATCH. Ele considera, que, com base nos conhecimentos anteriores, a disciplina ofereceu um conteúdo "*muito básico*" e "*por ter experiência não acrescentou muito*", considerando que seu conhecimento manteve-se o mesmo de antes de ingressar no curso.

Entretanto, por iniciativa própria, após o término do semestre, ele construiu jogos usando o SCRATCH, um desejo mencionado durante os encontros, já que havia projetado jogos no passado e não conseguiu, na época, programá-los.

Um dos jogos construídos e compartilhados com o grupo foi o Megamania do Atari, conforme apresentado na figura 14. Além do jogo ele ainda elaborou um tutorial

detalhado para quem desejasse recriá-lo. Durante seus relatos, aponta que o mais interessante em todo o processo foi o auxílio prestado aos seus colegas de classe. Pela sua fala, temos a impressão de que ele entende o LOGO e o SCRATCH como linguagens distintas.



Figura 15: Tela do Jogo Megamania

Os demais alunos foram favoráveis a utilização do LOGO. O Aluno 1 questiona: "*Me pergunto por que não temos essa linguagem no ensino fundamental*". De fato, Papert desenvolveu essa linguagem para trabalhar com crianças em séries que hoje denominamos Ensino Fundamental. O uso da mesma contribuiria para o desenvolvimento do raciocínio lógico matemático e desenvolvimento de estruturas necessárias não só para a programação de computadores, mas para o conhecimento matemático. A percepção deste aluno foi interessante, pois ele percebeu que se a lógica de programação fosse enfatizada no ensino fundamental, a dificuldade existente no ensino superior poderia ser minimizada.

Outros alunos relataram a importância desta Linguagem no que se refere a facilitar o conhecimento em relação ao conteúdo da disciplina em questão. O Aluno 8 explica que "*foi muito relevante pra me estimular a programar*", já o aluno 25 "*Importante, pois explicita todos os caminhos minimamente detalhados dos passos a*

serem dados", o aluno 33 ressalta que "*ajuda a aprimorar o entendimento lógico*", o que mostra que eles puderam perceber ganhos significativos com a experiência, não só a linguagem LOGO, mas a Filosofia LOGO e todo o processo.

Eles viram a oportunidade de desenvolver jogos, trabalhar com robótica, acrescentar como currículo e até mesmo usar como forma de lazer. É válido ressaltar que dois alunos informaram conhecer a linguagem e que outros dois informaram o desejo de ter mais tempo com a linguagem.

O Aluno 34 destacou o LOGO como um bom recurso didático para o ensino de programação, e os alunos 33 e 17, abordaram o fato da boa didática e do domínio do conteúdo da professora, ainda assim, acreditamos que sem a estrutura e planejamento baseado na Filosofia LOGO não obteríamos um resultado positivo.

Alguns alunos perceberam que a disciplina lhes deu uma noção sobre a área de TI e sobre a área específica de programação, trazendo mais conhecimento no aspecto geral e o desejo de aprender mais. Apenas os alunos 20 e 28 desprezaram a experiência, isso justifica-se pelo fato de terem conhecimento anterior e terem realizado outros cursos. O Aluno 20 conhece as principais linguagens de programação e já fez e o Aluno 28 afirmou ter desenvolvido sistemas para fins acadêmicos.

6.1. Elementos Marcantes

Ao questionar os alunos sobre o que foi marcante (ver questão 11 do anexo 4), visto que a maioria deles não possuía qualquer conhecimento, os relatos mencionaram a estrutura dos programas, a organização dos mesmos, o Português Estruturado, entre outros elementos isolados do conteúdo. Verificamos que as respostas estavam enquadradas em sete categorias: **Respostas Negativas**, **Respostas Positivas simples**, **Respostas Neutras**, **Esperavam mais**, **Abordaram o aprimoramento do conhecimento**, **Novidade** e **Outras Percepções**.

O número de respostas negativas, ou seja, que relacionaram a vivência com algo ruim foi três. Enquadrava-se na categoria das respostas positivas simples as de alunos que consideraram-na boa, de maneira geral. Foi nove o número de respostas positivas simples. Consideramos respostas neutras como aquelas de alunos que relacionavam o

que foi vivenciado apenas com o Português estruturado, ou o VisualG. Oito respostas foram consideradas neutras.

Neste grupo de respostas nenhum aluno expôs o desejo de ter mais conhecimento acerca do conteúdo.

Quatro alunos consideraram sua vivência como elemento de aprimoramento do conhecimento e relataram isso de forma clara. Um dos alunos relacionou sua boa experiência com o desempenho nas disciplinas do segundo período.

Relacionamos à novidade, as respostas que mencionaram tudo que foi vivenciado como elemento marcante e importante para seu desenvolvimento profissional e acadêmico. Foram oito as respostas nesta categoria.

Consideramos outras percepções, alunos que mencionaram pontos que não tinham relação com o conteúdo da disciplina em si, mas, itens específicos como foi o caso do Aluno 23 que percebeu o SCRATCH como uma boa ferramenta para criação de jogos e os Alunos 17 e 33 que atribuíram sua vivência positiva à metodologia de ensino da disciplina e relacionaram este elemento com a professora.

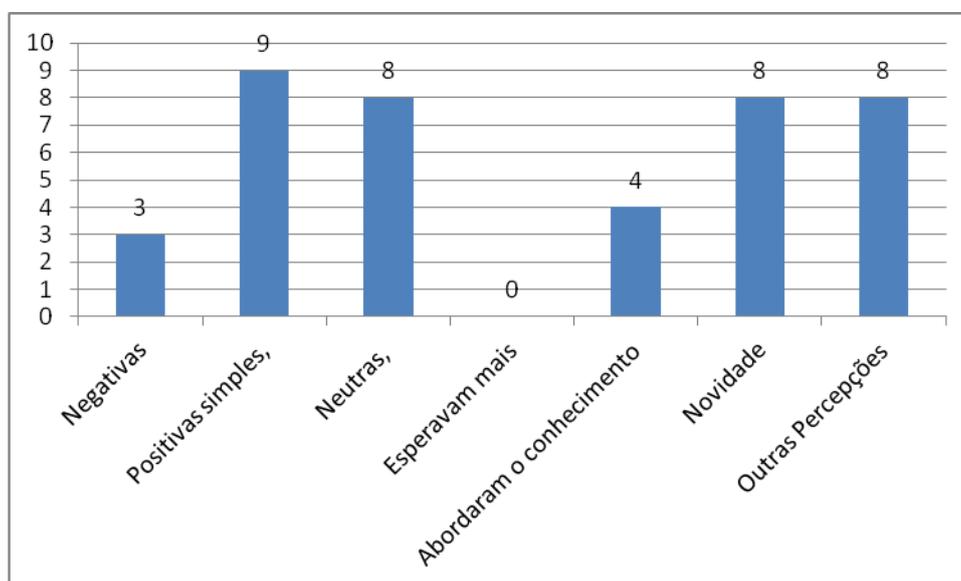


Gráfico 8 - Categorias das Questões

Os alunos 4 e 10 reconheceram que o conhecimento obtido nas aulas "*foi fundamental e está sendo útil*". O aluno 24 considera que consegue desenvolver os conteúdos do segundo período por causa do que foi aprendido no primeiro período, o

aluno 15 explica que "*o desenvolvimento do raciocínio lógico foi de suma importância nas demais disciplinas, mas também para o nosso cotidiano*".

As questões 12 e 13 foram analisadas em conjunto e utilizando as mesmas categorias descritas anteriormente. A questão 12 abordava sobre o fato dos alunos terem gostado da Linguagem LOGO e a questão 13 sobre o que acharam de ter conhecido tal Linguagem.

Isolamos as respostas dos alunos que gostaram da Linguagem e verificamos que dos sete que não gostaram da vivência, quatro também posicionaram-se negativamente a respeito da linguagem. Das duas respostas positivas simples que tivemos deste grupo, o Aluno 17 diz ser uma linguagem interessante, mas não para seu uso. Já o Aluno 21 diz que tudo que vivenciou foi muito bom. A resposta do Aluno 23 foi considerada neutra por não representar uma contribuição interessante, pois refere-se somente ao fato que já conhecia a linguagem.

Dos alunos que gostaram de ter conhecido o LOGO, excluímos 3 respostas da análise, pois embora fossem positivas, não apresentaram elementos que nos permitisse uma análise.

Oito respostas foram classificadas como positivas simples, avaliando a vivência como "*boa*" e "*interessante*". Duas respostas foram negativas, porque mesmo os alunos tendo gostado da Linguagem a consideram "*desnecessária*".

Dois alunos, o Aluno 9 e o Aluno 34, esperavam mais, sugerindo mais tempo com a ferramenta.

Sete respostas caracterizaram a vivência como algo novo, que lhes deu novas perspectivas sobre o que poderia ser feito com programação.

Das respostas que abordavam o aprimoramento do conhecimento, cinco alunos em citaram a linguagem como importante, facilitadora do aprendizado, estimulante ou como elemento que ajudou a aprimorar o entendimento da lógica.

Três dos alunos foram enquadrados como "*Outras percepções*", considerando o potencial da Linguagem para desenvolvimento de Jogos, conhecimento que poderia ser acrescentado no currículo, lazer e até revelação de novos horizontes na área de atuação. O Aluno 34 explica que a Linguagem, para ele, é "*um bom recurso didático*".

