

A UTILIZAÇÃO DA LINGUAGEM DE PROGRAMAÇÃO LOGO NA EDUCAÇÃO GEOGRÁFICA/CATOGRAFICAT

Vandeir Robson da Silva Matias
Mestre em Geografia pelo IGC/UFMG
vandeirgeo@yahoo.com.br

RESUMO

Considerando a importância de se incorporar recursos da tecnologia da informação, particularmente os ambientes computacionais, à práticas pedagógicas, foi realizado um estudo sobre a linguagem de programação LOGO e as habilidades de orientação espacial, tidas como fundamentais para a construção do conhecimento geográfico e cartográfico. O objetivo geral desse estudo compreendeu a verificação das possibilidades e limites de uso do LOGO, em favorecer a emersão de capacidades cognitivas que propiciam a aquisição de noções espaciais topológicas, projetivas e euclidianas concernentes aos estudos geográficos.

Palavras-chave: Geografia, Cartografia e LOGO

THE UTILIZATION OF THE PROGRAMMING LANGUAGE LOGO AT GEOGRAPHIC/CARTOGRAPHIC EDUCATION

ABSTRACT

Considering the importance of incorporating resources of the technology of the information, particularly the atmospheres computacionais, to you practice her pedagogic, a study was accomplished on the programming language LOGO and the abilities of space orientation, had as fundamental for the construction of the geographical and cartographic knowledge. The general objective of that study understood the verification of the possibility of use of the LOGO, in to favour the emersion of cognitive capacities that they propitiate the acquisition of notions space topologicals, projectives and euclidean. Concerns a geographs studies

Key-Words: Geography, Cartography and LOGO

INTRODUÇÃO

Diante da multiplicidade de recursos computacionais¹ que podemos encontrar no mercado, a fim de auxiliar no processo educacional, o presente artigo centrará suas análises em um único software, denominado LOGO². Esse software é uma linguagem de programação que está de acordo com a filosofia da informática educativa, representando um instrumento adequado para explorar atividades espaciais. Além disso, ele apresenta uma concepção bem próxima das idéias de Jean Piaget.

O objetivo geral do artigo é analisar a linguagem de programação LOGO, visando entender quais são os seus limites e possibilidades para o desenvolvimento das habilidades de orientação espacial. Já o objetivo específico será o de verificar qual noção espacial desenvolver-se-á com mais propriedade com a utilização do software referido. Considera-se necessário a intervenção do professor no processo de aquisição das noções de orientação espacial junto aos educandos, visando uma construção de habilidades espaciais bem sucedida. O desenvolvimento desse tipo de habilidade

Recebido em 13/02/2006
Aprovado para publicação em 18/05/2006

hoje, ocorre em uma sociedade inserida na era digital e acredita-se que os recursos computacionais possam auxiliar nesse desenvolvimento, apesar de nem toda a população tem acesso a eles. Pode-se notar também que é exigida cada vez mais do educador uma mediação pedagógica mais eficiente, criativa e interativa. É nesse sentido que deve ser (re) pensado o papel do computador no auxílio da educação geográfica. Informática e educação geográfica devem entrar em sintonia, buscando integrar recursos tecnológicos de forma significativa para a aquisição, processamento, orientação e representação de informações espaciais. O desenvolvimento cognitivo é extremamente importante para a construção das habilidades, como aquelas que tratam da orientação espacial.

ETAPAS DO DESENVOLVIMENTO COGNITIVO ESPACIAL

O desenvolvimento cognitivo que é essencial à construção e à aquisição do conhecimento diz respeito à formação das estruturas mentais que por sua vez é gradual e passa por etapas específicas em busca de ajustamento e equilíbrio. Desenvolver-se cognitivamente significa desenrolar fases, preencher lacunas, fechar ciclos, abrir outros, assimilando noções.

As noções espaciais são elementos que auxiliam o indivíduo a orientar-se no espaço e estão relacionadas com o desenvolvimento cognitivo, pois tais noções não estão presentes na criança desde o seu nascimento, elas são construídas com o tempo, seguindo o desenrolar cognitivo das suas estruturas mentais. Por isso, para que as noções espaciais sejam desenvolvidas é necessário estimular as estruturas mentais, ou seja, propiciar o desenvolvimento cognitivo da criança. O sucesso da aquisição desses conhecimentos pelos indivíduos, depende da interação que estes estabelecem com o ambiente onde estão inseridos.

