

# LOGO NO ENSINO SUPERIOR DE PROGRAMAÇÃO DE COMPUTADORES: MOTIVAÇÃO E *EMPOWERMENT* DISCENTE

Márcio Roberto de Lima<sup>1</sup> e Murilo Cruz Leal<sup>2</sup>

## **Abstract**

*This article reports an experiment conducted with 21 senior students of a course in Information Systems, with which it sought to verify the validity of the Logo programming language as a tool facilitating the learning program in higher education. The work involved a course of 40 hours-to-face, which looked a (re) vision of programming concepts from the perspective of the LOGO language. At the end of working hours available five students agreed to participate in a semi-structured. The interviews were recorded, transcribed and subjected to content analysis. The fragments of interest were identified and categorized based on units of meaning related to the focus of research. In addition to the testimonies gathered, the results show the LOGO as a tool for easy and proper ownership for the ongoing work of introduction to computer programming in connection with other languages.*

**Keywords:** LOGO, programming, constructionism

## **Resumo**

Este artigo relata um experimento realizado com 21 alunos formandos de um curso de Sistemas de Informação, com os quais se buscou verificar a pertinência da linguagem de programação LOGO enquanto ferramenta facilitadora da aprendizagem de programação no Ensino Superior. O trabalho envolveu uma unidade curricular de 40 horas, em regime presencial, a qual contemplou uma (re)visão de conceitos de programação sob a perspectiva da linguagem LOGO. Ao final da carga horária disponível cinco alunos concordaram em participar de uma entrevista semi-estruturada. As entrevistas foram gravadas, transcritas e submetidas à análise de conteúdo. Os fragmentos de interesse foram identificados e categorizados com base nas unidades de significado relacionadas ao foco de pesquisa. Em adição aos depoimentos colhidos, o resultado da análise apontou a LOGO como uma ferramenta de fácil apropriação e adequada para ser trabalhada em cursos de introdução à programação de computadores no Ensino Superior, buscando sua conexão com outras linguagens.

**Palavras-chave:** LOGO, programação, construcionismo

---

<sup>1</sup> Mestre em Educação, Universidade Federal de São João del-Rei (UFSJ), 2009. Professor da Fundação Presidente Antônio Carlos (FUPAC). E-mail: marcinho@marcinholima.com.br

<sup>2</sup> Doutor em Educação, UFMG, 2003 – Membro do Departamento de Ciências Naturais (DCNAT/UFSJ) – Professor do Mestrado em Educação da Universidade Federal de São João del-Rei (UFSJ). E-mail: mcleal@ufs.edu.br.

## INTRODUÇÃO

O cotidiano da sociedade contemporânea reflete permanentemente o rápido desenvolvimento tecnológico em curso. A presença das Tecnologias de Informação e Comunicação (TIC) proporciona, a cada novo ciclo de seu desenvolvimento, quadros outrora inimagináveis, mas que na modernidade se avolumam e provocam mudanças em todas as áreas de conhecimento. Neste cenário de transformações, em que se instaura a cibercultura, tornou-se imprescindível a todos os seguimentos da sociedade, o acompanhamento racional da evolução das TIC, bem como a (re)adequação de seus processos.

É nessa perspectiva de mudança e inovação, que surgiu o uso da “Informática na Educação”, ou seja: a inserção do computador no processo de ensino-aprendizagem de todos níveis e modalidades educacionais. Assim, uma melhor compreensão do uso do recurso nos processos educacionais fez-se necessária, buscando identificar: a validade do computador enquanto ferramenta de apoio à aprendizagem e as implicações de seu uso por parte de professores e alunos.

Neste sentido, a adoção das TIC por todos os segmentos sociais suscitou a criação de cursos – a nível técnico, superior e mesmo livre – para preparar os profissionais que exercem atividades específicas, tais como: programadores de computadores, técnicos em hardware, analistas de sistemas, administradores de banco de dados, engenheiros de *software* e tecnólogos em geral. Especificamente, no Ensino Superior, destaca-se o surgimento dos cursos de Ciência da Computação, Engenharia da Computação, Licenciatura em Computação, Sistemas de Informação, Tecnologia de Redes e Banco de Dados, entre outros. Entretanto, além do domínio específico da área da computação, os processos educacionais do Ensino Superior das demais áreas de conhecimento vêm se adequando e, gradativamente, incorporando unidades curriculares relacionadas às TIC em suas estruturas curriculares.

