

O LOGO COMO FERRAMENTA AUXILIAR NO DESENVOLVIMENTO DO RACIOCÍNIO LÓGICO – UM ESTUDO DE CASO

Scheila Wesley Martins¹ and Luiz Henrique de Andrade Correia²

Abstract - *This article describes an experiment carried out with the students of the first year of a computer science course using the LOGO Language as a support methodology for the development of a logical reasoning for computers programming. This is a interdisciplinary proposal of integration of Mathematics and Algorithms.*

Index Terms - LOGO, logical reasoning, programming.

Resumo - Este artigo apresenta um experimento realizado com os alunos do primeiro ano do curso de Ciência da Computação empregando a Linguagem LOGO como metodologia de apoio ao desenvolvimento do raciocínio lógico para programação em uma proposta de integração de conteúdos das disciplinas de Matemática e Algoritmos na forma de um projeto de contexto interdisciplinar.

Palavras-Chave: LOGO, raciocínio lógico, programação.

1. Introdução

O problema da compreensão e aplicação da lógica é muito comum nas disciplinas de programação e bem conhecido nas graduações dos cursos de informática e computação [02][06]. Não raro essa grande dificuldade apresentada pelos alunos é uma das maiores causas das reprovações e desistências no decorrer do curso, seja em instituições de ensino públicas ou privadas. É uma dificuldade que se apresenta já nas primeiras atividades das disciplinas de Matemática, Álgebra e, principalmente, Algoritmos.

Quando não solucionada a tempo traz prejuízos imensos aos discentes e docentes da instituição no cumprimento dos programas disciplinares do curso, tornando morosa a formação acadêmica do aluno e muitas vezes incompatível com o nível de atividades práticas e intelectuais que lhe serão exigidas ao longo da graduação.

Essa deficiência tem sido verificada pelos professores nas cinco turmas que ingressaram no curso de graduação em Ciência da Computação das Faculdades Integradas de

Rondonópolis – FAIR. Assim a instituição optou por oferecer aos ingressos do último concurso vestibular de 2002 um curso de extensão de Matemática Básica [13]. Baseado em projetos que auxiliem a minimizar deficiências de formação [04][07] a exemplo do curso de extensão, a linguagem de programação LOGO foi oferecida, na forma de um experimento com atividades em laboratório sob acompanhamento dos professores de Matemática e Algoritmos, como mais uma alternativa para buscar solucionar os problemas e as deficiências no desenvolvimento e na aplicação da lógica para programação.

A intenção do trabalho foi realizar um experimento no laboratório de informática, observando a reação dos alunos do primeiro ano do curso utilizando a linguagem de programação LOGO, que possui características pedagógicas reconhecidas, como ferramenta no desenvolvimento da lógica matemática e da sistematização do raciocínio na abstração e resolução de problemas de pequena complexidade. O trabalho teve ainda como objetivo demonstrar a possibilidade de empreender um projeto interdisciplinar a partir do uso da linguagem LOGO. Foi utilizado o ambiente de programação SuperLOGO 3.0 por ser uma distribuição livre da linguagem e pelos recursos do ambiente que favorecem sua utilização em projetos de outras disciplinas como Estruturas de Dados e Inteligência Artificial.

Como os alunos em questão já estavam fazendo uma recuperação do programa de matemática do ensino médio, foram utilizados alguns conteúdos de geometria e uma revisão em paralelo dos conteúdos já abordados pela disciplina regular de Algoritmos da graduação.

A análise dos resultados obtidos com esse experimento em conjunto com os professores das referidas disciplinas, estes foram considerados satisfatórios, possibilitando a implementação do SuperLOGO 3.0 como metodologia auxiliar para promover atividades de interdisciplinariedade, a priori, entre os conteúdos das disciplinas de Algoritmos e

¹ Scheila Wesley Martins, Faculdades Integradas Cândido Rondon – UNIRONDON, Av. Beira Rio, 3001, 78.065-780, Cuiabá, MT, Brasil, swrs@terra.com.br, aluna do curso de Especialização em Informática em Educação - FAEPE/UFLA.