O espaço é objeto de estudo de vários pesquisadores: matemáticos, filósofos, geógrafos, psicólogos, sociólogos, professores, etc. Ele é de natureza empírica, baseado em uma intuição perceptiva e figurada. Entender, pois, as noções que o caracterizam é primordial para a sua análise, interpretação e representação. De acordo com Boardman (1986, p.5):

Space is itself a concept which is difficult to define. The nature of space has long been debated by psychologists and philosophers, whilst geographers use the concept in different contexts. 'space in a room', 'space for building', 'space between towns' and 'territorial space' illustrate different uses of the word. Space is an idea in the mind which permits the structuring of relationships between objects. Space is subjective and relative, depending on the way it is structured by the mind on a particular occasion. (...) there are variations in the idea of space held by different people at different times. There are similarly considerable variations in the idea of space held by children at successive stages of their intellectual development³.

Para Boardman, a idéia de espaço é variável de acordo com as concepções do pesquisador e do tempo histórico vigente, ou seja, têm-se diferentes significados para esse significante. Tendo como base a psicologia evolutiva, Piaget propõe que o espaço seja do tipo genético, ou seja, espaço é uma experiência subjetiva com base na realidade objetiva. Espaço pode ser também a relação entre os corpos que percebem ou concebem relações para estruturar os corpos, ou ainda pode ser a lógica do mundo sensível.

As habilidades de orientação espacial estão ligadas ao desenvolvimento cognitivo da criança que possui diferentes concepções de espaço de acordo com o seu nível de aquisição de conhecimento (LE SANN, 1989). Desvendar o espaço infantil, que é ativo e operatório, faz parte de uma investigação prática. Piaget e Inhelder investigaram a construção das noções espaciais no Laboratório de Psicologia de Genebra e afirmam que *somente os fatos genéticos são capazes de nos informar sobre os fatores reais da construção do espaço* (1993, p.11).

As noções espaciais mais elementares são chamadas de topológicas, elas se baseiam em correspondências qualitativas de vizinhança, separação, ordem, envolvimento e continuidade. Posteriormente, tem-se o emprego de noções euclidianas (relações métricas, coordenadas, etc.) e

noções projetivas (pontos de vistas, perspectivas, etc.) desenvolvidas concomitantemente.

O desenvolvimento das habilidades de orientação espacial pode ser comparado à ação de se subir uma escada, ou seja, é necessário passar pelos degraus inferiores como condição *sine qua non* para atingir o seu patamar. Essa idéia é expressa por Sisto (1996, p.85):

Dessa maneira os vários níveis de desenvolvimento (que são patamares verticais) são mutuamente dependentes no sentido de que o posterior é uma derivação do anterior e nenhum conteúdo poderá ser organizado nos últimos patamares de desenvolvimento, sem antes terem passado pelo primeiro.

Através da aquisição das noções espaciais é que se pode entender questões referentes ao espaço geográfico. O domínio das habilidades espaciais topológicas, projetivas e euclidianas é subjacente à análise geográfica e essencial para os cidadãos no seu cotidiano, pois no espaço se estabelecem relações de proximidade, separação, ordem, envolvimento, continuidade, lateralidade, reversibilidade, perspectiva, proporção, distância, localização, etc.

A LINGUAGEM DE PROGRAMAÇÃO LOGO: ASPECTOS BÁSICOS

O *LOGO*, *software* desenvolvido por Seymour Papert em 1968 no *Massachusetts Institute of Technology* Estados Unidos, atingiu sua versão mais atraente em 1985 através de um sistema computacional gráfico e atualmente vem passando por constantes modificações. Esse software foi concebido com o propósito de desenvolver o raciocínio lógico em crianças por meio de atividades variadas (PFUHL e TUCCI, 1985). Ele traz uma concepção educacional significativa, através do uso instrumental do computador.