Todas essas mudanças ainda estão em curso e geram um fértil ambiente de pesquisas. Alguns indicativos apontam caminhos para a produção de conhecimentos nesse campo; por exemplo, Perrenoud (2000), propõe um deslocamento do foco de discussão da utilização ou não das TIC para a forma de sua utilização. Em comum acordo, Lévy (1999, p.26) afirma que “enquanto discutimos sobre os possíveis usos de uma dada tecnologia, algumas formas de usar já se impuseram”. Quartiero (2007), ao discutir os trabalhos de Salomon (1992; 1998; 2001), amplia essa corrente de pensamento ao destacar que “[...] a principal preocupação que se deve ter quando se introduz uma nova tecnologia em sala de aula é com relação à qualidade

da aprendizagem resultante do uso dessa tecnologia” (Quartiero, *op. cit.*, p.58). Assim, a relevância dos aspectos relacionados à qualidade da aprendizagem com as TIC torna-se ainda mais imperativa em ambientes educacionais onde o computador é um elemento indispensável, como é o caso dos cursos específicos da área da computação.

Nesse sentido, este trabalho teve como ambiente de investigação um curso de Sistemas de Informação de uma universidade do interior de Minas Gerais e, especificamente, concentrou-se no núcleo de unidades curriculares (UCs) relacionados à programação de computadores. Conforme apresentado por Martins e Correia (2003), Rodrigues Júnior (2002), Schultz (2003), Chaves de Castro *et al* (2003), Delgado *et al* (2004), Maltempi e Valente (2000) e Petry (2005), os resultados obtidos nas UCs que envolvem programação de computadores são insatisfatórios, constatando-se: baixo nível de aprendizagem, desestímulo, evasão e reprovação. O estudo empreendido orientou-se pela perspectiva construcionista e pelo potencial pedagógico-computacional da linguagem LOGO, desenvolvida por Seymour Papert, tendo como objetivo específico compreender se (e como) esta linguagem é facilitadora da aprendizagem de programação em cursos do Ensino Superior. Para isso, contou-se com a colaboração de 21 alunos formandos de um curso de Sistemas de Informação, com os quais se desenvolveu uma UC em regime presencial visando promover uma revisão de conceitos de programação sob a perspectiva da linguagem LOGO. O aporte teórico do curso contemplou os conceitos gerais de programação sob a perspectiva do LOGO, sempre os relacionando às linguagens academicamente exploradas (PASCAL, JAVA etc), além de tópicos referentes ao uso de computadores na Educação. O curso teve duração de 40 horas e foi realizado no período de 07 de Agosto a 05 de Dezembro de 2008.

## BREVE APRESENTAÇÃO DO CONSTRUCIONISMO

A década de 60 do século passado marcou, ainda que de forma discreta, o início da introdução dos computadores nos sistemas educacionais. O objetivo inicial de seu uso foi o de sofisticar a instrução programada, o que ficou conhecido como *Computer Aided Instruction* (CAI). Com o advento dos microcomputadores na década de 80, o *software* CAI ganhou força, o que ampliou o processo de inserção dos computadores nas escolas, principalmente nos países desenvolvidos. Diversificaram-se, então, os tipos de *softwares* disponíveis. Além dos tutoriais, surgiram os programas de demonstração, exercício/prática, jogos didáticos e simuladores (Valente, 1993).

Assim, em meio a esse panorama de inovação tecnológica, os sistemas educacionais passaram a conviver com os computadores. Dentre os céticos que acompanharam esse evoluir, a figura de Seymour Papert destacou-se no questionamento de que qual seria a melhor via de integração dos computadores na educação. Papert é Sul-Africano e tem formação em matemática. Dedicou-se a pesquisas na área de matemática na *Cambridge University* no período de 1954 a 1958. Posteriormente, transferiu-se para a Universidade de Genebra onde trabalhou de 1958 a 1963. No início da década de 60 filiou-se ao *Massachusetts Institute of Technology* (MIT).

Valente (1993, p.12) explica que na visão de Papert o computador não deveria ser um “instrumento que ensina o aprendiz, mas a ferramenta com a qual o aluno desenvolve algo, e, portanto, a aprendizagem ocorre pelo fato de estar executando uma tarefa por meio do computador”. Fica explícita a idéia de que com o “computador ferramenta” o aluno será o sujeito promotor de uma ação, ou seja: seu lugar deixa de ser o de espectador (ou recebedor de conteúdos) e passa a ser o de agente. Neste sentido, Papert (1986) sugeriu o termo construcionismo para designar a modalidade em que um aluno utiliza o computador como uma ferramenta com a qual ele constrói um produto do conhecimento, por exemplo, um programa de computador.

Percebe-se, portanto, que ao associar o uso dos computadores à Educação Papert propôs uma abordagem alternativa à que estava sendo, até então, implantada. Em sua concepção os computadores podiam e deviam ser utilizados “como instrumentos para trabalhar e pensar, como meios de realizar projetos, como fonte de conceitos para pensar novas idéias” (Papert, 2008, p.158) e não apenas como uma forma sofisticada de apoio à instrução automatizada. Nesse sentido, Papert e sua equipe do MIT, entre os anos de 1967 e 1968, passaram a desenvolver uma forma de uso do computador que viabilizasse tais idéias: a ferramenta educacional LOGO.