² Luiz Henrique de Andrade Correia, Universidade Federal de Lavras – UFLA, Campus Universitário Cx. Postal 37, 37200-000, Lavras, MG, Brasil, lcorreia@ufla.br, orientador e professor da disciplina Computador Tutelado do curso de Especialização em Informática em Educação – FAEPE/UFLATEC.

Matemática na graduação do curso de Ciência da Computação.

Entre os aspectos da avaliação da ferramenta SuperLOGO3.0, que a metodologia proposta esperava observar, podemos destacar as seguintes:

- Verificação de evidências na melhora do desenvolvimento e do aprimoramento do raciocínio lógico e matemático dos alunos;
- Possibilitar uma dinamização do contexto da recuperação dos conteúdos de Matemática do ensino médio e, de Algoritmos, como revisão e fixação de conceitos introdutórios;
- Verificação de evidências na melhora do desenvolvimento da capacidade de abstração de problemas e na sistematização do raciocínio das soluções desses problemas;
- Possibilitar uma introdução menos traumática do aluno iniciante e inexperiente a um ambiente de programação;
- Verificação de uma melhora na recuperação e nivelamento dos alunos com deficiências nas disciplinas de Algoritmos e Matemáticas na graduação em Ciência da Computação;

Durante as seções que seguem serão descritas com maiores detalhes o modelo aplicado no experimento, as características que levaram a escolha da linguagem LOGO para composição da metodologia, bem como os resultados observados e as considerações finais sobre o experimento realizado.

2. Metodologia

Durante os meses de abril e maio de 2002, foram feitas reuniões para delinear as formas de aplicar a ferramenta SuperLOGO 3.0 a partir da avaliação de estudos similares como os descritos em [02] adequando a metodologia e o modelo do projeto executado à realidade da instituição, de maneira a maximizar os resultados da recuperação dos alunos nos conteúdos do curso de extensão de Matemática Básica, dinamizando e otimizando essa recuperação pela revisão em paralelo dos conteúdos introdutórios de Algoritmos e de Matemática e promover uma atividade de caráter interdisciplinar entre os conteúdos abordados.

Foi tomado o cuidado de esclarecer que, embora todos os docentes estivessem empolgados com as perspectivas positivas do experimento, era preciso reduzir um pouco essas expectativas devido a diferenças eminentes nos projetos, levando em consideração a qualidade do público alvo, os objetivos propostos e a forma de desenvolvimento das atividades.

Durante esse período foram envidadas atividades individuais de análise da Linguagem LOGO e de experimentação da ferramenta SuperLOGO3.0 pelos docentes [01][03] e realizada a confecção de listas de exercícios, bem como a preparação das instalações do

laboratório de informática e a formalização de um convite aos alunos para que participassem do experimento.

As listas de exercícios preparadas não continham apenas os exercícios, também traziam especificados os objetivos e os conhecimentos a serem gerados, numa forma de indicar ao aluno que o que estava sendo realizado, embora a primeira vista pudesse parecer trivial, não era uma brincadeira sem sentido. Haviam vários conhecimentos implícitos como elaboração de fórmulas, uso de variáveis e contantes e uso de estruturas de controle a cada atividade apresentada.

Dessa forma foram elaboradas três listas de exercícios a serem aplicadas, cujos os conteúdos abordavam respectivamente: reforço da capacidade de abstração para a confecção de algoritmos, movimentação pelo plano cartesiano, rotinas para construção e manipulação de polígonos básicos e noções de geometria; uso de estruturas de controle e escrita de procedimentos.

Uma das exigências da instituição para viabilizar a execução do experimento era justamente que não houvesse a necessidade de aquisição de nenhum equipamento ou programa que implicasse em custo adicional, procurando fazer uso de material já disponível. Assim sendo, a ferramenta utilizada e que implementaria a linguagem LOGO em um ambiente de programação bastante acessível do ponto de vista didático-pedagógico, uma vez que integra sabiamente módulos de desenvolvimento dos canais visual, auditivo e textual foi o SuperLOGO3.0, ambiente traduzido e disponibilizado pelo Núcleo de Informática Aplicada a Educação – NIED/UNICAMP, que tem como característica ser um software de distribuição livre.