A questão 15 permitia que os alunos expusessem opiniões sobre itens não abordados no questionário. Onze alunos declararam não ter mais nada a relatar. O

Aluno 34 ressaltou a importância de ter sido trabalhado tópicos e atividades que auxiliaram na disciplina.

Oito respostas foram enquadradas como positivas por reafirmarem que os alunos gostaram do que vivenciaram. Cinco ressaltaram que esperavam mais da disciplina e das próximas que cursarão. Quatro respostas destacavam que consideraram muito importante todo o processo por se tratar do conhecimento de elementos novos. E, por fim, as quatro que enquadrámos como "Outras percepções" foram enfáticas em afirmar que todo o processo serviu para confirmarem a importância e o desejo de permanecer na área de tecnologia da informação.

As respostas combinadas nos permitiram compreender a percepção dos alunos sobre todo o processo. Consideramos a análise acima para tal verificação.

6.2. Análise dos Testes Diagnósticos

Como descrito na Metodologia da Pesquisa, foram realizados dois testes diagnósticos um no início, quando os alunos iniciaram as aulas e outro após o trabalho com a Filosofia LOGO e o conteúdo da disciplina¹⁵. Aqui iremos analisar o primeiro e o segundo teste, comparando com as características do aluno e apresentando as conclusões individuais de maneira sucinta. Pelo tamanho da turma não foi possível analisar individualmente cada aluno durante as aulas, sendo assim, lançamos mão destes instrumentos como recursos de análise.

No primeiro teste (Ver Anexo 5) apresentamos duas questões simples, onde o aluno deveria apresentar a solução do problema com a estrutura de raciocínio que desejasse, não deveria se prender a nenhuma linguagem ou metodologia, o foco era solucionar o problema, descrevendo os passos necessários para essa solução.

Já no segundo teste (Ver Anexo 6) as questões possuíam o mesmo contexto do primeiro porém havia mais complexidade e a necessidade de utilizar o conteúdo que foi trabalhado em sala de aula.

Quando os testes foram estruturados, foi pensado no que desejaríamos analisar, ou seja: através do teste 1, verificar quais alunos possuíam conhecimento das estruturas de programação ou de parte delas e, ainda, se eram capazes de estruturar logicamente

¹⁵ Os testes podem ser verificados nos Anexos 5 e 6

seu pensamento na solução de problemas. Para o segundo teste acrescentamos mais complexidade e nos baseamos nas estruturas de programação que foram trabalhadas durante as aulas, a fim de verificar se houve aprimoramento tanto em termos de conhecimento das estruturas quanto em estruturação lógica do pensamento.

Categorizamos os testes e suas respectivas respostas a fim de organizá-los de forma adequada. Separamos assim, (1) os alunos que solucionaram adequada ou parcialmente o primeiro teste e o segundo, (2) alunos que solucionaram adequada ou parcialmente somente o primeiro, (3) alunos que não solucionaram adequada ou parcialmente somente o segundo teste e (4) alunos que solucionaram adequada ou parcialmente apenas a primeira questão dos dois testes.

As observações sobre os alunos que são apresentadas a seguir são aquelas dos quais obtivemos as informações mais significativas em todos os sentidos, ou seja, responderam o questionário que delineou o perfil da turma e aos dois testes diagnósticos.

(1) os alunos que solucionaram adequada ou parcialmente as duas questões dos dois testes

<p>Aluno 1 - Sexo: Masculino, 23 anos, com conhecimento anterior Deslocamento Diário: Trabalha, reside e estuda em locais distintos</p>
<p>Percepção do Aluno Possui conhecimentos anteriores de Programação fez curso técnico em Informática. Considera que o LOGO não contribuiu para as disciplinas dos períodos posteriores, mas achou interessante o trabalho e mencionou que tal disciplina deveria ser trabalhada no ensino fundamental. Afirma que houve aprimoramento de seus conhecimentos ao longo da experiência e que relembrou elementos de programação que havia esquecido.</p>
<p>Percepção da Pesquisadora O aluno demonstra conhecer o conteúdo e é participativo em todas as aulas. Está sempre presente e envolvido em todas as atividades. Durante as aulas teve dúvidas no desenvolvimento de algumas atividades, mas que foram sanadas rapidamente.</p>
<p>Análise dos Testes</p> <p>Teste1 Questão 1: Solucionou adequadamente. Atingiu os objetivos da questão. Questão 2: Ele compreendeu a questão, mas não conseguiu solucioná-la adequadamente. Resolveu apenas uma parte da questão.</p>

<p>Teste2 Questão 1: Solucionou adequadamente. Atingiu os objetivos da questão. Questão 2: Solucionou adequadamente. Atingiu os objetivos da questão.</p>
<p>Conclusão O aluno além de dedicado já conhecia lógica de programação. Percebemos que ele já possuía estruturas lógicas construídas ao ingressar no curso e que, de fato essas estruturas foram aprimoradas na solução de problemas, o que é revelado pela própria escrita das soluções dos testes. A questão 2 dos testes estavam relacionadas e inicialmente ele não conseguiu formular uma solução satisfatória, mas no segundo teste, onde havia elementos mais complexos, ele fez isso adequadamente e com êxito.</p>

Tabela 3: Análise do Aluno 1

<p>Aluno 5 - Sexo: Masculino, 25 anos, com conhecimento anterior Deslocamento Diário: Não trabalha</p>
<p>Percepção do Aluno Em conversa informal o aluno relatou que iniciou a graduação em Sistemas de Informação em outra Universidade, mas devido a suas dificuldades resolveu reiniciar em outra instituição. Já possui conhecimentos anteriores de programação mas não trabalha na área. Achou interessante o trabalho com LOGO, mencionando que foi muito simples o aprendizado e serviu de base para outras disciplinas e que contribuiu para aprimorar o conhecimento que já possuía.</p>
<p>Percepção da Pesquisadora O aluno esteve presente em todos os encontros e procurou executar todas as atividades. Algo que o prejudicava bastante é o fato de possuir deficiência auditiva e, na maioria dos casos usar leitura labial, o que faz com que diminua sua compreensão do conteúdo.</p>
<p>Análise dos Testes</p> <p>Teste1 Questão 1: Houve uma tentativa de solução, mas não solucionou o problema. Questão 2: O aluno apresentou cálculos que nada representavam.</p> <p>Teste2 Questão 1: Solucionou adequadamente apresentando erros de sintaxe. Questão 2: Iniciou a solução, mas não concluiu, não demonstrou uma lógica de raciocínio.</p>
<p>Conclusão O aluno demonstrou uma pequena evolução, mesmo tendo estado presente nas aulas e participado das atividades. Acreditamos que o maior empecilho é sua deficiência auditiva visto que ele parece não acompanhar o que está sendo discutido mesmo usando um aparelho auricular. Durante as aulas, apesar do relato de estar compreendendo o que estava sendo trabalhado, não foi o que foi demonstrado na prática.</p>

Tabela 4: Análise do Aluno 5

<p>Aluno 7 - Sexo: Masculino, 21 anos, com conhecimento anterior Deslocamento Diário: Trabalha no mesmo local que reside, porém distante da Faculdade</p>
<p>Percepção do Aluno Já possuía conhecimentos anteriores de Programação, pois fez curso técnico em informática. Não trabalha na área. Achou interessante o trabalho com LOGO, e sua lógica simples.</p>
<p>Percepção da Pesquisadora O aluno esteve presente na maioria das aulas e se mostrou comprometido com a disciplina realizando todas as atividades.</p>
<p>Análise dos Testes</p> <p>Teste1 Questão 1: Solucionou adequadamente. Atingiu os objetivos da questão. Questão 2: Solucionou adequadamente. Atingiu os objetivos da questão.</p> <p>Teste2 Questão 1: Solucionou adequadamente. Atingiu os objetivos da questão. Questão 2: Iniciou a solução mas não terminou, abandonou o teste.</p>
<p>Conclusão O aluno demonstrou evolução mesmo já tendo trazido conhecimentos anteriores do curso técnico. As questões dos testes que foram respondidas estavam todas de acordo com o esperado.</p>

Tabela 5: Análise do Aluno 7

<p>Aluno 9 - Sexo: Masculino, 23 anos, com conhecimento anterior Deslocamento Diário: Trabalha, reside e estuda em locais distintos</p>
<p>Percepção do Aluno Já possuía conhecimentos anteriores de Programação. Não trabalha na área. Gostou de conhecer o LOGO, porém gostaria de ter tido mais tempo com a Linguagem, ainda assim, não percebeu utilidade.</p>
<p>Percepção da Pesquisadora O aluno esteve presente na maioria das aulas e se mostrou comprometido com a disciplina realizando as atividades.</p>
<p>Análise dos Testes</p> <p>Teste1 Questão 1: Solucionou adequadamente. Atingiu os objetivos da questão. Questão 2: Solucionou adequadamente. Atingiu os objetivos da questão.</p>

<p>Teste2 Questão 1: Solucionou adequadamente. Atingiu os objetivos da questão. Questão 2: Solucionou parcialmente. Tentou resolver o problema, desenvolveu uma lógica com início, meio e fim, mas não atingiu o objetivo proposto.</p>
<p>Conclusão O aluno demonstrou alguma evolução mesmo já tendo trazido conhecimentos anteriores. No decorrer do processo demonstrou muitas dúvidas, mas apresentou melhora.</p>