Do ponto de vista computacional, a Linguagem de Programação LOGO foi concebida para ser inteligível e, dessa forma, de fácil assimilação para iniciantes. O ambiente de programação LOGO tipicamente disponibiliza a Tartaruga<sup>3</sup>, uma criatura robótica, que é direcionada por meio de comandos. Desde sua introdução, a linguagem LOGO foi implementada em diferentes *softwares*, cada qual com sua particularidade. Atualmente, esses

---

<sup>3</sup> A Tartaruga pode ser um objeto virtual que se desloca nos monitores de vídeo dos computadores, ou, como na sua origem, um objeto físico como os robôs que se deslocam no chão.

ambientes possuem uma “Janela de Comandos”, que é uma área de interação para o usuário usada para a escrita de comandos usando a Linguagem, bem como a “Janela Gráfica” – o *habitat* virtual da Tartaruga – nela os comandos propostos podem ser executados e seus efeitos observados (figura 1).

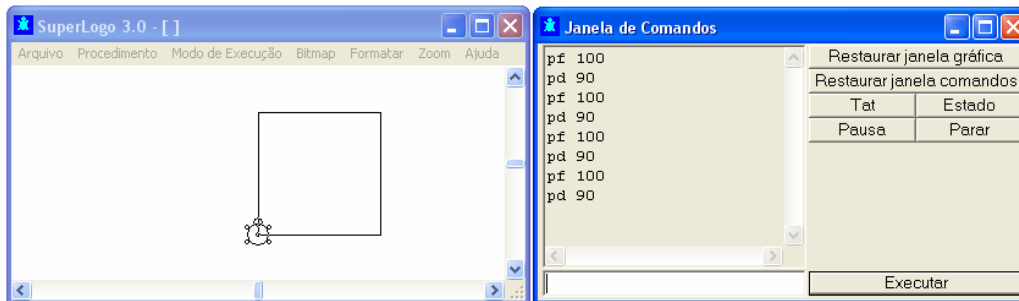


Figura 1 – Ferramenta SuperLogo: Janela Gráfica e Janela de Comandos.

A construção de figuras usando os comandos básicos da linguagem LOGO é uma atividade que permite a exploração de suas propriedades geométricas. No “passo a passo” dessa modalidade, o usuário tem a oportunidade de escrever um comando, a Tartaruga executá-lo, com o rastro deixado por ela o usuário recebe o *feedback* e passa a refletir sobre os comandos utilizados. Daí, se tudo correr conforme o pretendido, o aluno continua com sua descrição; na eminência de um equívoco, ele terá a oportunidade de voltar atrás e rever e depurar seus conceitos, apropriando-os. Em um primeiro momento, a idéia de programação está na expressão de um raciocínio lógico que, quando executado pela Tartaruga, revela uma figura geométrica.

As UCs que envolvem a programação de computadores compreendem inúmeros conceitos fundamentais, os quais podem ser explorados em diferentes linguagens. Entre esses, destacam-se: variáveis, estruturas de repetição, estruturas condicionais, recursão, passagem de parâmetros, entre outros. Usualmente, no nível universitário, o que insere o uso de uma Linguagem de Programação de Computadores (LPC) em um processo de ensino-aprendizagem é exatamente o suporte à implementação desses conceitos.

Dessa maneira, uma LPC deve servir como meio de expressão de idéias, sendo para isso utilizadas as regras sintáticas e semânticas dos termos da linguagem. A depender da linguagem em uso, a compreensão da forma de descrição de um problema em seu vocabulário pode gerar dificuldade, uma vez que os termos da linguagem são geralmente provenientes da língua inglesa, bem com as mensagens de erro. Além disso, a complexidade (cultura de *hards*) do modelo de representação de idéias em uma linguagem pode, também, ser um fator

complicador, principalmente para iniciantes. Em contrapartida, se uma linguagem de programação oferecer suporte a esses conceitos e puder ser aplicada, sobre uma “ótica *softer*” em um processo de ensino-aprendizagem, as dificuldades relacionadas a essa dinâmica de aprendizagem podem ser minoradas. Esse é o caso do LOGO.