Em uma primeira análise foi possível lançar mão de uma ferramenta como a linguagem de programação LOGO para diminuir as dificuldades encontradas por professores e, principalmente, por alunos em relação a lógica, tal como foi realizado em [02], porque ela trás muitas vantagens. Para os docentes permite uma maior dinamização no contexto dos conteúdos a serem abordados, facilitando o processo de nivelamento e recuperação dos alunos com deficiência nas áreas de matemática, lógica e algoritmos em função da simplicidade com a qual exercícios podem ser implementados nessa ferramenta e, por viabilizar a aplicação de um modelo interdisciplinar e/ou multidisciplinar da construção do conhecimento. O docente teria uma forma de estruturar com maior eficiência medidas adicionais ou corretivas de reger o processo de recuperação do aluno [10].

Por outro lado, o LOGO proporcionaria ao discente uma metodologia que acrescentaria a este medidas alternativas de aprendizagem. Pelas características próprias de trabalho da linguagem, o aluno poderia evoluir mais efetivamente no processo de construção do seu conhecimento, em seu próprio ritmo, desenvolvendo e aperfeiçoando sua capacidade de abstração pela verificação do erro sob uma outra perspectiva [11].

É nesse sentido que o aluno com dificuldades no aprendizado das disciplinas pode sair ganhando em tomar

contato primeiramente com uma linguagem como o LOGO que notoriamente favorece a construção e o exercício do raciocínio lógico. Suas características permitem ainda que o aluno vá além e se exercite na elaboração de rotinas de programação que trazem também um contato prévio com um ambiente de programação. Porém nesse ambiente a ocorrência do erro é menos traumática e mais saudável por possibilitar a promoção de conhecimentos gradualmente. O aluno passa a ser um agente mais ativo na recuperação de suas deficiências e falhas.

Na próxima seção serão apresentados detalhes sobre do experimento.

3. O Experimento

O ambiente SuperLOGO 3.0 foi apresentado aos alunos paulatinamente, ao longo de 08 horas de atividades, divididas em 05 aulas de aproximadamente 01h 30min de duração. As atividades no laboratório se realizaram duas vezes por semana em horário anterior ao início das aulas regulares, a partir das 17h 30 min indo até as 18h 50min.

As listas de exercícios abrangendo considerações sobre comandos do ambiente e os conteúdos de geometria associados aos conceitos básicos da construção de algoritmos foram reconvidadas pelos discentes a cada aula e, ao final do curso, um pequeno projeto individual foi implementado e entregue para avaliação.

A resolução das listas de exercícios foi efetuada sob acompanhamento dos professores das duas disciplinas que observaram a reação dos alunos durante essas atividades e durante as atividades em sala de aula das disciplinas de Matemática e Algoritmos. Foram elementos muito observados pelos professores: comportamento dos alunos com relação a fixação dos conteúdos apresentados no experimento e os apresentados em sala de aula, as dificuldades apresentadas no ambiente SuperLOGO 3.0 e nos exercícios das listas, bem como o desempenho por eles demonstrado nas disciplinas da graduação.

Ao final de cada atividade no laboratório era realizada uma verificação informal junto aos alunos questionando-os sobre as atividades desenvolvidas no dia e uma identificação das perspectivas destes em relação ao curso como um todo.

Todos esses elementos eram alvos de análise e discussão entre os professores ao final de cada atividade no laboratório e ao final do experimento, a razão de verificar a eficácia da ferramenta nos objetivos específicos propostos.

Em função de não se tratar de um curso de extensão nos moldes definidos pela instituição e, dada a capacidade das instalações utilizadas não suportarem o total de alunos do primeiro ano, pois comportavam apenas vinte máquinas, e estes foram convidados pelos professores a frequentar o curso. A intenção era ter um aluno por máquina para melhor observar suas reações. Todos os participantes estavam cientes de que a presença não era obrigatória e que a participação no experimento não implicava na distribuição de pontos de bonificação para qualquer das disciplinas de Algoritmos ou Matemática.

A seguir são apresentados os resultados obtidos com a realização do experimento.

4. Resultados

As atividades começaram a ser realizadas com um número de participantes bastante razoável nas primeiras aulas. Na segunda aula, por exemplo, o número de participantes foi bastante elevado, sendo necessário a acomodação de dois alunos por máquina, o que foi constatado como em [02] não muito produtivo, uma vez que a operação da máquina ficava delegada, quase sempre, a apenas um dos alunos da dupla. Ocasionalmente, alunos em dupla se envolviam em tal nível com a atividade que dois comportamentos eram notados: ou eles se integravam na busca pela solução do exercício, ou acabavam discordando a tal ponto que torna-se necessário a separação da dupla e a realocação dos alunos em nova composição de parceria.