Tabela 6: Análise do Aluno 9

<p>Aluno 10 - Sexo: Masculino, 27 anos, com conhecimento anterior Deslocamento Diário: Trabalha distante da instituição e da residência, porém mora próximo da instituição</p>
<p>Percepção do Aluno O aluno já possuía conhecimentos anteriores de programação e já havia programado academicamente. Trabalha com tecnologia, porém não descreveu seu cargo. Gostou da experiência com Logo, mas o reconhece como uma ferramenta para iniciantes.</p>
<p>Percepção da Pesquisadora O aluno esteve presente na maioria das aulas e se mostrou comprometido com a disciplina realizando as atividades.</p>
<p>Análise dos Testes</p> <p>Teste1 Questão 1: Solucionou parcialmente, apresentando apenas a primeira parte da solução. Questão 2: Solucionou parcialmente, estruturou a lógica da solução, mas não apresentou o pedido (passo a passo).</p> <p>Teste2 Questão 1: Solucionou adequadamente. Atingiu os objetivos da questão. Questão 2: Solucionou parcialmente. Mais uma vez estruturou a lógica, mas não o algoritmo.</p>
<p>Conclusão O aluno demonstrou através do teste um avanço na estrutura, porém ainda um distanciamento entre a estruturação da lógica de pensamento e do programa em si.</p>

Tabela 7: Análise do Aluno 10

<p>Aluno 16 - Sexo: Masculino, 22 anos, com conhecimento anterior Deslocamento Diário: Trabalha, reside e estuda em locais distintos</p>
<p>Percepção do Aluno O aluno possuía algum conhecimento prévio. Trabalha com manipulação de Banco de Dados. Considera a experiência boa no geral, mas não gostou de ter trabalhado com o LOGO, por acreditar que sua lógica é muito simples e esperava mais da disciplina.</p>
<p>Percepção da Pesquisadora O aluno faltou algumas aulas importantes, mas nas que esteve presente se demonstrou comprometido.</p>
<p>Análise dos Testes</p> <p>Teste1 Questão 1: Solucionou adequadamente. Atingiu o objetivo proposto. Questão 2: Apresentou o passo a passo, mas não solucionou completamente o problema.</p> <p>Teste2 Questão 1: Solucionou adequadamente. Atingiu o objetivo proposto. Questão 2: Solucionou superficialmente, deixando muitos itens sem solução.</p>
<p>Conclusão O aluno demonstrou uma pequena evolução no contexto geral, que pode ser, em parte, atribuída às suas faltas. Como o aluno não esteve presente em todas as aulas não é possível afirmar que não se adaptou à metodologia.</p>

Tabela 8: Análise do Aluno 16

<p>Aluno 17 - Sexo: Masculino, 21 anos, com conhecimento anterior Deslocamento Diário: Trabalha, reside e estuda em locais distintos, apesar de residir na mesma região da instituição.</p>
<p>Percepção do Aluno O aluno já possuía conhecimento anterior, cursou técnico em informática. Já desenvolveu programas academicamente e conhece algumas linguagens de programação. Embora não tenha gostado do LOGO, diz que é uma linguagem interessante e menciona como elemento marcante a didática.</p>
<p>Percepção da Pesquisadora O aluno compareceu à maioria dos encontros e sempre se mostrou comprometido com a disciplina e as atividades propostas.</p>
<p>Análise dos Testes</p> <p>Teste1 Questão 1: Solucionou adequadamente. Atingiu o objetivo proposto.</p>

<p>Questão 2: Solucionou adequadamente. Atingiu o objetivo proposto.</p> <p>Teste2 Questão 1: Solucionou adequadamente. Atingiu o objetivo proposto. Questão 2: Não solucionou</p>
<p>Conclusão O aluno demonstrou uma pequena evolução no contexto geral, tanto na estruturação lógica do pensamento quanto dos algoritmos.</p>

Tabela 9: Análise do Aluno 17

<p>Aluno 21 - Sexo: Masculino, 26 anos, sem conhecimento anterior Deslocamento Diário: Trabalha e reside em locais próximos a instituição.</p>
<p>Percepção do Aluno O aluno não possuía qualquer conhecimento anterior. Embora não tenha gostado de conhecer o LOGO, considera relevante no contexto geral e positiva a experiência como um todo.</p>
<p>Percepção da Pesquisadora O aluno compareceu a maioria dos encontros e se mostrou interessado em solucionar as atividades propostas.</p>
<p>Análise dos Testes</p> <p>Teste1 Questão 1: Solucionou adequadamente. Atingiu o objetivo proposto. Questão 2: Solucionou adequadamente. Atingiu o objetivo proposto.</p> <p>Teste2 Questão 1: Solucionou adequadamente. Atingiu o objetivo proposto. Questão 2: Solucionou parcialmente, realizou as operações solicitadas, mas não fez os testes necessários.</p>
<p>Conclusão O aluno demonstrou uma grande evolução, especialmente porque não possuía nenhum conhecimento prévio.</p>

Tabela 10: Análise do Aluno 21

<p>Aluno 23- Sexo: Masculino, 40 anos, com conhecimento anterior Deslocamento Diário: Trabalha, reside e estuda em locais distintos, porém reside próximo à instituição.</p>
<p>Percepção do Aluno O aluno já possuía conhecimento anterior, já havia programado em linguagens e desenvolvido jogos. Já conhecia o LOGO e considerou a experiência interessante,</p>

mencionado o SCRATCH como diferencial e como ponto alto seu auxílio aos colegas.
<p>Percepção da Pesquisadora O aluno compareceu a maioria dos encontros e se mostrou interessado em solucionar as atividades propostas.</p>
<p>Análise dos Testes</p> <p>Teste1 Questão 1: Solucionou adequadamente. Atingiu o objetivo proposto. Questão 2: Solucionou adequadamente. Atingiu o objetivo proposto.</p> <p>Teste2 Questão 1: Solucionou adequadamente. Atingiu o objetivo proposto. Questão 2: Solucionou adequadamente. Atingiu o objetivo proposto</p>
<p>Conclusão O aluno já tinha um bom conhecimento anterior, sendo assim as aulas foram mais para rever conceitos que propriamente aprender novos. Ainda assim, como abordado anteriormente, ele usou os recursos do LOGO e SCRATCH para trabalhar ideias antigas de jogos, criou tutoriais e compartilhou com a turma.</p>

Tabela 11: Análise do Aluno 23

<p>Aluno 26 - Sexo: Masculino, 21 anos, com conhecimento anterior Deslocamento Diário: Trabalha, reside e estuda em locais próximos</p>
<p>Percepção do Aluno O aluno já possuía conhecimento anterior, fez curso técnico. Gostou de conhecer o LOGO, percebeu sua importância para o trabalho com robótica e considerou interessante a forma que a lógica foi utilizada.</p>
<p>Percepção da Pesquisadora O aluno compareceu a maioria dos encontros e se mostrou interessado em solucionar as atividades propostas.</p>
<p>Análise dos Testes</p> <p>Teste1 Questão 1: Solucionou adequadamente. Atingiu o objetivo proposto. Questão 2: Solucionou adequadamente. Atingiu o objetivo proposto.</p> <p>Teste2 Questão 1: Solucionou adequadamente. Atingiu o objetivo proposto. Questão 2: Solucionou superficialmente, não tratando o problema central.</p>
<p>Conclusão O aluno já tinha conhecimento anterior, mas houve uma evolução no que se refere às estruturas de programação e a estruturação lógica do pensamento.</p>

Tabela 12: Análise do Aluno 26

<p>Aluno 28 - Sexo: Masculino, 20 anos, com conhecimento anterior Deslocamento Diário: Reside e estuda no mesmo local, porém trabalha distante.</p>
<p>Percepção do Aluno O aluno já possuía conhecimento anterior, já havia programado em algumas linguagens, fez curso técnico e desenvolveu programas academicamente. Gostou de ter conhecido o LOGO, considera um ferramenta interessante, mas desnecessária. Mencionou que não houve nada novo baseando-se pelo que ele já conhecia.</p>
<p>Percepção da Pesquisadora O aluno compareceu a maioria dos encontros e se mostrou interessado em solucionar as atividades propostas.</p>
<p>Análise dos Testes</p> <p>Teste1 Questão 1: Solucionou adequadamente. Atingiu o objetivo proposto. Questão 2: Solucionou adequadamente, porém não se preocupou com os passos iniciais.</p> <p>Teste2 Questão 1: Solucionou adequadamente. Atingiu o objetivo proposto. Questão 2: Solucionou adequadamente. Atingiu o objetivo proposto</p>
<p>Conclusão O aluno já tinha um bom conhecimento anterior, sendo assim as aulas foram mais para rever conceitos que propriamente aprender novos. Durante as aulas foram feitos alguns ajustes na estrutura dos algoritmos (forma como ele desenvolvia), havendo assim um avanço mesmo que pequeno.</p>