Papert, um dos discípulos de Jean Piaget, estudou Psicologia Genética em Genebra e participou de significativas experiências com crianças, as quais tinham como objetivo verificar a construção do pensamento na criança. As inúmeras experiências realizadas ao lado de Piaget suscitaram em Papert o desejo de criar a linguagem de programação *LOGO*. Tal software permitiria à criança programar o computador, assim, ela estabeleceria um contato com esse recurso e com uma área de conhecimento a desenvolver, seja no campo da Biologia, Geografia, Matemática, Letras, etc. Almeida (1987, p.67) diz que Papert:

Vem fazer uma conjugação importantíssima de teoria e prática de informática com a teoria e prática de Piaget. O computador vem permitir que as crianças que tem acesso a ele construam elas mesmas suas estruturas intelectuais espontaneamente, sem que estas lhe sejam inculcadas. O que não significa que elas sejam elaboradas a partir do nada, ao contrário a criança elabora suas estruturas de pensamento manipulando os materiais que encontram em seu ambiente.

Na Ciência da Computação, programar significa estabelecer uma comunicação com o computador, ou seja, utilizar uma linguagem própria que tanto o sujeito quanto a máquina sejam capazes de decodificar a fim de assimilar uma informação.

Além do processamento de uma lista de comandos, o *software LOGO* permite a incorporação de novos procedimentos durante a programação realizada pelo usuário. Assim, as possibilidades do programa dependem da habilidade e interesse de quem opera com essa linguagem (BRAÚLIO, 1989).

Na linguagem *LOGO*, a criança aprende a programar o computador, dando determinados comandos a uma tartaruga, que é a sua referência na tela. Segundo Papert (1985, p.26):

(...) A tartaruga é um animal cibernético controlado pelo computador. Ela existe dentro das miniculturas cognitivas do "ambiente *LOGO*", sendo *LOGO* a linguagem computacional que usamos para nos comunicar com a tartaruga. Essa tartaruga serve ao único propósito de ser fácil de programar e boa para pensar.

O contato realizado entre sujeito e tartaruga expressa o modo como o usuário se relaciona-se com esse objeto (a tartaruga), revelando o seu modo de pensar diante de uma determinada situação. O *LOGO* é ideal para desenvolver atividades espaciais, pois a tartaruga irá revelar o

desenvolvimento das habilidades de orientação espacial do sujeito, favorecendo o diagnóstico do processo cognitivo. Segundo Freire (2001, p.125):

A atividade de se deslocar no lugar da Tartaruga e identificar seu próprio corpo com o dela faz com que o sujeito realize uma atividade reflexiva que envolve conhecimentos espaciais, proprioceptivos e lingüísticos que podem ser extremamente úteis para a (re) construção das atividades espaciais.

Na filosofia do *software* em questão, o computador não é meramente uma máquina de ensinar, como observa-se na concepção de muitos softwares que são despejados no mercado das novas tecnologias da educação (GOODYER, 1986). No *LOGO*, o computador deixa de ser a máquina de ensinar para ser a máquina de aprender. Assim, é a criança que fará com que o computador seja uma ferramenta que possibilitará a (re) formulação de seus conhecimentos, sempre voltando ao início da programação quando necessário (VALENTE e VALENTE, 1988).

A máquina coopera com o sujeito quando o auxilia no estabelecimento de uma nova relação com o conhecimento que não está pronto nem acabado. Na concepção *LOGO*, o conhecimento está sempre fluindo, incorporando novas idéias e valores à medida que barreiras vão sendo superadas.

Nesse ambiente do *software*, os sujeitos pensam mais sobre o que estão realizando, principalmente porque sempre precisam resolver uma situação problema. Por conseguinte, ocorre o estímulo das estruturas cognitivas através do processo de aprendizagem pela mediação pedagógica da informática. Por ser um software interativo⁴, o *LOGO* é bem aceito nas escolas, pois permite ao sujeito refletir sobre todo o processo envolvido na programação do computador, voltando e pensando sobre os erros executados quando necessário.