A partir da terceira aula, um problema de ordem médica obrigou um dos professores ao afastamento por aproximadamente uma semana, provocando a suspensão das atividades por esse período. Ao retorno, na semana seguinte, notou-se uma redução expressiva no número de participantes.

Observou-se também que o horário escolhido para a realização das atividades, antes do início das aulas, acabou por tornar-se inconveniente. Foi constatada uma sobrecarga de atividades nos alunos devido ao término das aulas do curso de extensão de Matemática Básica e, também, devido a proximidade da semana de provas bimestrais.

Ficou constatado que poucos alunos do total esperado estavam participando e, destes, apenas uma parcela reduzidíssima dos que iniciaram as atividades chegaram efetivamente ao final do experimento. Além do percentual de participação dos alunos ter sido bastante reduzido, também pudemos observar que os alunos não exploraram todo o seu potencial criativo durante as atividades, limitando-se a resolução dos exercícios e prendendo-se aos temas já apresentados.

Apesar da presença não ser obrigatória, de qualquer forma, todas as aulas tiveram listas de frequência para acompanhamento dos níveis de participação e evasão como especificados na tabela abaixo:

TABELA I
Índice de Participação dos Alunos.

	N ° DE PRESENTES	(%)*
Aula 1	15	23
Aula 2	21	37
Aula 3	08	12,7
Aula 4	08	12,7
Aula 5	05	7,7

O fato das atividades não oferecerem algum tipo de bonificação e, devido ao reduzido número de alunos que

* Calculado em relação ao total de alunos matriculados no primeiro ano.

entregaram de fato os trabalhos, a avaliação destes foi realizada de maneira conceitual, levando em consideração a criatividade e a perícia do aluno em solucionar o problema a partir do tema proposto para o projeto, pois não se tratava de um curso formal do currículo, e isso pode ter também contribuído para a falta de participação dos alunos e para que a evasão ocorresse.

Em termos de programação LOGO, os resultados corresponderam razoavelmente ao nível que se esperava na metodologia. O desenvolvimento das atividades demonstraram uma reação positiva quanto a metodologia e a ferramenta SuperLOGO3.0, porém existe uma necessidade maior de tempo e um número maior de atividades para que os alunos cheguem a um nível onde os recursos do ambiente tais como a técnicas de recursão e o trabalho com vetores, listas e arquivos possam ser explorados com maior abrangência. Contudo foi possível perceber entre os alunos que realizaram o experimento até o final, mudanças comportamentais interessantes.

Em primeiro lugar, uma das mudanças mais evidentes foi a apresentação de uma reação mais pró-ativa dos alunos em relação as atividades sugeridas pelas disciplinas regulares do curso. De um modo geral, foi notada uma postura menos reativa às exigências decorrentes dos problemas aplicados em sala de aula, pois estes passaram de problemas “impossíveis de resolver” para problemas “difíceis de resolver” na perspectiva desses alunos. No âmbito específico da disciplina de Algoritmos, percebeu-se uma melhora na atitude desses alunos que passaram a ser mais dedicados e menos propensos a desistência, mesmo que o resultado em termos de nota não tenha correspondido ao esperado pelo aluno.

Em segundo lugar, observou-se que esses alunos começam a encarar o trabalho com o computador de maneira diferente. Estavam mais conscientes de que a atividade de programar necessita da compreensão da natureza e especificações dos problemas, para poderem então solucioná-los, e da habilidade de raciocinar e aplicar a lógica, isto é, a solução computacional começa no programador e não na máquina.

Embora persista a atitude, muito comum nos alunos, de associar à máquina qualidades humanas, onde ela deveria apresentar uma postura mais “colaborativa” no sentido de efetuar as correções necessárias para que os programas e comandos se executem de maneira correta, percebeu-se que esse atitude vem sendo substituída por uma maior racionalidade do aluno com relação as suas atribuições enquanto programador.