Tabela 13: Análise do Aluno 28

<p>Aluno 31 - Sexo: Masculino, 19 anos, sem conhecimento anterior Deslocamento Diário: Não trabalha</p>
<p>Percepção do Aluno O aluno não possuía conhecimento anterior, já havia programado em linguagens e desenvolvido jogos. Gostou de conhecer o LOGO e considerou a experiência muito boa por ter assimilado praticamente o conteúdo todo.</p>
<p>Percepção da Pesquisadora O aluno faltou alguns dos principais encontros, mas se mostrou interessado em solucionar as atividades propostas.</p>
<p>Análise dos Testes</p> <p>Teste1 Questão 1: Solucionou parcialmente o problema. Questão 2: Solucionou parcialmente o problema não se preocupando com os passos</p>

<p>iniciais.</p> <p>Teste2 Questão 1: Solucionou adequadamente com alguns pequenos erros de estrutura. Questão 2: Solucionou apenas o problema superficialmente.</p>
<p>Conclusão O aluno teve uma grande evolução, pois inicialmente não tinha qualquer conhecimento de programação e, ao final, conseguia estruturar os algoritmos de maneira lógica mesmo tendo faltado a alguns dos principais encontros.</p>

Tabela 14: Análise do Aluno 31

<p>Aluno 33 - Sexo: Masculino, 22 anos, com conhecimento anterior Deslocamento Diário: Trabalha e estuda no mesmo local, mas mora em local distante</p>
<p>Percepção do Aluno O aluno já possuía conhecimento anterior, já havia programado em algumas linguagens e feito curso técnico em informática. Sua experiência com programação foi apenas o projeto de conclusão do ensino médio. Gostou de conhecer o LOGO e reconhece a linguagem como um recurso para aprimorar o entendimento lógico. Esperava mais da experiência, mas destacou a didática das aulas e o domínio da professora como elementos chaves no processo.</p>
<p>Percepção da Pesquisadora O aluno compareceu a maioria dos encontros e se mostrou interessado em solucionar as atividades propostas.</p>
<p>Análise dos Testes</p> <p>Teste1 Questão 1: Solucionou adequadamente. Atingiu o objetivo proposto. Questão 2: Solucionou adequadamente. Atingiu o objetivo proposto.</p> <p>Teste2 Questão 1: Solucionou adequadamente. Atingiu o objetivo proposto. Questão 2: Solucionou adequadamente. Atingiu o objetivo proposto.</p>
<p>Conclusão O aluno já tinha um bom conhecimento anterior, sendo assim as aulas foram mais para rever conceitos que propriamente aprender novos. Ainda assim, houve uma evolução satisfatória no que se refere a estruturação lógica para resolução de problemas.</p>

Tabela 15: Análise do Aluno 33

Conforme esperávamos, a maioria dos alunos teve um avanço considerável durante o processo, o que atribuímos à Filosofia LOGO pela sua contribuição ao aprimoramento do processo de aprendizado, facilitando a compreensão dos conceitos

inerentes à Lógica de Programação. Alguns alunos, mesmo não tendo qualquer conhecimento de programação foram capazes de estruturar logicamente uma solução.

Como podemos observar, os alunos 23 e 33 responderam adequadamente as duas questões dos dois testes. Tais alunos tem experiência com programação e dominam algumas linguagens de programação comerciais. Nesses casos, observei pessoalmente, que a evolução nos dois se deu no que se refere ao aumento de complexidade dos problemas e a adequação das respostas aos problemas solucionadas. Foram dois alunos assíduos e que estavam sempre auxiliando os colegas em suas soluções.

(2) alunos que resolveram adequada ou parcialmente somente o primeiro teste

Não identificamos alunos neste caso, sendo assim, ainda que alguns tenham avançado significativamente e outros não, concluímos que o processo de ensino permitiu que todos os alunos ampliassem seu conhecimento.

(3) alunos que solucionaram adequada ou parcialmente somente o segundo teste.

<p>Aluno 3 - Sexo: Feminino, 23 anos, sem conhecimento anterior Deslocamento Diário: Trabalha, reside e estuda em locais próximos</p>
<p>Percepção do Aluno Não possuía conhecimentos anteriores de Programação, mas leciona informática básica para educação infantil e ensino fundamental. Achou interessante o trabalho com LOGO, mencionando que facilitou o aprendizado de técnicas de programação.</p>
<p>Percepção da Pesquisadora O aluno esteve ausente na maioria das aulas e demonstra ter avançado muito pouco durante o período. Todas as atividades propostas foram elaboradas com muita dificuldade e desenvolvendo muito pouco mesmo com o auxílio dos colegas. Não concluiu o período.</p>
<p>Análise dos Testes</p> <p>Teste1 Questão 1: Não solucionou a questão alegando não saber fazer. Questão 2: Não solucionou a questão alegando não saber fazer.</p> <p>Teste2 Questão 1: Solucionou adequadamente o foco principal ignorando os detalhes iniciais (entrada de dados). Questão 2: Solucionou parcialmente, resolveu a questão de maneira superficial sem resolver o problema proposto.</p>

Conclusão

O aluno demonstrou uma pequena evolução embora sua dedicação tenha sido pequena. Acreditamos que se fosse mais presente nas aulas e se dedicasse ao estudo das técnicas de programação teria êxito.

Tabela 16: Análise do Aluno 3

Consideramos que o Aluno 3 teve um excelente avanço, pois como podemos observar no primeiro teste não conseguiu realizar nenhuma das duas questões e no segundo solucionou, mesmo que com algumas falhas, adequadamente. Na segunda questão do segundo teste, embora não tenha sido completamente compreendida, houve um esforço em solucioná-la.

(4) alunos que solucionaram adequada ou parcialmente apenas a primeira questão dos dois testes.

Aluno 13 - Sexo: Masculino, 24 anos, sem conhecimento anterior

Deslocamento Diário: Trabalha distante da instituição, mas estuda e reside em locais próximos.

Percepção do Aluno

Não possuía conhecimentos anteriores de Programação. Não trabalha na área. Mencionou que teve bons resultados no primeiro período, gostou de conhecer o LOGO e de toda a experiência.

Percepção da Pesquisadora

O aluno faltou algumas aulas importantes, mas nas que esteve presente se demonstrou comprometido.

Análise dos Testes

Teste1

Questão 1: Solucionou adequadamente apenas a primeira parte da questão.

Questão 2: Tentou solucionar o problema principal da questão, mas não abordou os passos necessários para se chegar lá.

Teste2

Questão 1: Apresentou a lógica adequada, porém estruturou o algoritmo com algumas falhas.

Questão 2: Não solucionou a questão.

Conclusão

O aluno demonstrou evolução no que se refere à estruturação lógica do pensamento para solução das questões. Embora tenha aprendido as estruturas de programação teve

dificuldades para aplicá-las mesmo no final do processo.

Tabela 17: Análise do Aluno 13

<p>Aluno 18- Sexo: Masculino, 26 anos, com conhecimento anterior Deslocamento Diário: Trabalha no mesmo local que reside, porém distante da instituição</p>
<p>Percepção do Aluno O aluno já possuía conhecimento anterior, cursou técnico em informática e realizou alguns cursos. Já desenvolveu programas academicamente e conhece algumas linguagens de programação. Gostou da experiência com LOGO, por considerar um novo método, mas gostaria de ter aprendido outra linguagem de programação ao mesmo tempo.</p>
<p>Percepção da Pesquisadora O aluno compareceu à maioria dos encontros e sempre se mostrou comprometido com a disciplina e as atividades propostas.</p>
<p>Análise dos Testes</p> <p>Teste1 Questão 1: Solucionou adequadamente, porém foi direto ao resultado, esquecendo-se dos passos intermediários. Questão 2: Solucionou parcialmente o problema, pois não se atentou para os passos iniciais.</p> <p>Teste2 Questão 1: Solucionou adequadamente. Atingiu o objetivo proposto. Questão 2: Não solucionou</p>
<p>Conclusão O aluno demonstrou uma evolução no contexto geral no que se refere à estruturação dos algoritmos. Desde o primeiro momento já apresentava uma estruturação lógica de pensamento coerente.</p>

Tabela 18: Análise do Aluno 18

<p>Aluno 19 - Sexo: Masculino, 20 anos, com conhecimento anterior Deslocamento Diário: Trabalha, reside e estuda em locais próximos</p>
<p>Percepção do Aluno O aluno já possuía conhecimento anterior, cursou técnico em informática. Já desenvolveu programas academicamente e conhece algumas linguagens de programação. Considera a experiência marcante como um todo.</p>
<p>Percepção da Pesquisadora</p>

<p>O aluno faltou alguns encontros importantes, mas se mostrou interessado em solucionar as atividades propostas.</p>
<p>Análise dos Testes</p> <p>Teste1 Questão 1: Solucionou adequadamente. Atingiu o objetivo proposto. Questão 2: Solucionou parcialmente o problema, pois não se atentou para os passos iniciais.</p> <p>Teste2 Questão 1: Solucionou adequadamente. Atingiu o objetivo proposto. Questão 2: Não solucionou.</p>
<p>Conclusão O aluno demonstrou uma evolução no contexto geral, tanto na estruturação lógica do pensamento quanto dos algoritmos.</p>