Analisando as características do software, observa-se a possibilidade de um espaço de reflexão sobre a atividade que está sendo desenvolvida. E essa reflexão é que produz um auto-conhecimento, pois o processo que é o essencial e não o resultado final. Sendo assim, o que o *LOGO* procura é desenvolver sujeitos pensadores ativos e críticos, maximizando o seu potencial intelectual, desenvolvendo suas habilidades e fazendo com que adquira novas, em suma, tornando o cidadão competente para a sociedade contemporânea. Segundo Almeida (1987) para Piaget a educação deve proporcionar justamente a reflexão do processo pedagógico, formando cidadãos livres para exercer os seus direitos e deveres na sociedade.

A produção de ambientes de aprendizagem no computador não substitui a realidade da aula convencional e a necessidade de operar com o real. Ambientes computacionais ajudam a produzir um ato novo de pensar a aperfeiçoar as estruturas cognitivas. Portanto, a linguagem computacional *LOGO* deve ser usada como uma ferramenta auxiliar no processo de ensino e aprendizagem, devendo ser complementada com outros processos, como os de socialização e cultura.

AVALIAÇÃO DE SOFTWARES EDUCACIONAIS

As conquistas tecnológicas da humanidade, sobretudo aquelas relacionadas à microeletrônica, no período técnico-científico-informacional, possibilitaram ganhos cognitivos aos indivíduos, aumento da produção industrial, surgimento de novas profissões, desaparecimento de outras, aceleração do fluxo de informações, etc. Dentre os recursos computacionais que contribuíram para a concretização dos fatos citados, destaca-se os *softwares*, que são ferramentas computacionais.

Alguns *softwares* muitas vezes possuem uma fundamentação pedagógica, finalidade didática, relação com o currículo, visando favorecer o processo de ensino-aprendizagem. Existem também outros tipos de *softwares* que são utilizados sobretudo em setores administrativos como os bancos de dados e as planilhas eletrônicas.

No processo educacional, percebe-se a tendência de conjugação desses diversos tipos de *softwares*. Ocorre, assim, a busca por sistemas híbridos, através dos quais o indivíduo navega por todos os *softwares*, tentando captar o que há de melhor em cada um para, então, construir seus projetos e dar respostas para as questões problema. Essa é uma tendência que pode caracterizar-

se como *transcomputacional*, fazendo referência ao termo transdisciplinaridade⁵.

A produção e avaliação de *softwares* são objetos de investigação de vários pesquisadores desde a incorporação das novas tecnologias no processo educacional. A partir dos anos 90, observa-se a existência de uma diversidade de *softwares* educacionais no mercado. Fato esse que suscitou uma maior atenção quanto a avaliação dos mesmos. Oliveira diz que (2001, p119):

A avaliação de um SE⁶ é um processo que se inicia antes mesmo da sua criação. No momento em que a equipe produtora do software educativo é escolhida, os critérios básicos que direcionarão seu desenvolvimento e que servirão conseqüentemente como parâmetro para a sua avaliação inicial estarão refletidos no perfil daquele grupo.

Os *softwares* devem ser avaliados com base em critérios que servem para mensurar qualitativamente ou quantitativamente, a presença ou ausência de uma característica no *software*. O objetivo da avaliação é verificar a qualidade do *software*. Entende-se por qualidade de software, segundo Rocha (1987), citada em Campos (1994, p.78), *um conjunto de propriedades a serem satisfeitas em determinado grau, de modo que o software satisfaça as necessidades de seus usuários*.

Pesquisadores ou estudiosos sobre o assunto, na busca de desenvolver metodologias e estabelecer critérios para avaliação da qualidade de *softwares*, definiram categorias e fatores relevantes para esse fim. Oliveira (2001), baseando-se em Guimarães et al. (1987), Campos e Rocha (1993), Fitzgerald (1997) e outros, formula critérios inseridos em quatro categorias de análise: *interação aluno-SE-professor, fundamentação pedagógica, conteúdo e programação*. Essas categorias levam em conta determinados fatores e sub-fatores que funcionam como critérios de avaliação.

A categoria interação aluno-SE-professor diz respeito ao papel do professor na facilitação da aprendizagem do aluno, à possibilidade de aprendizagem em conjunto e à possibilidade de interação entre o SE e usuários. Um dos critérios de análise dessa categoria é a facilidade de uso do software.