Os conceitos introdutórios explorados nas linguagens PASCAL, JAVA, C, C++, PROLOG, entre outras podem, igualmente, serem implementados com LOGO. O que muda é o vocabulário, o ambiente de programação e a forma subsequente de representação das idéias em cada linguagem. Dessa maneira, do ponto de vista educacional, o LOGO não fica devendo em nada para as outras linguagens, uma vez que possui recursos avançados para a exploração dos conceitos relacionados à programação de computadores. Pelo contrário, o seu aspecto gráfico aliado a sua simplicidade sintática/semântica, possibilita a rápida visualização de um determinado conceito, o que é definitivo para a compreensão e aprendizagem.

Assim LOGO não é um “brinquedo”, uma linguagem apenas para crianças. Os exemplos mais simples [...] mostram algumas maneiras em que LOGO é especial por ter sido planejada para fornecer muito facilmente e bastante cedo acesso à programação de computadores para principiantes [...]. Logo não é um brinquedo, mas uma poderosa linguagem [...]. (PAPERT, 1985, p.22)

Das palavras de Papert, colhe-se a impressão que o LOGO oferece a possibilidade da exploração computacional, em qualquer que seja o nível educacional praticado. E mais, seu uso pode ser aplicado como uma referência inicial aos conceitos de programação para os outros ambientes de programação.

Dessa forma, sintetiza-se a abordagem construcionista em seu objeto de estudo: um problema e a sua compreensão, a elaboração de uma estratégia de solução no computador, pelo aluno, mediado por um profissional da educação; e no ferramental: um computador e uma linguagem de programação usados para a construção do conhecimento.

## O EXPERIMENTO REALIZADO

Conforme estabelecido anteriormente, a experiência empreendida contou com a participação de 21 alunos formandos de um curso de Sistemas de Informação, os quais frequentaram uma UC de 40 horas onde se explorou os conceitos de programação de computadores sob a perspectiva da linguagem LOGO, buscando estabelecer conexões com

outras LPCs academicamente adotadas. Optou-se pelo SuperLogo<sup>4</sup> como ambiente de programação, haja vista sua natureza livre, regionalização e amigabilidade. Além disso, o laboratório de informática onde o curso foi desenvolvido possuía licenciamento da plataforma *Microsoft*, o que exigia um ambiente de programação compatível.

O primeiro passo para o planejamento da referida UC foi a aplicação de um questionário diagnóstico que buscava determinar o perfil dos educandos. Com o instrumento de coleta de dados enfocou-se, primordialmente, a relação pregressa da turma com as unidades curriculares de programação de computadores. A síntese do perfil dos alunos envolvidos na UC em questão destacou um grupo que, ao longo de sua formação acadêmica, não aproveitou satisfatoriamente os conteúdos relacionados à programação de computadores. A relação com tais UC remontou a um sentimento de desmotivação, frustração, descontentamento e de pouca interação entre professor, aluno e ferramentas de *software*. De posse desses dados, seguiu-se o planejamento e execução da UC em regime regular. Ao final do curso, cinco alunos foram convidados para participar de uma entrevista individual, isso com o objetivo de se buscar evidências sobre o aproveitamento da UC em questão, bem como de elementos que justificassem o uso da LOGO como uma linguagem facilitadora da aprendizagem de programação de computadores no Ensino Superior. Destacam-se a seguir as percepções e concepções dos alunos, principais objetos da pesquisa empreendida.

### **LOGO: as primeiras impressões são as que ficam?**

Mediante o perfil traçado da turma, a oferta da UC estava diante de um grande desafio: trabalhar com alunos relutantes frente à programação e que, entre muitas outras matérias, estavam concluindo sua graduação. Era de se esperar certa apatia e descrença em relação ao que estava por vir nas 40 horas do curso ministrado. Durante a entrevista com os participantes, percebeu-se que algumas impressões iniciais convergiam com essa expectativa. Destaca-se que nenhum dos entrevistados havia tido um contato prévio com a linguagem LOGO, desconhecendo totalmente suas potencialidades. O discurso da Aluna<sup>5</sup> sintetiza as percepções do grupo:

---

<sup>4</sup> SuperLogo é um ambiente de programação LOGO que foi desenvolvido por George Mills e Brian Harvey, da Universidade de *Berkeley*. Foi traduzido e adaptado pela equipe do Núcleo de Informática Aplicada à Educação da Universidade Estadual de Campinas (UNICAMP).

*Aluna5: No principio, assim, a primeira e a segunda aula eu achei que fosse para criança, mas depois que eu vi que eu podia criar com ele... E criar coisas que crianças não criariam... Então, seria uma linguagem fácil para aprender programação, que um adulto maduro, conseguiria enxergar o que era realmente a programação.*

De modo geral as primeiras impressões discentes conduziram à idéia de perda de tempo, ou ainda, que aquela Linguagem nada acrescentaria ao que aqueles alunos concebiam em termos de programação. Entretanto, percebeu-se que tais concepções eram transitórias e, em tempo, assumiriam outra perspectiva. O choque proporcionado pela cultura *softer* idealizada por Papert fica evidente no posicionamento dos alunos quando tomaram contato com o LOGO. Formandos e acostumados a ferramentas concebidas para um universo completamente diferente do que lhes surgia, seus sentimentos foram balizados por uma perspectiva ligada ao universo infantil: uma tartaruga robótica que cumpria ordens não poderia estar associada a uma ferramenta de programação, outrora impenetrável, complexa e definitivamente fora de seus domínios.