Tabela 19: Análise do Aluno 19

<p>Aluno 20 - Sexo: Masculino, 20 anos, com conhecimento anterior Deslocamento Diário: Reside próximo à instituição, mas trabalha em locais distante</p>
<p>Percepção do Aluno O aluno já possuía conhecimento anterior, pois aprendeu em cursos. Gostou de ter conhecido o LOGO, mas não considerou a experiência marcante porque esperava mais da mesma e achou que o conteúdo foi muito básico.</p>
<p>Percepção da Pesquisadora O aluno compareceu a maioria dos encontros e se mostrou interessado em solucionar as atividades propostas.</p>
<p>Análise dos Testes</p> <p>Teste1 Questão 1: Solucionou adequadamente. Atingiu o objetivo proposto. Questão 2: Não estruturou logicamente a solução e nem apresentou o passo a passo.</p> <p>Teste2 Questão 1: Solucionou adequadamente com alguns erros na estrutura do algoritmo. Questão 2: Não solucionou. Houve um esboço, mas nada além disso.</p>
<p>Conclusão O aluno demonstrou uma pequena evolução principalmente por já ter realizado cursos na área. Espera-se de alunos que já tiveram experiências anteriores com programação que tenham as estruturas lógicas de pensamento mais desenvolvidas que aqueles que não tiveram qualquer contato.</p>

Tabela 20: Análise do Aluno 20

<p>Aluno 22- Sexo: Masculino, 20 anos, sem conhecimento anterior Deslocamento Diário: Reside distante da instituição, mas não trabalha</p>
<p>Percepção do Aluno O aluno não possuía qualquer conhecimento anterior. Gostou de conhecer o LOGO e considerou a experiência interessante como um todo.</p>
<p>Percepção da Pesquisadora O aluno compareceu a maioria dos encontros e se mostrou interessado em solucionar as atividades propostas.</p>
<p>Análise dos Testes</p> <p>Teste1 Questão 1: Solucionou adequadamente. Atingiu o objetivo proposto. Questão 2: Solucionou parcialmente. Solucionou o problema principal sem se preocupar com os passos iniciais.</p> <p>Teste2 Questão 1: Solucionou adequadamente com alguns erros de estrutura. Questão 2: Não solucionou. Apenas esboçou.</p>
<p>Conclusão O aluno demonstrou uma evolução, embora não tenha atingido todos os objetivos propostos.</p>

Tabela 21: Análise do Aluno 22

<p>Aluno 25- Sexo: Masculino, 34 anos, sem conhecimento anterior Deslocamento Diário: Trabalha distante da instituição, mas reside e estuda em locais próximos.</p>
<p>Percepção do Aluno O aluno não possuía conhecimento anterior. Gostou de conhecer o LOGO por reconhecer que através dele podemos verificar os passos detalhados para a construção de algoritmos.</p>
<p>Percepção da Pesquisadora O aluno compareceu a maioria dos encontros e se mostrou interessado em solucionar as atividades propostas.</p>
<p>Análise dos Testes</p> <p>Teste1 Questão 1: Apresentou a solução através de equações matemáticas. Questão 2: Tentou solucionar o problema mas não levou em consideração o passo a passo. A solução ficou incompleta.</p> <p>Teste2</p>

<p>Questão 1: Solucionou parcialmente com erros na estrutura.</p> <p>Questão 2: Esboçou mas não solucionou a questão.</p>
<p>Conclusão</p> <p>O aluno demonstrou uma evolução embora não tenha atingido todos os objetivos propostos.</p>

Tabela 22: Análise do Aluno 25

Os Alunos 18, 19 e 20, já possuíam conhecimento em programação, mas seu conhecimento anterior não foi suficiente para que solucionassem nem a questão dois do primeiro teste, nem a do segundo teste. Concluimos que, por mais que declarem ter cursado cursos livres de programação ou o ensino médio técnico na área de computação, as aulas com disciplinas nesta área não foram suficientes para desenvolver as estruturas lógicas de pensamento. Seria importante investigar futuramente estes alunos para que fosse possível afirmar com precisão o motivo pelo qual houve pouca evolução.

Já os alunos 22 e 25, mesmo não tendo conhecimento anterior, apresentaram uma pequena evolução se comparada ao restante da turma.

6.3. A Percepção do professor de Algoritmos e Programação II

No segundo e terceiro períodos são oferecidas as disciplinas Algoritmos e Programação II e III que complementam Algoritmos e Programação I, disciplina na qual introduzimos o LOGO e o SCRATCH.

Nestas disciplinas o aluno programa diretamente em uma linguagem de programação específica e possui conceitos mais avançados que vão servir de base para outras disciplinas.

Enquanto os alunos que participaram de nossa pesquisa cursavam a disciplina de Algoritmos e Programação II, optamos por realizar uma entrevista com o Professor desta visando conhecer sua percepção acerca da turma.

O Professor que aqui chamaremos de Professor Carlos, começou a trabalhar com o ensino de programação em 1983 e trabalha na instituição desde 2004, além disso tem experiência como professor em outras instituições de ensino.

Quando Carlos foi questionado sobre como está a turma em relação ao aprendizado da disciplina, ele mencionou que "apesar de uma pequena evidência de

melhoria com relação às turmas anteriores, persiste a falta de embasamento lógico para resolver os problemas propostos". Ele descreve a turma como uma turma homogênea no que se refere a conhecimentos técnicos, onde apenas alguns se destacam apresentando um perfil adequado para programação.

Ao solicitarmos uma avaliação sobre o conhecimento em relação ao que deveriam ter aprendido no primeiro período, ele classificou como mediano, afirmando que houve uma tênue melhora nesta turma em relação às demais.

Por fim, questionamos sobre o LOGO, se o Professor a conhecia e se já havia pensado em trabalhar com a mesma. Em seu relato ele descreve que conhece e acredita que essa e outras ferramentas podem contribuir para o aprendizado de programação, embora haja fatores que dificultem, de um modo geral, o domínio de técnicas de programação, como a escola atual, que não tem seu foco em análise, crítica e solução de problemas, falta de habilidade interpretativa no que se refere ao que se deve fazer (compreensão do problema proposto) e a deficiência em matemática "que é primordial como base nos cursos da área tecnológica".

Ele encerra comentando que conteúdos de Lógica deveriam ser lecionados no Ensino Fundamental e no Médio, sugerindo o LOGO como uma ferramenta para isso.

CAPÍTULO 7

CONSIDERAÇÕES FINAIS

A presente pesquisa teve, como objetivo principal, verificar se, por meio da aplicação da Filosofia LOGO e com o apoio das Linguagens criadas para esse fim, os alunos tiveram um aprimoramento das suas estruturas lógicas de Pensamento.

As atividades propostas em forma de desafios e problemas reais para serem solucionados permitiram que os alunos se demonstrassem empenhados durante todo o processo e buscando ampliar seus conhecimentos para solucionar estes desafios.

Ao iniciar a pesquisa, acreditávamos que esta filosofia seria responsável pelo sucesso no que se refere ao desenvolvimento das estruturas lógicas de pensamento para solução de problemas relativos à programação de computadores.

Ao final de um semestre letivo de aulas de programação espera-se que, independente do método aplicado, os alunos ampliem seu conhecimento relacionado à lógica de programação. Para verificar o processo de cada aluno foram aplicados testes no início e no final do semestre.

Os testes aplicados verificaram o conhecimento técnico, como eles estruturavam inicialmente as soluções e como passaram a estruturá-las após certo período. Analisando o grupo como um todo, percebe-se que poucos alunos não tiveram uma evolução adequada. Fato esse que se deu por diversos motivos que relacionamos a seguir. Alguns alunos eram faltosos e não participaram das aulas e das propostas da disciplina. Existem pessoas que, por mais dedicação que tenha a uma determinada área do conhecimento, não tem aptidão para tal, o que não se refere ao método de ensino ou a qualquer outra questão didática e/ou metodológica. Outros ainda não têm facilidade de aprender de determinado modo e, mesmo que a metodologia seja adaptada e adaptável durante o processo, não é possível ter êxito com todos os alunos.

Percebemos que, independente dos alunos terem cursado ensino médio técnico em informática, alguns deles não foram capazes de estruturar as soluções mais complexas, o que nos leva a concluir que suas estruturas lógicas de pensamento ainda precisam amadurecer ou serem desenvolvidas. Já outros alunos, mesmo sem nenhum conhecimento de programação eram capazes de estruturar soluções que atingiam os objetivos. Concluímos que apenas a programação não é responsável por desenvolver

tais estruturas lógicas e que, um trabalho mais pontual em toda a extensão do ensino básico irá contribuir não só para aqueles que desejam seguir carreiras nas áreas de exatas, mas para solucionar questões do dia a dia. Não se pode afirmar que, mesmo tendo todo o grupo passado pelas mesmas experiências de aprendizado que, ao final, todos estarão no mesmo nível de desenvolvimento das estruturas lógicas necessárias à programação de computadores.

Outro ponto que não foi abordado pontualmente por nossa pesquisa, mas que ficou bastante claro no contexto geral é a capacidade do aluno interpretar as questões propostas e os enunciados dos problemas, o que muitas vezes não está relacionada às estruturas, mas a capacidade interpretativa relacionada à linguagem escrita. Podemos estabelecer três categorias distintas:

- (1) Alunos que não conseguem compreender o problema e por isso não conseguem solucioná-lo através de algoritmos
- (2) alunos que conseguem compreender, mas não solucioná-lo através de algoritmos
- (3) alunos que compreendem o problema e desenvolvem sua solução.