O *software* precisa apresentar instruções claras e facilidade para a sua utilização por parte dos usuários. São necessários elementos básicos, como instruções, ícones, dicas, linguagem adequada, organização, navegabilidade, mapeamento, memória e integração, os quais irão maximizar o trabalho tanto do professor quanto do usuário.

Os recursos motivacionais, outro critério importante, visam manter o interesse dos usuários em relação ao *software*. Esse critério é essencial diante da ascensão dos programas de comunicação⁷ que oferecem nível de motivação e atratividade permanente durante a sua utilização. Assim, para motivar, é necessário que o *software* apresente principalmente desafios pedagógicos e um bom *lay out* de tela como forma de atrair o interesse dos indivíduos.

Os recursos devem encontrar-se presentes no software de maneira adequada. O objetivo é facilitar a aprendizagem dos alunos e não desviar a sua atenção para elementos desnecessários, que não os ajudarão a construir habilidades. Para fazer essa análise é necessário inserir o critério, *adequação dos recursos de mídia às atividades pedagógicas* no processo de avaliação de *softwares*. Itens como presença de hipertextos, imagens, animações, som, etc. devem ser verificados.

A interatividade social e o favorecimento do papel de facilitador do professor são critérios que completam essa primeira categoria de análise (interação-SE-professor). O primeiro considera a possibilidade de desenvolver o trabalho em grupo ao utilizar o *software*. O segundo considera a importância do professor em desenvolver habilidades e competências na utilização do recurso, verifica também a presença de orientações didático-pedagógicas e indicações bibliográficas complementares.

Fundamentação pedagógica e conteúdo são duas categorias de análise que estão interligadas. A primeira corresponde aos pressupostos pedagógicos que permeiam o *software*. A segunda

categoria refere-se à possibilidade de trabalho com determinada área de conhecimento, visando desenvolver competências.

Na fundamentação pedagógica, é necessário verificar apenas se o programa apresenta clareza epistemológica. O ideal é encontrar no *software* o paradigma no qual baseia o recurso, a opção didática-pedagógica, etc. Essa categoria será decisiva para a escolha do *software* por parte da equipe pedagógica da instituição de ensino.

O conteúdo engloba critérios que se complementam na sua totalidade, tais como pertinência do conteúdo, correção do conteúdo, estado da arte, adequação à situação de aprendizagem, variedade de abordagens e conhecimentos prévios.

O *software* deve ser adequado ao trabalho do professor e o conteúdo deve ser pertinente aos objetivos educacionais. Talvez essa seja a categoria (conteúdo) mais importante, uma vez que a análise desse critério representa condição *sine qua non* para a utilização do *software* pelo educador.

Soma-se ao critério adequação ao trabalho do professor, à adequação à situação de aprendizagem, em especial ao currículo escolar, que muitas vezes norteia o processo de ensino e aprendizagem. O desenvolvimento das habilidades que o educador deseja atingir, deve estar inserido em um determinado contexto de aprendizagem, no qual possa se verificar o nível cognitivo do aluno em relação ao currículo proposto.

Verificar as várias abordagens que um *software* possui é um critério pertinente de se investigar, pois influenciará no desenvolvimento dos alunos que utilizarem o programa. Tais informações devem estar contidas no *software* para que o usuário saiba o conhecimento prévio necessário para operar com o recurso.

Por fim, soma-se a essa categoria conteúdo, critérios como correção do conteúdo e estado da arte. No primeiro, ocorre simplificações do conteúdo sem perda e empobrecimento do mesmo. No segundo critério verifica-se a atualidade do conteúdo e inovações na sua forma de abordagem.

A última categoria sugerida por Oliveira (2001) é a programação. Ela refere-se às especificações técnicas existentes para que um *software* tenha um bom funcionamento. Nessa categoria, pode-se observar dois critérios importantes: a confiabilidade conceitual e a facilidade de uso.

A confiabilidade conceitual diz respeito à veracidade das informações contidas no software que vão gerar o seu desenvolvimento. Nesse critério, destacam-se itens como fidedignidade e integridade.