Por fim, notou-se que as dificuldades desses alunos com relação a compreensão e emprego dos conceitos básicos da construção de algoritmos tais como a utilização de variáveis, constantes, estruturas de repetição, o trabalho com ângulos, implementação e desenvolvimento de funções matemáticas e a visão espacial foram bastante minimizadas.

Ao final dos trabalhos, devido aos imprevistos decorrentes por nos encontrarmos dentro do período de

provas bimestrais, foi realizada uma avaliação verbal com questionamentos sobre as expectativas e impressões dos alunos e pudemos observar uma empatia tanto com relação a ferramenta SuperLOGO 3.0 quanto com a metodologia aplicada.

5. Conclusão

As razões que levam a presença dessa dificuldade na aplicação da lógica na programação nos cursos de graduação de informática podem remeter a alguns fatores críticos pelos quais passam o ensino nacional. Um desses fatores está fundamentado nas deficiências de formação observadas no ensino médio e fundamental, trazida a tona a partir da necessidade dos governos de instaurar e participar de exames de avaliação das qualidades de ensino de âmbito nacional e internacional. Dois exemplos claros dessa prática é a realização do ENEM e do PISA.

O ENEM (Exame Nacional do Ensino Médio) [13], instituído pelo Ministério da Educação e realizado anualmente a partir de 1998, com adesão facultativa dos alunos que estão concluindo ou já concluíram o ensino médio no país, com o objetivo de verificar o desempenho do aluno ao término da escolaridade básica (ensino fundamental e médio), visando aferir o desenvolvimento das competências e habilidades necessárias ao exercício pleno da cidadania.

O PISA (Programa Internacional de Avaliação de Alunos)[12], desenvolvido e coordenado internacionalmente pela Organização para Cooperação e Desenvolvimento Econômico (OCDE), que é realizado de três em três anos a partir de 2000 com a adesão do Brasil e de outros 35 países é um programa de avaliação comparada e tem por finalidade avaliar o desempenho de alunos a partir de 15 anos de idade, produzindo indicadores sobre a efetividade dos sistemas educacionais dos países participantes.

A análise dos últimos resultados observados nessas provas de qualificação bem como os resultados dos próprios vestibulares de várias universidades do país sugerem que o ensino de segundo grau, não só mas principalmente no Brasil, não vem atingindo com efeito o seu objetivo na formação intelectual do jovem, demonstrando um número cada vez maior de estudantes ingressando, ou tentando ingressar, no ensino universitário despreparados, sem os conhecimentos básicos necessários e bem fundamentados, além de competências fundamentais pouco ou pobremente desenvolvidas. E não estamos falando apenas das deficiências de conhecimento de conteúdos e da aplicação do raciocínio lógico, incluímos também nessa lista aquelas deficiências que decorrem da dificuldade na concatenação de idéias e da construção do pensamento crítico.

Outro fator que temos observado contribuir para o número de desistentes dos primeiros anos de graduação tem haver com o desconhecimento, por parte dos vestibulandos, das características e do perfil de aluno que os cursos da área de computação e informática exigem. Muitas vezes o estudante não tem a mínima idéia do que vai encontrar na

graduação e, não é de admirar que ao se deparar com as perspectivas do curso opte por deixá-lo, especialmente se sua expectativa quanto ao currículo não condiz com o seu perfil.

Também não podemos deixar de indicar como fator importante, que deva ser considerado, o formato dos processos de seleção executados bem como a qualidade da clientela atendida pelas instituições que oferecem cursos de graduação em informática e computação.

Todos esses fatores tem particularidades que, em conjunto, contribuem para o cenário de deficiências apresentado pelas graduações em informática e, em separado, são passíveis de produzirem discussões mais abrangentes sobre suas causas e efeitos.

Contudo, o problema do raciocínio lógico, pelo que se tem observado, não é um problema isolado desse ou daquele tipo de instituição e, esforços tem sido despendidos em estratégias de nivelamento e recuperação na grande maioria dos cursos de computação do país, talvez devido a não ser comum nas graduações a inclusão, em seus currículos, de disciplinas introdutórias que focalizem formas de melhor explorar os métodos de solução de problemas. Essa é uma habilidade extremamente necessária a formação e ao bom desempenho do discente durante o curso.