A utilização da Filosofia LOGO, embora não possa ser considerada como único fator, contribuiu para que o processo fosse mais prazeroso e mais eficiente. A linguagem por si só também não é suficiente para melhorar o processo de desenvolvimento das estruturas lógicas de pensamento, mas a aplicação de uma metodologia contextualizada ao ambiente acadêmico/escolar faz toda a diferença. Foi possível perceber que a turma se mostrou comprometida e disposta a realizar as atividades e buscar entre eles soluções ou novos caminhos para essas soluções.

No geral, é possível afirmar que houve um avanço significativo, embora menor que o esperado, no aproveitamento da disciplina e no uso das estruturas lógicas de pensamento e dos conceitos abordados em sala de aula para a construção de soluções para os problemas propostos.

Compreendemos que não é possível dar esta pesquisa por finalizada devido ao fato de estarmos tratando de uma disciplina dinâmica e que teve a aplicação de uma metodologia como forma de melhorar o processo de aprendizagem uma única vez. Percebemos, assim, alguns pontos importantes que precisam ser reformulados para que tenhamos cada vez mais êxito ao atingir nosso foco principal que é ampliar as estruturas lógicas de pensamento.

Acreditamos, que essa pesquisa representa uma contribuição para transformar o ensino não só da disciplina Algoritmos e Programação I, mas do próprio curso, em algo prazeroso, interessante, sem, com isso, perder seus objetivos de formação do aluno como profissional da área de tecnologia.

Acerca da reformulação, concluímos alguns pontos essenciais. A disciplina mencionada torna-se mais interessante do ponto de vista do aluno se houver prática por meio de alguma ferramenta que possibilite o teste das soluções. Com base nisso, compreendemos que é necessário que os alunos estejam alocados, em todas as aulas, em um laboratório com capacidade para acomodar, no máximo, dois alunos por computador. O uso de recursos multimídia como datashow favorece o processo.

Reformulamos a ementa e o conteúdo programático para adequá-los aos objetivos deste estudo. A nova ementa deverá conter os seguintes itens¹⁶: *Conceitos Básicos de Computação. Introdução ao raciocínio Lógico.* Algoritmos: conceituação, características, técnicas de construção de algoritmos. Tipos primitivos de dados, variáveis, constantes. Declaração de variáveis. Operações básicas: comandos de atribuição e operadores. Comandos de entrada e saída. Estruturas de Controle: Sequencial, Condicional e de Repetição. Conceitos de funções e procedimentos. *Resolução de problemas no contexto computacional. Técnicas de interpretação de problemas algorítmicos*¹⁷.

Proposta de Conteúdo Programático:

I - Conceitos Básicos de Computação

- 1.1. *Características do Computador*
- 1.2. *Entrada e Saída de Dados*
- 1.3. *Processamento*

II – Construção de Algoritmos

- 2.1. *Conceituando Algoritmos*
- 2.2. *Características dos Algoritmos*
- 2.3. *Técnicas de Construção de Algoritmos*

III - Elementos Básicos do Programa.

- 3.1. *Tipos Primitivos de Dados*
- 3.2. *Variáveis e Constantes*

¹⁶ Os elementos em itálico são os apresentam os conceitos novos para ampliação da ementa.

¹⁷ As técnicas de interpretação incluiriam um trabalho sobre a transformação de enunciados de problemas em soluções computacionais.

- 3.3. Operadores
- 3.4. Funções pré-definidas
- 3.5. Comandos de atribuição
- 3.6. Comandos de Entrada e Saída

IV – Estruturas de Controle do Fluxo de Execução dos Comandos

- 4.1. Estrutura Condicional *Se*
- 4.2. Estrutura Condicional *Caso*;
- 4.3. Estruturas de Repetição
 - 4.3.1. Para
 - 4.3.2. Enquanto...Faça
 - 4.3.3. Repita...Até

V – Procedimentos e Funções

- 5.1. Criação e chamada de procedimentos
- 5.2. Criação e chamada de funções

V – Resolução de Problemas Computacionais

- 6.1. *Identificando um problema*
- 6.2. *Técnicas para interpretação de problemas algorítmicos*
- 6.3. *Técnicas para resolução de Problemas*

O ideal é apresentar os elementos que permitam elaborar algoritmos e manter o foco maior na resolução de problemas que necessitem de tais elementos.

O uso da Linguagem LOGO deve ser iniciado nos primeiros contatos com o grupo, ou seja, no momento que os alunos ainda não tiveram contato com a teoria relativa à programação de computadores, fato que colabora com nossa conclusão a respeito da necessidade de alocação em um laboratório. Durante a pesquisa percebemos que, como tal linguagem trabalha as estruturas lógicas de forma simples, intuitiva e interativa, os alunos compreenderão sua contribuição para o processo.

Não podemos deixar de abordar a Filosofia LOGO como metodologia de Ensino para esta e outras disciplinas relacionadas à programação de computadores e análise de sistemas. A criação de um ambiente favorável à aprendizagem, onde o aluno possa estar imerso no conteúdo, mas ao mesmo tempo livre e assumindo o controle da construção do conhecimento, além de torná-lo mais independente favorece a aprendizagem e não permite que o aluno perca o foco.

Aliado ao uso da Linguagem LOGO, consideramos fundamental utilizar uma ferramenta capaz de testar as soluções em pseudolinguagem, como o VisualG, por exemplo, pois após a proposta dos desafios, ao buscar uma solução, se o aluno tiver uma ferramenta que propicie testes em busca da melhor solução com a possibilidade de verificar sozinho se tal solução atende ao que foi proposto e, ainda, mergulhar nesse processo aprofundando seus conhecimentos de maneira prática, fará com que esse aluno sintase impulsionado, dando a ele maior independência.

A Linguagem LOGO baseada nos princípios de sua Filosofia pode permear todo o processo visto que também permite gerarmos soluções com os conceitos de programação que constituem a ementa. O importante é que o aluno tenha autonomia para gerenciar seu aprendizado e escolher as ferramentas que lhe confira segurança na elaboração das soluções.

Percebemos que, durante as aulas, alguns alunos avançam mais que outros por já dominarem conceitos de programação ou por terem as estruturas lógicas de pensamento mais desenvolvidas, o que gera desinteresse. Seria interessante após conhecer o perfil da turma, que esta fosse dividida em grupos e fossem propostas atividades diferentes para cada grupo. As atividades seriam divididas em fáceis (nível 1), medianas (nível 2) e avançadas (nível 3) Cada aluno optaria por integrar o grupo no qual se sentisse mais à vontade e resolver as questões que estivessem mais adequadas ao seu nível de conhecimento. Seriam atividades independentes, não sendo necessário que o aluno passasse pelo(s) nível(is) mais simples caso já possuísse um conhecimento mais avançado.

Outro ponto essencial nessa mudança metodológica é a questão da avaliação. Sabemos que em qualquer curso superior é necessário que haja um processo avaliativo capaz de gerar uma nota que será a medida para verificar se o aluno está apto a prosseguir ou deve repetir o processo. Não podemos abrir mão da avaliação, mas para garantir que permaneçamos a utilizar os princípios da Filosofia LOGO é necessário uma reformulação. Sugerimos a criação de projetos e não de provas em papel. O conceito de desenvolver projetos objetiva apresentar um problema inicial e permitir que o aluno o desenvolva. Outra possibilidade é dar liberdade ao aluno para que escolha qual problema deseja solucionar com base nas suas experiências pessoais.

O aluno poderia, por exemplo, selecionar elementos reais do seu cotidiano e criar soluções baseadas em tais situações. Não seria o caso de deixá-lo completamente

livre, mas estabelecer parâmetros de como definir o problema e como estruturar sua resolução com base nos conceitos abordados em sala de aula.

Acreditamos que avaliar a evolução individual do aluno de forma qualitativa contribui para o empenho do aluno e torna a avaliação mais transparente e clara. Durante o processo foi possível verificar que temos os seguintes casos:

- Alunos que não possuíam nenhum conhecimento: (1) o aluno que saiu do conhecimento nível zero e atingiu os objetivos propostos, (2) o aluno que saiu do nível zero e superou os objetivos e (3) o aluno que não atingiu os objetivos, não atendendo aos requisitos mínimos para prosseguir.
- Alunos que possuíam algum conhecimento: (1) o aluno que manteve seus conhecimentos e atingiu os objetivos propostos, (2) o aluno que manteve seus conhecimentos e não atingiu os objetivos propostos e (3) o aluno que ampliou os seus conhecimentos atingindo ou superando as expectativas¹⁸.
- Alunos que já programam ou já conhecem o conteúdo previsto: (1) o aluno que manteve seus conhecimentos, (2) o aluno que ampliou seus conhecimentos.

O importante neste processo avaliativo é alocar os alunos nas categorias corretas, por isso a importância de conhecer a turma e realizar um acompanhamento individual. Não consideramos a medição, mas a nota substituída por conceitos, como por exemplo:

- EX - Excelente – aproveitamento entre 90% (noventa por cento) a 100% (cem por cento)
- MB - Muito Bom – aproveitamento entre 80% (oitenta por cento) a 89% (oitenta e nove por cento)
- B - Bom – aproveitamento entre 70% (setenta por cento) a 79% (setenta e nove por cento)
- S - Suficiente – aproveitamento entre 60% (sessenta por cento) a 69% (sessenta e nove por cento)
- I - Insuficiente – até 59% (cinquenta e nove por cento).