A facilidade de uso é um critério já verificado na primeira categoria. Contudo, nesse momento, é preciso observar outros itens para avaliar a facilidade de integração do usuário com o programa. Dentre os itens, encontram-se destacados a legibilidade, a manutenibilidade, a operacionalidade, a reutilizabilidade, o custo/benefício, a avaliabilidade, a modularidade e a documentação.

Outra metodologia de avaliação de software menos detalhada é a de Campos (1994), construída com referência em Rocha (1987). Essa metodologia leva em conta três categorias: utilizabilidade, confiabilidade conceitual, confiabilidade de representação.

A primeira categoria verifica a característica de utilização do *software* ao longo do tempo, englobando a operacionalidade, a eficiência, a manutenibilidade, a portabilidade e a rentabilidade. A segunda categoria confiabilidade visa analisar se houve implementação satisfatória do que foi especificado, para isso utiliza critérios como integridade e fidedignidade, os mesmos utilizados por Oliveira (2001). Já a terceira categoria confiabilidade de representação analisa a facilidade de entender os recursos existentes no *software*, sendo verificada através do critério de legibilidade.

Existem algumas diferenças entre as duas metodologias consultadas. As *categorias* em Campos (1994) são consideradas *critérios* em Oliveira (2001); ainda não se nota em Campos (1994) uma preocupação pedagógica, como na primeira metodologia descrita, a de Oliveira (2001). Apenas no item eficiência de processamento percebe-se a inserção do pedagógico como item de verificação em Campos (1994).

Pode-se verificar dois métodos de avaliação de software dois tipos deles: avaliação formativa e objetiva. Oliveira (2001, p.121) caracteriza esses dois tipos:

A avaliação formativa compõe-se, entre outros recursos, de entrevistas, questionários, acompanhamento de perto do desempenho dos usuários durante a utilização do software (...). A avaliação objetiva, por sua vez, é realizada com a utilização de listas de critérios, disponibilizadas por diferentes autores. A composição dessas listas reflete a posição teórica de cada um deles, em relação a como se dá o conhecimento.

O primeiro método representa uma avaliação mais interativa e participativa, pois envolve mais de um indivíduo no processo de avaliação. É um processo de acompanhamento sistemático do sujeito por parte do professor, com o objetivo de perceber como ele reage ao utilizar o recurso e quais estruturas cognitivas emergem desse momento de análise. O segundo método representa uma avaliação específica, diagnóstica, com parâmetros definidos e embasados em uma concepção teórica do avaliador. É uma maneira de averiguar a presença ou ausência de determinado parâmetro em um software.

Quaisquer que sejam os tipos de avaliação utilizados e critérios sugeridos, os professores são essenciais à produção e à avaliação dos softwares educacionais. São eles que vão definir os critérios mais pertinentes aos objetivos pedagógicos estabelecidos na utilização desse recurso computacional. De maneira geral, os critérios devem privilegiar a construção do conhecimento e desenvolvimento de habilidades e competências. Contudo, o avaliador mais preciso sempre será o usuário do software, ou seja, o educando.

O procedimento metodológico utilizado terá como base a avaliação objetiva. Acredita-se que ela seja a mais propícia para investigação do objetivo proposto, pois disponibiliza um elenco de critérios e desperta a necessidade de criação de outros para análise do software. Assim, esse tipo de avaliação possibilitará a realização de um trabalho de cunho qualitativo e quantitativo.

RESULTOS E DISCUSSÕES

As especificações técnicas são essenciais para o bom funcionamento do software e para o manuseio eficiente do mesmo por parte do usuário. Tanto o MultiLogo 3.0 quanto o SuperLogo 3.0 apresentam bom desempenho nessa categoria, como em outras analisadas, sendo os seus pontos falhos toleráveis, pois os mesmos não comprometem o trabalho pedagógico. Ver QUADRO 1.

No que tange ao critério facilidade de uso, composto por oito itens, as versões analisadas, no geral, apresentam alta qualidade técnica. Embora encontrem-se problemas, como mapeamento e adequação cognitiva, eles podem ser contornados pelo facilitador da aprendizagem. O papel do facilitador, nesse caso, será o de desenvolver atividades possíveis de serem realizadas pelos educandos, despertando neles outras estruturas cognitivas com a finalidade de alçar atividades mais complexas futuramente.