Estudos recentes indicam que os índices de reprovação e desistências nos primeiros anos dos cursos de graduação em informática e computação têm sido muito elevados, motivando uma série de preocupações e discussões entre professores e coordenadores [06]. A conclusão a que se tem chegado para justificar esses índices é simples: as características de um aluno de graduação em informática passam, indubitavelmente, pela capacidade deste de abstrair, organizar e aplicar a lógica.

Programar, aprender a programar e, principalmente, ensinar a programar não são atividades fáceis de realizar. Para uma parcela pequena da população de discentes dos cursos de computação e informática programar, ou melhor, solucionar problemas utilizando linguagens de programação com consciência científica e metodológica do que é programar[08], seja talvez uma habilidade natural mas para a grande maioria essa consciência da natureza dessa atividade e da habilidade de abstração no uso de uma ferramenta de programação não está clara. É preciso esforço e paciência, do docente e do discente, envolvidos no processo da análise e compreensão da natureza dos problemas, no levantamento de seus requisitos, na elaboração e refinamento das idéias que possibilitarão chegar a solução e, principalmente, é preciso obedecer o ritmo de cada um no desenvolvimento dessas competências. Quando todo esse processo não faz parte da natureza ou não está totalmente claro para o indivíduo é preciso construir essa perspectiva dentro dele. Talvez esse seja o maior problema relacionado ao ensino de programação.

Embora, erroneamente, o LOGO seja associado por alunos e professores ao ensino infantil e fundamental, alguns trabalhos [02][09] provam sua viabilidade e aplicabilidade

no ensino superior. Por ser uma ferramenta de características reconhecidas no desenvolvimento de múltiplas habilidades de conhecimento, suas várias versões cobrem do ensino de crianças até o de adultos, e tem sido empregado com sucesso em projetos que vão da alfabetização até o aprendizado de noções de física e matemática, sendo uma ótima alternativa ao propósito da lógica de programação.

É sabido que muitas vezes o aluno que apresenta deficiência de conteúdo se sente inferiorizado em não se adequando ao ritmo e evolução dos conteúdos em sala de aula ministrados, utilizando o modelo tradicional de ensino de programação a partir do paradigma imperativo. Como na graduação em computação, toda a evolução do aluno dentro do curso vai ser baseada na qualidade com que seus conhecimentos na disciplina de Algoritmos sejam bem absorvidos e, como esta disciplina requisita em quase sua totalidade conhecimentos matemáticos fortemente esclarecidos e ampliados, não é difícil observar porque os rendimentos nas duas disciplinas, Algoritmos e Matemática, é muitas vezes sofrível. E, não raro, no caso de Algoritmos venham a ser desastrosos, pois leva ao aluno a um desânimo contundente, o que resulta em um número muito elevado de reprovações e, conseqüentemente, em desistências.

Desde a muito tempo vêm sendo apresentados a comunidade acadêmica pesquisas direcionadas a amenizar a problemática do emprego da lógica no ensino da programação. Muitas pesquisas, com especial atenção a atividade de programar, especificam novas metodologias de ensino atentando para o fato de que o ensino de programação envolve muito mais que o domínio de uma ferramenta ou linguagem.

O domínio da ferramenta de programação só poderia ser devidamente iniciado quando a capacidade de abstração e o exercício do raciocínio estiverem devidamente fundamentados no aluno. O aluno já deveria ingressar no curso com total domínio para exercer essa capacidade de abstração e exercício da lógica, como herança desenvolvida no ensino fundamental, porém os resultados das provas qualitativas, como o ENEM, e dos vestibulares comprovam que em muitos casos essa competência ou não foi desenvolvida ou o foi muito precariamente.

O surgimento de novas metodologias no ensino de programação que levam em consideração as deficiências de formação estão se popularizando e sendo reconhecidas como medidas do mais alto nível na construção do conhecimento, pela adoção de abordagens que levam em consideração estratégias e formas de promover e capacitar o aluno a solucionar problemas adequadamente. Solucionar problemas adequadamente é raciocinar adequadamente.