A sugestão apresentada é apenas um parâmetro para discussão. Pode-se estabelecer mais ou menos conceitos para representar de maneira mais adequada o resultado das avaliações. Vale ressaltar que as regras da instituição relacionadas à quantidade de avaliações, média de aprovação e o próprio regimento interno devem ser levados em consideração para estabelecimento desses conceitos.

Acreditamos, assim, que o processo de avaliação se torna mais justo, pois no método atual percebemos que alguns alunos desenvolvem muito bem suas atividades no

¹⁸ Acreditamos que não é possível que o aluno que já tenha um conhecimento anterior, amplie seu conhecimento e não atinja os objetivos.

dia a dia da sala de aula, mas não atingem a média nas avaliações. Muitas vezes interferem fatores psicológicos ou físicos, como: cansaço, problemas pessoais, ansiedade na realização de provas, entre outros. Medir o desempenho, na maioria dos casos, não reflete o que o aluno construiu ou não em termos de conhecimento.

Como trabalhos futuros sugere-se que a proposta realizada seja aplicada e verificada para que se comprove sua eficácia ou necessidade de novos ajustes. Um trabalho pontual e focado no aluno sendo realizado no primeiro período da faculdade pode minimizar a ideia de que os cursos da área de informática são cursos difíceis de serem concluídos e preparar melhor o aluno para os demais períodos, evitando também desistências nessa fase inicial.

REFERÊNCIAS

ALVES-MAZZOTTI, A. J.; GEWANDSZNAJDER, F. **O Método nas Ciências Naturais e Sociais: Pesquisa Quantitativa e Qualitativa**. São Paulo: Pioneira, 1998.

BALDISSERA, Adelina. **Pesquisa-ação: uma metodologia do “conhecer” e do “agir” coletivo**. Sociedade em Debate, v. 7, n. 2, p. 5-25, 2012.

BODGAN, R.; BIKLEN, S. **Investigação Qualitativa em Educação**. Porto: Porto Editora, 1999.

BRASIL. Ministério da Educação. Conselho Nacional de Educação. **RESOLUÇÃO CNE/CP 1, DE 18 DE FEVEREIRO DE 2002**. Diário Oficial da União, Brasília, 9 de abril de 2002. Seção 1, p. 31. Republicada de 4 de março de 2002. Seção 1, p. 8. Disponível em < <http://portal.mec.gov.br/cne/arquivos/pdf/CP012002.pdf>>

_____. Ministério da Educação. Conselho Nacional de Educação. **RESOLUÇÃO CNE/CP 2, DE 19 DE FEVEREIRO DE 2002**. Diário Oficial da União, Brasília, 4 de março de 2002. Seção 1, p. 9. Disponível em < <http://portal.mec.gov.br/cne/arquivos/pdf/CP022002.pdf>>

Coll, C. (1987). **As Contribuições da Psicologia para a Educação: Teoria Genética e Aprendizagem Escolar**. In: L. Banks-Leite (Org). Piaget e a Escola de Genebra. São Paulo: Cortez.

FEUC. **Projeto Pedagógico do Curso Bacharelado em Sistemas de Informação**. Rio de Janeiro: 2014. Disponível em: < <http://www.feuc.br/index.php/graduacao-bsi-projeto-pedagogico>>

FEUC. **Projeto Pedagógico do Curso Licenciatura em Informática**. Rio de Janeiro: 2013. Disponível em: < <http://www.feuc.br/index.php/graduacao-computacao-projeto-pedagogico>>

GOMES, A. et al. Aprendizagem de programação de computadores- dificuldades e ferramentas de suporte. **Revista Portuguesa de Pedagogia: Tecnologias Educacionais e da Comunicação**. Ano 42-2, 2008. Disponível em: <http://hdl.handle.net/10316.2/4672>. Acesso em: 30/03/2013.

GUIMARÃES, A. , LAGES, N. **Algoritmos e Estruturas de Dados**. 28ª tiragem. Rio de Janeiro: LTC, 1994.

LEITE, V. et al. VisuAlg: Estudo de Caso e Análise de Compilador destinado ao ensino de Programação. In: **TISE**, 2013. Santiago, Chile. Disponível em < <http://www.tise.cl/volumen9/TISE2013/637-640.pdf>>

LIMA, M. R.; LEAL, M. C. Motivação discente no ensino-aprendizagem de programação de computadores. **Revista Educação & Tecnologia**, v. 17, n. 1, 2013.

MACEDO, Lino. **Ensaios Construtivistas**. 3. Ed. São Paulo: Casa do Psicólogo, 1994.

PAPERT, S. **Logo: Computadores e Educação**. 3ª ed. São Paulo: Editora Brasiliense, 1988.

_____. **A máquina das Crianças: Repensando a Escola na Era da Informática**. Ed. Rev. Porto Alegre: ArtMed, 2008.

PIAGET, J. **A Epistemologia Genética**. Ed. Vozes, Petrópolis, RJ, 1972.

_____. **Biologia e Conhecimento**. (Guimarães, F.M., Trad.). Petrópolis: Vozes, 1973. (Original publicado em 1967).

_____. **Psicologia e Pedagogia**. Rio de Janeiro: Forense Universitária, 1982.

_____. **Seis Estudos de Psicologia**. 23 ed. Rio de Janeiro: Forense Universitária, 1998.

PIVA, Dilermando, et al. **Algoritmos e Programação de Computadores**. Rio de Janeiro: Elsevier, 2011.

RIZZINI, I., CASTRO, M. R. e SARTOR, C. D. **Pesquisando... Guia de Metodologias da Pesquisa Para Programas Sociais**. Rio de Janeiro: Universidade Santa Úrsula, 1999.

ROSA, Maurício. **Role Playing Game Eletrônico: uma tecnologia lúdica para aprender e ensinar matemática**. Rio Claro, SP: UNESP, 2004. Dissertação de Mestrado, Instituto de Geociências e Ciências Exatas, Universidade Estadual Paulista, 2004.

SANTANCHÈ, A.; TEIXEIRA, C. Integrando instrucionismo e construcionismo em aplicações educacionais através do Casa Mágica. In: **V Workshop de Informática na Escola– XIX Congresso da SBC**. 1999.

SANTOS, Viviane. **SuperLogo: Programação para o estudo de Geometria**. São Paulo: UNESP, 2006.

THIOLLENT, Michel. **Metodologia da Pesquisa-Ação**. São Paulo: Cortez, 1985.

TRIPP, David. **Pesquisa-ação: uma introdução metodológica**. Educação e pesquisa, v. 31, n. 3, p. 443-466, 2005.

VALENTE, J. A. **Computadores e conhecimento: repensando a educação**. 1ª ed. Campinas, NIED - Unicamp, 1993.

_____. (org). **O computador na sociedade do conhecimento**. Ed. Campinas: UNICAMP/NIED, 1999.

ANEXOS

Anexo 1 - Matriz Curricular Bacharelado em Sistemas de Informação

P	DISCIPLINA		CARGA HORÁRIA			
	CÓDIGO	NOME	PRESENCIAL	1	2	TOTAL
1°	0685	Algoritmos e Programação I	60	50	0	110
	0677	Oficina de Produção de Textos	60	50	0	110
	0678	Cultura e Sociedade	30	25	0	55
	0679	Mercado de Trabalho	30	25	0	55
	0422	Linguagem Brasileira de Sinais	30	0	0	30
	TOTAL			210	150	0
2°	0686	Algoritmos e Programação II	60	50	0	110
	0178	Arquitetura de Computadores	60	50	0	110
	0116	Lógica Matemática	30	0	0	30
	0002	Fund. Históricos e Fil. da Educ.	30	0	0	30
	0404	Métodos e Técnicas de Estudo	30	0	0	30
	TOTAL			210	100	0
3°	0687	Algoritmos e Programação III	60	0	0	60
	0415	Matemática Discreta	60	0	0	60
	0416	Análise de Sistemas	60	0	0	60
	0688	Programação Web	30	50	0	80
	0023	Psicologia da Educação	30	50	0	80
	TOTAL			240	100	0
4°	0183	Estruturas de Dados	60	0	0	60
	0689	Programação Orient. a Objeto	60	50	0	110
	0581	Educação Especial e Inclusiva	30	0	0	30
	0691	Projeto de Banco de Dados	60	0	0	60
	0234	Didática Geral	30	50	0	80
	0449	Informática Educativa	30	0	0	30
TOTAL			270	100	0	370
5°	0693	Projeto Aplicado – Sist. Desk.	60	0	0	60