Alguns critérios, como instruções, menu, auxílio, capacidade de armazenamento de respostas, integração e documentação, estão presentes de maneira adequada nas duas versões. Esse fato proporciona ao usuário uma maior segurança na interação com o recurso utilizado e confiança ao realizar os projetos almejados. Apesar de serem importantes, o critério mapeamento assim como a adequação cognitiva não foram considerados satisfatórios em nenhuma das versões analisadas nessa pesquisa.

No caso do mapeamento, o *software* deveria apresentar um caminho a ser seguido pelo usuário e mostrar também em qual parte da programação o usuário se encontra. Entretanto, ele utiliza somente um recurso denominado *procedimento* que mostra tudo que o usuário já desenvolveu ao longo de sua trajetória no software. Outra possibilidade, de constatar o mapeamento, é acessar o comando *Estado*, no SuperLogo 3.0 ou *propriedades*, no MultiLogo 3.0, os quais apresentam informações como direção e posição da tartaruga. Contudo, mesmo utilizando esses comandos, não é possível obter informações sobre quais caminhos ainda estão disponíveis.

Quadro 1
 Categoria Técnica - Síntese das avaliações

Critério	Itens	Resultado das avaliações
Facilidade de uso	Instruções	Sim
	Menu, botões e ícones	Sim
	Auxílio e dicas	Sim
	Mapeamento	Parcialmente
	Capacidade de armazenamento de respostas	Sim
	Integração	Sim
	Adequação cognitiva	Não
Recursos motivacionais	Documentação	Sim
	Atratividade	Sim
	Desafios pedagógicos	Sim
	Layout de telas	Sim
	Existência de mensagem de erro	Sim
Confiabilidade conceitual	Fidedignidade	Sim
	Integridade	Sim

Produzido por: Vandeir Robson da Silva Matias

Quanto à adequação cognitiva, ela se encontra ausente na concepção LOGO. O usuário realiza sua programação de acordo com o nível em que se encontram suas estruturas cognitivas, todavia, essas são instigadas pelos comandos existentes, auxiliando o indivíduo na construção de habilidades.

No ambiente LOGO, pode-se desenvolver projetos básicos em modo bidimensional (2D), ou programações mais complexas em modo tridimensional (3D). O modo bidimensional comporta apenas dois eixos X e Y, já o modo tridimensional engloba três eixos X, Y e Z. É a estágio de desenvolvimento cognitivo do usuário que será levado em consideração no momento de operação com o software, entretanto, observa-se que é necessário desenvolver primeiro, habilidades em modo 2D e posteriormente em 3D.

Botões e ícones, dois itens que facilitam o acesso do usuário aos comandos, estão disponíveis nas duas versões analisadas do LOGO, contudo eles encontram-se melhor dispostos e em maior quantidade no MultiLogo 3.0. Nessa versão, os botões e ícones apresentam também maior atratividade, pois são maiores e apresentam cores que chamam mais atenção do usuário, ver (Figura 1).

Outra característica importante constatada nas análises é a presença de integração do LOGO com outros recursos computacionais, o que irá facilitar muito o trabalho do professor. A linguagem de programação LOGO pode ser conjugada com recursos de comunicação, como a internet ou mesmo editores de textos e planilhas eletrônicas. Se durante o processo de ensino e aprendizagem com LOGO for necessário utilizar outros recursos, basta acessá-los através do software.

O software possui também boa facilidade de uso, pois verifica-se que 75% dos itens do critério *facilidade para o uso* estão presentes nele de maneira clara e eficiente. Apenas um item foi observado parcialmente, o mapeamento, que corresponde a 12,5% dos itens observados. E a adequação cognitiva não foi observada no software, representando 12,5% dos itens analisados.

O critério *recursos motivacionais* abrange itens como *atratividade*, *desafios pedagógicos*, *layout de tela*, *existência de mensagem de erro*. Todos eles encontram-se presentes nas versões MultiLogo 3.0 e SuperLogo 3.0. O software, em relação a esses itens, apresenta alta qualidade técnica, não sendo encontrado nenhum problema que dificultará o uso do recurso, ou seja, 100% dos itens encontram-se de forma satisfatória no software.