### **LOGO: é possível aprender a programar!**

Com o desenvolver do curso, com a apropriação do vocabulário sintático e, principalmente, da semântica LOGO, os alunos começaram a vislumbrar uma nova realidade em programação. Mesmo sendo um grupo de alunos relutantes à programação, LOGO possibilitou uma retomada do contato com a área e, em alguns casos, até mesmo o efetivo entendimento de conceitos abstratos. O Aluno4 e a Aluna5 manifestaram o quanto estavam felizes por poderem experimentar programação:

*Aluno4: Como uma criança não sabe ler, nós éramos crianças que não sabíamos programar. [...] Para mim foi perfeito, [...] Aquela teoria inicial, que todo aluno precisa, ela [LOGO] dá suporte. Eu vou arriscar a falar 100%.*

*Aluna5: Eu até achei que fosse um programa de editoração gráfica [...] e como eu adoro editoração gráfica, ascendeu a luzinha de interesse! Só que com o decorrer do tempo eu descobri que através dos desenhos eu estava entendendo o que era programação.*

Percebeu-se, portanto, o surgimento de uma nova possibilidade com a programação para alunos adultos. Naquele momento, o LOGO não era mais uma ferramenta pueril. Uma nova fase começava para o grupo: programação era uma algo acessível! Eles poderiam se apropriar, compreender e, finalmente, construir programas de computador. Notou-se o início



de um rompimento com o repúdio à área de programação e que as perspectivas discentes se ampliaram com o conhecimento da Linguagem, o que possibilitou a retomada do processo motivacional para a construção de programas de computador.

A motivação de um aluno, em qualquer processo de aprendizagem, é fundamental para o seu sucesso. Guimarães e Boruchovitch (2004, p.143) afirmam que “um estudante motivado mostra-se ativamente envolvido no processo de aprendizagem, engajando-se e persistindo em tarefas desafiadoras, despendendo esforços, [...] buscando desenvolver novas habilidades de compreensão e de domínio”. Os autores ampliam esse argumento elucidando que alunos motivados mostram-se entusiasmados durante a execução das tarefas propostas e que, ao concluí-las, revelam-se orgulhosos acerca dos seus desempenhos. No contexto deste trabalho, o referido sentimento encontra-se vinculado à atividade de programação de computadores e diz respeito ao *empowerment* (sentimento de eficácia) manifestado – ou não – diante dos desafios de conceber e estruturar com sucesso um programa.

Em um processo educacional é comum identificar alunos que variam entre dois pólos: aqueles que se aprofundam no conhecimento em busca de um embasamento que possa lhes valer no futuro – motivação intrínseca – e os que apenas se dedicam a cumprirem o “regulamento”, ou seja, fazer provas e trabalhos visando à conclusão da unidade curricular com um resultado minimamente satisfatório – motivação extrínseca (Guimarães *et al*, 2002). Dessa forma, para um grupo que se limita à motivação extrínseca, o sentimento de apatia no processo educacional é praticamente inevitável, uma vez que seu compromisso está ligado estritamente a assegurar a “nota” para garantir sua aprovação e a conclusão de um curso. Entretanto, no cotidiano do curso de LOGO, de conceitos simples, os alunos se interessaram por situações que envolviam maior nível de complexidade, o que é tido como desejável em um processo educacional e que acaba por revelar maior grau de motivação intrínseca do grupo.