Muitos autores [02][05][06][08] propõe o emprego de novas metodologias e ferramentas de auxílio ao ensino da programação, incluindo em seus estudos algumas sugestões interessantes, como a aplicação do paradigma de programação declarativo pelo emprego de ferramentas da Programação Funcional, inclusive o LOGO. O modelo

tradicional de ensino de programação pelo paradigma imperativo é mantido, mas em virtude de atender a necessidade de melhorar ou desenvolver no aluno a capacidade de solucionar problemas, a programação funcional é introduzida primeiro, com o objetivo de promover um nivelamento tornando o paradigma imperativo menos traumatizante e o aluno menos propenso a desistência.

Analisando as perspectivas iniciais do experimento e comparando os resultados obtidos podemos concluir que o experimento obteve um resultado positivo em relação aos objetivos propostos pela metodologia, apesar dos problemas decorridos durante as atividades, entre os quais podemos destacar a evasão dos alunos.

Ao que parece essa evasão pode ter sido ocasionada em função do período de aplicação do experimento não ter sido o mais propício, pois os alunos se encontravam sobrecarregados pelo curso de extensão e a proximidade da semana de provas bimestrais; e, também favorecida pelo afastamento ocorrido durante o início dos trabalhos, que acabaram por quebrar o ritmo das atividades e obrigaram a alterações e adequações constantes na metodologia empregada.

Verificou-se que muitos alunos não estavam totalmente conscientes dos vários conceitos computacionais explorados durante a realização dos exercícios, deixando a desejar no que tange ao aprimoramento e recuperação de suas deficiências. Uma das grandes características da linguagem LOGO é possibilitar ao aluno estar mais ativo no processo ensino-aprendizagem imprimindo nesse processo o seu ritmo de absorção de conteúdo e transformação deste em conhecimento.

No entanto, uma grande maioria se aproximou do experimento movido mais pela curiosidade, como pode-se observar na tabela de frequência de aulas, sendo que alguns chegaram a entrar no laboratório pela primeira vez na última aula para “ver”, e não encararam a ferramenta com a devida seriedade, confirmando o pré-conceito de que ela estaria única e exclusivamente relacionada ao ensino infantil, deixando alguns, por esse motivo, de frequentar as atividades.

Para os professores envolvidos no projeto, apesar da evasão ocorrida durante o experimento, uma certa melhora no nível e atitude dos alunos foi realmente percebida, por um posicionamento mais ativo e mais consciente do quão importantes são as atividades de recuperação no aprimoramento e desenvolvimento de sua formação, e, espera-se que o experimento tenha contribuído para facilitar aos alunos a introdução de um ambiente de programação imperativo como a Linguagem Pascal.

Entretanto foi consenso entre eles de que para uma melhor análise quanto a eficácia da metodologia seria melhor que esta fosse readequada e ampliada em alguns aspectos, de modo que ela pudesse ser reaplicada em um período menos conturbado no terceiro ou quarto bimestre

senão no próximo ano. Entre as alterações discutidas foram especificadas:

- Ampliar o conteúdo das listas de exercícios para abranger além de geometria e conceitos básicos de algoritmos, explorar mais conteúdos de Matemática, de Algoritmos e recursos da ferramenta SuperLOGO 3.0;
- Ampliar da carga horária de 08 para 40 horas, configurando ao experimento características de um projeto educacional na modalidade de curso de extensão tal como o de Matemática Básica;
- Incluir nas atividades exercícios que possibilitem a correlação entre o ambiente de programação LOGO e sua implementação na linguagem Pascal;
- Especificar um índice de bonificação das atividades efetuadas durante o curso para incentivar e aumentar o índice de participação dos alunos;

Para os alunos que participaram do experimento até o final, as impressões foram bastante positivas e ultrapassaram as expectativas, pela intenção declarada destes em participarem novamente do experimento se este fosse realizado no segundo semestre.

A boa impressão indicada por alunos e professores sobre a ferramenta SuperLOGO 3.0 reforça a certeza de que o experimento chegou a resultados positivos quanto a aplicabilidade do LOGO como ferramenta para promoção de projetos de interdisciplinaridade, não apenas aplicada ao caso da lógica de programação como também na possibilidade de aplicação em conhecimentos mais avançados tratados por outras disciplinas como Linguagem de Programação e Inteligência Artificial.

De um modo geral a proposta de desenvolver atividades com caráter interdisciplinar dos conteúdos das disciplinas de Algoritmos e Matemática aplicando a ferramenta SuperLOGO 3.0 foi alcançada e, o presente trabalho trouxe resultados que contribuirão para que esse projeto seja efetivado na instituição.