	0438	História da Educação	30	0	0	30
	0702	Ética, Comp. e Sociedade	30	0	0	30
	0695	Banco de Dados	60	50	0	110
	0446	Sociologia da Educação	30	0	0	30
	0417	Didática do Ensino de Comp.	30	50	0	80
	0033	Estágio Orientado I	30	0	170	200
	TOTAL			270	100	170
6°	0185	Sistemas Operacionais	60	0	0	60
	0699	Redes de Computadores	60	0	0	60
	0697	Projeto Aplicado – Sist. Web	60	0	0	60
	0418	Educação Brasileira	30	50	0	80
	0420	Elaboração de Projeto	30	0	0	30
	0041	Estágio Orientado II	30	0	170	200
	TOTAL			270	50	170
7°	0709	Projeto Aplicado – Multimídia	60	0	0	60
	0706	Redes Para Internet	60	0	0	60
	0592	Educação Com. e Tecnologia	30	0	0	30
	0421	Amb. Virtuais de Aprendizagem	30	0	0	30
	0680	Hist. Cultura Afro-Brasileira e Ind.	30	0	0	30
	0494	Monografia	60	0	120	60
	TOTAL			270	0	120
Atividades Complementares					200	200
TOTAL GERAL			1740	600	660	3000

Anexo 2 - Matriz Curricular Bacharelado em Sistemas de Informação

P	DISCIPLINAS OBRIGATÓRIAS		CARGA HORÁRIA			
	CÓDIGO	NOME	PRESENCIAL	1	2	TOTAL
1º	0685	Algoritmos e Programação I	60	50	0	110
	0677	Oficina de Produção de Textos	60	50	0	110
	0678	Cultura e Sociedade	30	25	0	55
	0679	Mercado de Trabalho	30	25	0	55
	0197	Probabilidade e Estatística	30	0	0	30
	TOTAL			210	150	0
2º	0686	Algoritmos e Programação II	60	50	0	110
	0178	Arquitetura de Computadores	60	50	0	110
	0116	Lógica Matemática	30	0	0	30
	0184	Fund. de Sistemas de Informação	30	50	0	80
	0702	Ética, Computador e Sociedade	30	0	0	30
	0404	Métodos e Técnicas de Estudo	30	0	0	30
TOTAL			240	150	0	390
3º	0687	Algoritmos e Programação III	60	0	0	60
	0415	Matemática Discreta	60	0	0	60
	0416	Análise de Sistemas	60	0	0	60
	0688	Programação Web	30	50	0	80
	0185	Sistemas Operacionais	60	50	0	110
	0670	Prática Profissional A	15	0	15	30
TOTAL			285	100	15	400
4º	0183	Estruturas de Dados	60	0	0	60
	0689	Programação Orientada a Objeto	60	50	0	110
	0690	Análise Orientada a Objeto	60	0	0	60
	0691	Projeto de Banco de Dados	60	50	0	110
	0692	Sistemas Distribuídos	30	0	0	30
	0671	Prática Profissional B	15	0	15	30
TOTAL			285	100	15	400
5º	0693	Projeto Aplicado – Sistemas Desktop	60	0	0	60
	0694	Pesquisa e Ordenação	60	0	0	60

	0695	Banco de Dados	60	50	0	110
	0284	Engenharia de Software	60	50	0	110
	0696	Legislação Aplicada	30	0	0	30
	0672	Prática Profissional C	15	0	15	30
	TOTAL		285	100	15	400
6º	0697	Projeto Aplicado – Sistemas Web	60	0	0	60
	0698	Tópicos Avançados em Banco de Dados	60	0	0	60
	0699	Redes de Computadores	60	0	0	60
	0700	Qualidade de Software	60	0	0	60
	0701	Gestão Empresarial	30	0	0	30
	0673	Prática Profissional D	15	0	15	30
	TOTAL		285	0	15	300
7º	0704	Projeto Aplicado – Dispositivos Móveis	60	0	0	60
	0705	Gerência de Projetos de Software	60	0	0	60
	0706	Redes Para Internet	60	0	0	60
	0707	Empreendimentos Empresariais	30	0	0	30
	0708	Inovação em TI	30	0	0	30
	0420	Elaboração de Projeto	30	0	0	30
	0674	Estágio Curricular Obrigatório*	0	0	200	200
	TOTAL		285	0	200	485
8º	0709	Projeto Aplicado – Interf. Gráfica e Multimídia	60	0	0	60
	0287	Segurança e Auditoria de Sistemas	30	0	0	30
	0710	Gerência de Infraestrutura em TI	60	0	0	60
	0711	Planejamento Estratégico em TI	30	0	0	30
	0712	Negócios na Internet	30	0	0	30
	0713	Projeto Final	60	0	60	120
	TOTAL		270	0	60	330
Atividades Complementares			0	0	200	200
TOTAL GERAL			2130	600	520	3250

P = PERÍODO

1 = REDE DE LEITURA

2 = OUTRAS: ESTÁGIO ORIENTADO, TCC OU ATIVIDADES COMPLEMENTARES.

* NO 7º PERÍODO, TEMOS 200HS DE ESTÁGIO, POIS O ALUNO DEVE ENTREGAR RELATÓRIO DO ESTÁGIO CURRICULAR OBRIGATÓRIO AO PROFESSOR ORIENTADOR DE ESTÁGIOS NESTE SEMESTRE. NÃO SIGNIFICA QUE ELE IRÁ CURSAR TODA A CARGA NESTE PERÍODO, MAS SIM QUANDO O RELATÓRIO É AVALIADO E A CARGA TOTAL É LANÇADA NO SISTEMA.

Anexo 3 - Questionário 1

O objetivo deste questionário é verificar qual seu conhecimento em relação a programação. É importante que você se identifique.

Nome:

- Você tem algum conhecimento de programação?
- Quais as linguagens que você conhece? Diga o nome da(s) linguagem(s):
- Em que linguagem já programou? Como aprendeu? Ainda que tenha aprendido sozinho, essa informação é importante para nós.
- Qual recurso você usa para tirar as dúvidas que possui? Participa de alguma comunidade? Tem contato com outros programadores?
- O que você já desenvolveu?

Anexo 4 - Questionário 2

¹ Nome:	² Idade:	³ Sexo: () F () M
⁴ Município / Bairro do Rio que reside:	⁶ Você Trabalha? () Sim () Não	
⁵ Município / Bairro do Rio que trabalha:	⁷ Quantas horas por dia?horas	

⁸ **Você trabalha com computação? Descreva sucintamente quais são suas atividades.**

.....

.....

⁹ **Como você descreve sua experiência com programação no primeiro período?**

.....

.....

¹⁰ **Você considera que a aprendizagem com uso do LOGO no primeiro período contribui para as disciplinas que está cursando nesse período? () Sim () Não**

¹¹ **Nas aulas de Algoritmos e Programação I, o que foi marcante para você?**

.....

.....

¹² **Você gostou de conhecer a linguagem LOGO? () Sim () Não**

¹³ **O que achou desta experiência?**

.....

.....

¹⁴ **Você considera que o conteúdo do primeiro período aprimorou seu conhecimento anterior sobre programação?**

() Não tinha conhecimento anterior () Manteve-se igual () Aprimorou-se

¹⁵ **Você tem mais algo a dizer sobre essa experiência vivida no primeiro período?**

.....

.....

Anexo 5 - Teste Diagnóstico 2**FACULDADES INTEGRADAS CAMPO-GRANDENSES
BACHARELADO EM SISTEMAS DE INFORMAÇÃO
LICENCIATURA EM COMPUTAÇÃO**

Nome: _____ Turno: _____

Caros Alunos,

O objetivo deste teste é saber um pouco mais sobre sua experiência com lógica de programação e programação. Gostaria de lembrá-lo que este não comporá sua nota. Não use nenhuma linguagem ao responder as questões, apenas descreva os passos necessários para solucionar os problemas propostos. Caso não saiba como responder, nos dê essa informação.

1. Um aluno da turma de sistemas de informação que cursa o primeiro período obteve três notas em uma determinada disciplina. Como você faria para ensiná-lo a verificar sua aprovação, sabendo que a média deve ser maior ou igual a 6. Não esqueça que é necessário calcular a média.

2. A empresa WZT solicitou uma aplicação para seus caixas eletrônicos que realize operação de saque. No caixa existem notas de três valores, a saber: 10, 20 e 50 reais. Descreva os passos necessários para a realização de um saque. O que deve fazer a aplicação quando o usuário solicitar um dos 3 valores a seguir: 100, 80 e 25 reais.

Anexo 6 - Teste Diagnóstico 2**FACULDADES INTEGRADAS CAMPO-GRANDENSES
BACHARELADO EM SISTEMAS DE INFORMAÇÃO
LICENCIATURA EM COMPUTAÇÃO****TESTE 2: (Uso de pseudolinguagem)**

Nome: _____

1. Sou professora da disciplina de Sistemas Operacionais. Na próxima semana aplicarei um teste para os alunos da turma. Desejo saber a média entre esta nota (A2) e a nota da A1 que tenho registrada em papel para cada um dos alunos. Além disso, gostaria de informá-lo sua situação de acordo com as regras da instituição: alunos com média abaixo de 4 estarão reprovados, alunos com média maior ou igual a 4 e inferior a 6 estão em Prova Final, já os com média a partir de 6 estão Aprovados. O programa deve ser repetido até que o usuário seja respondido não a pergunta "Deseja continuar?".

2. A empresa WZT solicitou uma alteração da aplicação desenvolvida para seus caixas eletrônicos e agora deseja as operações: saque, emissão de saldo e depósito. No caixa existem notas de apenas dois valores, a saber: 20 e 50 reais. Como deve ser codificada tal aplicação?