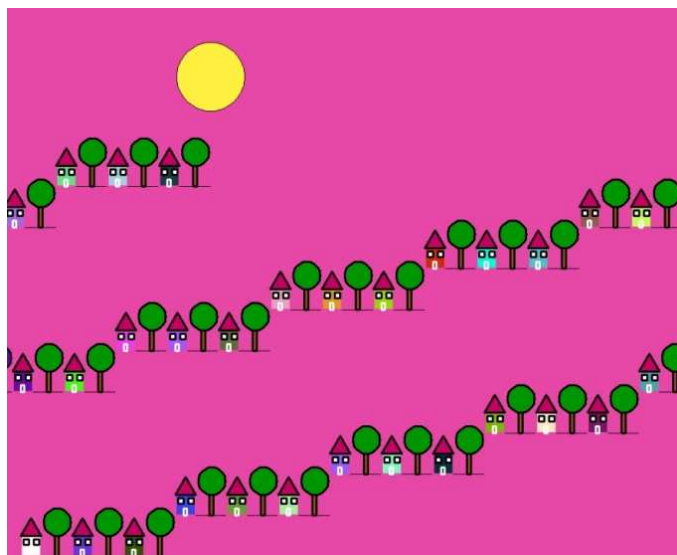


Figura 2 – Um bairro em LOGO programado pelo Aluno1

A figura 2 mostra o resultado gráfico de um trabalho do Aluno1. Por trás da aparente simplicidade do desenho, configuram-se importantes conceitos de programação de computadores, tais como estruturas de fluxo de controle de dados, parametrização de procedimentos, uso de randomização (sorteio) para cores e chamadas de procedimentos dentro de outros procedimentos (subprocedimentos). Chama-se a atenção para o fato de que essa gama de conceitos estava “resfriada” na idéias do aluno e o trabalho com o LOGO conseguiu “esquentá-la” novamente. Foi selecionada essa produção dado o efeito de *empowerment* experimentado pelo aluno, que acabou por contagiar muitos outros.

Nesse sentido, procurou-se diagnosticar entre os entrevistados, como eles situavam a utilidade da linguagem LOGO no processo de ensino-aprendizagem de programação. Destacam-se os seguintes trechos:

*Aluna2: Ela é útil. Muito útil. [taxativa]. Mas eu acho que ela seria muito mais útil, se ela fosse apresentada no começo. Para desmistificar uma coisa: que programar... “nossa se o cara é programador, o cara veio de outro planeta, veio de outro mundo, o cara cria uns negócios ali... Um botãozinho, o cara é de outro planeta”.*

Para a Aluna2, LOGO se consolidou como uma linguagem acessível e adequada à introdução à programação. O rompimento com a cultura de rígidos, que exclui os sujeitos da possibilidade de aprenderem a programar, cedeu espaço a uma nova forma de se apropriar dos conceitos de programação: a forma *softer* de construção de programas idealizada por Papert. Para aquele aluno, não era mais preciso ser possuidor de um dom especial para se ter a compreensão de programação.

**Aluno1:** *Ponto forte do LOGO é você conseguir fazer a escrita da programação e já ter o resultado no grafismo. Que, às vezes, é um resultado mais óbvio em cima do que você está querendo estruturar. Mesmo que seja uma advertência, que você não fez uma coisa certa, mas pelo menos ele já te deu o resultado que você queria. “– Ah, não é isso!”. Então eu já sei onde eu vou mudar. Ai você fez: “– Ah, é isso!”. Ai ele te apresenta o êxito na tela: “– Ah, eu consegui. Era isso que eu queria que ele fizesse!”. É uma coisa simples, mas que no fundo vai é te dar, ou acrescentar é... Uma facilidade de entendimento no que você se prestou a fazer.*

O Aluno destacou o forte apelo visual do LOGO. A sintaxe e o formalismo estrutural dos programas assumiram uma dimensão gráfica capaz de conduzir à expressão de idéias e sua depuração (melhoria do programa). Segundo ele, a fase de depuração passou a ser facilitada pelo aspecto gráfico do LOGO, que mostrava, muito claramente, onde existia um erro para a correção.

Finalmente, apresentam-se as falas dos entrevistados que apontam a viabilidade do uso do LOGO como uma ferramenta introdutória à programação de computadores. Cabe ressaltar, que essa impressão foi colhida não apenas entre os entrevistados.

**Aluno1:** *Eu acho que o LOGO tinha que ser apresentado de início [1º/2º períodos]. Não mudando, assim, a tradição de um ensino de programação. Mas, às vezes, de evoluir mesmo o ensino da programação: uma didática diferente. Porque você ter o contato com uma linguagem que te favorece na criação, na evolução, é melhor que você ter uma dificuldade ao ponto de você ter de passar por ela para conseguir entender. Porque ninguém gosta de entender uma coisa aos trancos e barrancos. [...] Às vezes poderia associar, né? É ter uma introdução com o LOGO e uma apresentação num JAVA. Exemplificando: você mostra o que é uma passagem de parâmetros no LOGO e depois você mostra uma passagem de parâmetro no JAVA.*

O Aluno1 vislumbrou a possibilidade do uso combinado do LOGO com outras linguagens desde o início de seu Curso. Compreendeu-se que o LOGO seria uma ponte entre os conceitos de programação e suas respectivas aplicações com as linguagens tradicionalmente exploradas.