6. Referências Bibliográficas

- [01] PRADO, Maria E. B. B.; ROCHA, Heloísa V. e FREIRE, Fernanda M. P. - Tartaruga, Figuras, Palavras, Listas e Procedimento: Um Primeiro Passeio pelo Logo-SuperLogo 3.0 - Memo N° 35. Publicação do Núcleo de Informática Aplicada à Educação da Unicamp/NIED: Campinas/SP – 2000.
- [02] ROCHA, Heloísa V. - O Uso da Linguagem Logo em um Curso de Introdução à Programação de Computadores para Alunos Ingressantes no Bacharelado de Ciência de Computação – Memo N° 10. Publicação do Núcleo de Informática Aplicada à Educação da Unicamp/NIED: Campinas/SP - 1988.
- [03] CORREIA, Luiz H. A. et al. – Computador Tutelado 2. ed. – Publicação do Curso de pós-graduação “Lato

Senso” (Especialização) a distância: Informática em Educação – Ed. UFLA/FAEPE: Lavras - 2001.

- [04] CASTRO, Thais H. C. de et. al. – Arquitetura SAAP: Sistema de Apoio à Aprendizagem de Programação – Trabalho apresentado no Workshop sobre Informática na Escola/WIE (Grupo Temático E: Inteligência Artificial Aplicada à Educação) – Páginas 990 a 998 - Anais do XXII Congresso da sociedade Brasileira de Computação: Florianópolis/SC – 2002.
- [05] CASTRO, Thais H. C. de et. al. – Utilizando Programação Funcional em Disciplinas Introdutórias de Computação – Trabalho apresentado no X Workshop sobre Educação em Computação/WEI (Seção Técnica II B) – Páginas 157 a 168 - Anais do XXII Congresso da sociedade Brasileira de Computação: Florianópolis/SC – 2002.
- [06] FERNANDES, Jorge H. C. – Ensino Introdutório de Computação e Programação: Uma Abordagem Concreta, Construtivista, Lingüística e Histórica – Trabalho apresentado no X Workshop sobre Educação em Computação/WEI (Seção Técnica II B) – Páginas 139 a 146 – Anais do XXII Congresso da sociedade Brasileira de Computação: Florianópolis/SC – 2002.
- [07] ALMEIDA, Eliana S. de – AMBAP: Um Ambiente de Apoio ao Aprendizado de Programação – Trabalho apresentado no X Workshop sobre Educação em Computação/WEI (Seção Técnica I B) – Páginas 79 a 88 – Anais do XXII Congresso da sociedade Brasileira de Computação: Florianópolis/SC – 2002.
- [08] RODRIGUES, Methanias Colaço R. Junior. – Como Ensinar Programação? – Artigo retirado do jornal Computação Brasil publicado pela Sociedade Brasileira de Computação/SBC – Página 05 - Ano III – Edição 07 - Setembro de 2002.
- [09] VALENTE, José A. - O Professor no Ambiente LOGO: Formação e Atuação – Publicação do Núcleo de Informática Aplicada à Educação da Unicamp/NIED: Campinas/SP - 1996.
- [10] TAJRA, Sanmya F. - Informática na Educação: Novas Ferramentas Pedagógicas Para O Professor Na Atualidade – 3º Edição Ampliada - Ed. Êrica: São Paulo – 2001
- [11] TUCCI, Wilson J. e VON PFUHL, Dulce M. A. – LOGO: Programação e Aprendizagem – (128 p.) - Ed. Nobel: São Paulo – 1986
- [12] PISA200, - Relatório Nacional PISA 2000 – Publicação Eletrônica do Instituto Nacional de Pesquisas Educacionais – INEP, publicado em <http://www.inep.gov.br/download/internacional/pisa/PI SA2000.pdf>, acessado em 16/11/02 as 17:05.
- [13] ENEM2002, - “Enem 2002: Resultados do Exame chegarão pelos Correios” – Publicação Eletrônica da seção de notícias do site Instituto Nacional de Pesquisas Educacionais – INEP http://www.inep.gov.br/imprensa/noticias/enem/news02_16, acessado em 19/11/02 as 15:37.