Os argumentos referentes à conexão entre o LOGO e as outras linguagens, apareceram também nos discursos da Alunas 2 e 5. Além disso, essas alunas também destacaram a necessidade de que a Linguagem fosse utilizada no início de suas graduações:

**Aluna2:** *Só que é igual ao que eu estou te falando, se isso [contato com o LOGO] fosse no começo... Facilitaria muito a você a visualizar... Um comando de repetição*

*dentro de JAVA, nesse sentido. Você aprende a visualizar, você aprende a raciocinar, só de visualizar. “Ah vocês vão fazer um programinha que faz isso, isso e isso”. Você já imagina o programa, e como que você vai fazer para ele funcionar. E para a gente não ensinaram assim. Ensinaram os comandos, para depois você imaginar o programa, para depois você voltar e imaginar como é que ia fazer.*

*Aluna5: Eu acho que seria muito útil [uso do LOGO], inclusive se ele tivesse sido aplicado nos períodos iniciais, talvez a gente tivesse tido mais facilidade... Eu acho que faria ver a programação de uma forma mais clara. Eu acho que depois, poderia deixá-lo e começar só nas formas matemática [outras linguagens de programação].*

A possibilidade do uso do LOGO no início dos cursos de introdução à programação de computadores estaria relacionada a uma forma de se propiciar uma adaptação mais suave ao seu aprendizado. Compreendeu-se que uma visão *softer* da programação pode cooperar significativamente para a sua aprendizagem, ampliando as possibilidades de aproveitamento nas UC relacionadas, despertando nos aprendizes o tão cobiçado *empowerment* e preparando-os, de uma forma didatizada, para a compreensão dos conceitos envolvidos na atividade.

Estabelecer as conexões entre o LOGO e as outras linguagens é uma forma que ficou entendida como viabilizadora do processo de aprendizagem de programação. As “áreas mais frias” – aquelas essencialmente abstratas dentro da programação – poderiam “ser esquentadas” – tornando-se assim, mais concretas – mesclando-se o uso do LOGO às demais linguagens.

## CONSIDERAÇÕES FINAIS

A abordagem construcionista proposta por Papert pode ser encarada como uma maneira de se (re)pensar a educação a partir das TIC, principalmente no que diz respeito a dinâmicas educacionais envolvendo programação de computadores. A linguagem LOGO, desenvolvida por Papert e equipe, representou a concretização de seu pensamento, que vislumbrava a ampliação do acesso a tais tecnologias de uma maneira plural e simplificada, porém não menos rigorosa. Enquanto norteador do ambiente de ensino-aprendizagem, o construcionismo e as ferramentas de *software* apresentaram-se como alternativas ao modelo clássico de ensino, procurando fornecer elementos capazes de propiciar uma formação reflexiva, de instigar o debate e o amadurecimento de idéias, de ampliar a capacidade crítica dos educandos, de concretizar a elaboração de produtos do conhecimento (programas de computador, por exemplo) e, o fundamental, possibilitar a grata sensação de se poder fazer algo concreto com programação, compreendendo o modo de se chegar a tal resultado.

No decorrer da realização do curso de LOGO com alunos formandos de um curso de Sistemas de Informação, as impressões colhidas e aqui apresentadas foram identificadas no comportamento de vários alunos e não apenas nos que foram entrevistados. Durante a UC, a atitude da maioria do grupo frente a atividades propostas era positiva e pró-ativa. Nos encontros, os alunos agrupavam-se espontaneamente e trabalhavam cooperativamente na construção dos programas LOGO. As solicitações de intervenções do professor eram em busca de aprimoramentos, novos recursos ou comandos, ou ainda, para pequenos acertos nos programas desenvolvidos. Alguns alunos buscaram aprimorar seus conhecimentos, dedicando parte de seu tempo extraclasse à melhoria de seus programas LOGO. Outros buscaram materiais de apoio na Internet. Após o término do curso, um aluno buscou ajuda para implementar LOGO em uma escola do Nível Fundamental e também com fins de continuidade de seus estudos em curso de pós-graduação.

De acordo com os dados do perfil da turma, definido antes da realização do curso, a maioria afirmou apresentar baixo conhecimento na área. Os alunos indicaram também um baixo nível motivacional ao longo dos cursos. Mediante a isso, o melhor retorno obtido com a UC foi ver o *empowerment* finalmente dominar os alunos, que puderam experienciar a grata satisfação de construir programas de computador, compreendendo o seu fazer.

Em adição aos depoimentos colhidos, evidencia-se dessa maneira, o potencial da linguagem LOGO enquanto ferramenta de ensino-aprendizagem de programação de computadores no Ensino Superior. Isso parece justificar-se em boa parte pelo forte apelo visual da Linguagem, que se mostrou como uma forma de concretizar abstrações inerentes à atividade de programação. Esta característica possibilitou a visualização das ações propostas nos programas, consolidando uma (re)aprendizagem de fundamentos lógicos e de estruturação em programação. Desta forma, a LOGO se constituiu como uma linguagem viabilizadora da motivação e do aprendizado dos estudantes. Isso no sentido de permitir o acesso ao ato de programar de uma maneira explícita e simplificada, porém, não menos rigorosa que as tradicionalmente exploradas, sendo recomendável seu uso combinado a outras linguagens.

## REFERENCIAS BIBLIOGRÁFICAS

CHAVES DE CASTRO, *et al.* Utilizando Programação Funcional em Disciplinas Introdutórias de Computação. *XI Workshop de Educação em Computação - WEI 2003*. Campinas, SP, Brasil.

DELGADO, C. *et al.* Uma Abordagem Pedagógica para a Iniciação ao Estudo de Algoritmos. *XII Workshop de Educação em Computação (WEI'2004)*. Salvador, BA, Brasil.

GUIMARAES, Sueli Édi Rufini.; BZUNECK, José Alonso; SANCHES, Samuel Fabre. Psicologia educacional nos cursos de licenciatura: a motivação dos estudantes. In *Psicologia Escolar e Educacional*. Jun. 2002, vol.6, no.1, p.11-19.

GUIMARAES, Sueli Édi Rufini; BORUCHOVITCH, Evely. O estilo motivacional do professor e a motivação intrínseca dos estudantes: uma perspectiva da Teoria da Autodeterminação. In *Psicologia Reflexão e Crítica*, Porto Alegre, v. 17, n. 2, 2004. p.143-150.

LÉVY, Pierre. *Cibercultura*. São Paulo: Editora 34, 1999.

MALTEMPI, Marcus V., VALENTE, José Armando. Melhorando e Diversificando a Aprendizagem via Programação de Computadores. In: *International Conference on Engeneering and Computer Education - ICECE 2000*. Proceedings (CDROM). São Paulo, Agosto/2000.

MARTINS, Sheila Wesley; CORREIA, Luiz Henrique de Andrade. O Logo como Ferramenta Auxiliar Desenvolvimento do Raciocínio Lógico – um Estudo de Caso. *International Conference on Engineering and Computer Education - ICECE 2003*, Santos. Acessado em 28-12-2004. <http://www.inf.ufsc.br/~scheila/icece2003.PDF>

PAPERT, Seymour. *Logo: Computadores e Educação*. Brasiliense, São Paulo, 1985.

\_\_\_\_\_. *Constructionism: A New Opportunity for Elementary Science Education. A proposal to the National Science Foundation*. Massachusetts Institute of Technology, Media Laboratory, Epistemology and Learning Group, Cambridge, Massachusetts. 1986.

\_\_\_\_\_. *A Máquina das Crianças: Repensando a Escola na Era da Informática*. Ed. rev. Porto Alegre, Artes Médicas, 2008.

PERRENOUD, Philippe. *10 Novas competências para ensinar*. Porto Alegre: ArtMed, 2000.

PETRY, Patrícia Gerent. *et al.* Estudo Exploratório sobre a Depuração de Programas em Logo, in *Anais do VII Congresso Internacional Logo e I Congresso de Informática Educativa do Mercosul*. Porto Alegre, LEC/UFRGS, 1995.

QUARTIERO, E. M. *Da máquina de ensinar à máquina de aprender*. Pesquisas em Tecnologia educacional. Revista Vertentes, São João del-Rei, n.29, p.51-62, jan./jun. 2007.

RODRIGUES JÚNIOR, Methanias Colaço. *Como Ensinar Programação?* Informática - Boletim Informativo Ano I n° 01, ULBRA. Canoas, RS, Brasil, 2002.

SALOMON, G.; PERKINS, D.N.; GLOBERSON, T. Coparticipando en el conocimiento: la ampliación de la inteligencia humana con las tecnologías inteligentes. *Comunicación, language y educación*, n.13.6-22, 1992.

\_\_\_\_\_. *Educational psychology and technology: A matter of reciprocal relations*

Teachers College Record, V.100, n.2, p.222-241, 1998.

\_\_\_\_\_. *Cogniciones distribuidas*. Consideraciones psicológicas y educativas. Buenos Aires: Amorrortu, 2001.

SCHULTZ, Max Rúbens de Oliveira. *Metodologias para Ensino de Lógica de Programação de Computadores*. Monografia de Especialização (Ciência da Computação). Universidade Federal de Santa Catarina (UFSC), Florianópolis, SC, Brasil. 2003, 69p

VALENTE, José Armando. Diferentes usos do computador na educação. In: *Computadores e conhecimento: repensando a educação*. 1ª ed. Campinas, NIED-Unicamp, 1993.