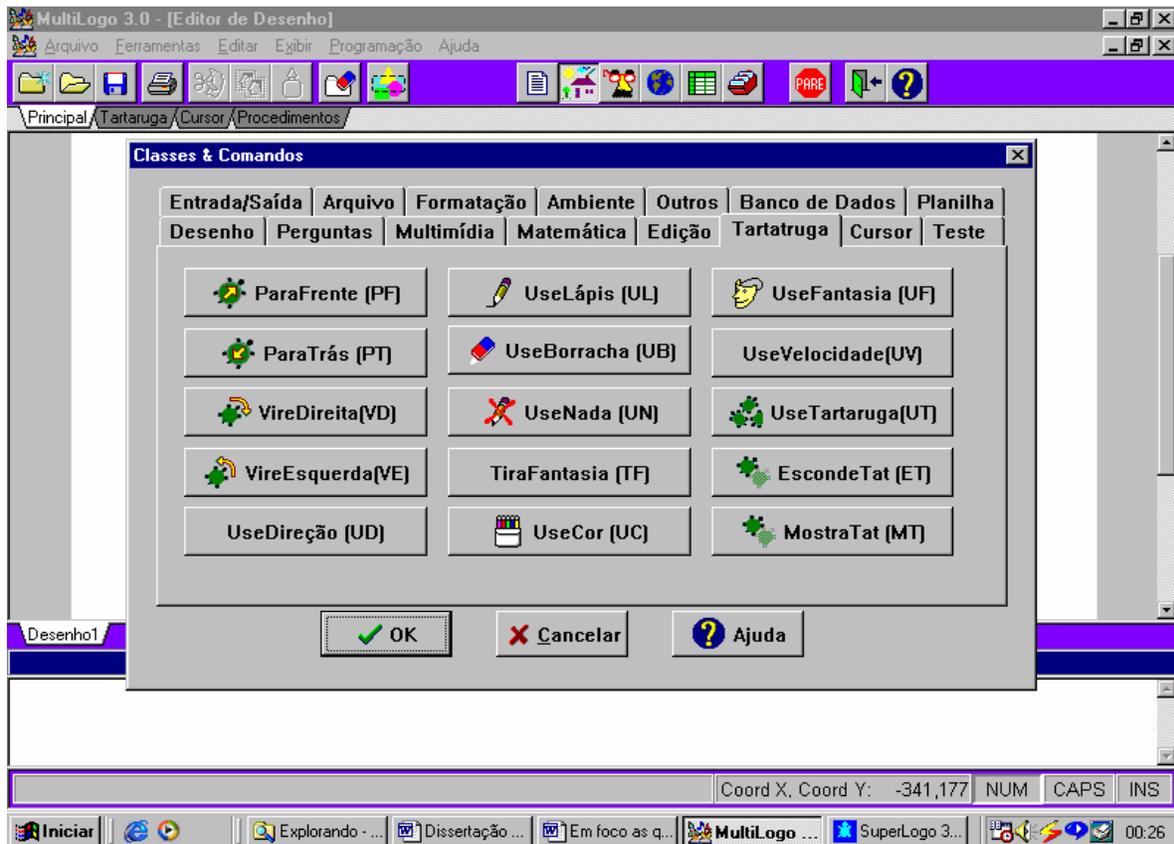


Figura 1 - Menu, ícones e botões - MultiLogo
Fonte: TECSO-Tecnologia de software - MultiLogo 3.0

Alguns itens, do critério *recursos motivacionais*, merecem destaque. Um deles diz respeito à capacidade do software em auxiliar o usuário na questão do erro. Como se sabe, é necessário a programação de alguns comandos para que o *software* os reconheça e os execute de maneira satisfatória. Caso não ocorra a programação, o item mensagem de erro, aparecerá através de mensagens na tela como *não me disse o que fazer com* ou *ainda não aprendi*.

Outro item importante dentre os *recursos motivacionais* é o *lay out* de tela que deve mostrar-se de maneira adequada, apresentando equilíbrio para não atrair a atenção do aluno para elementos supérfluos. As cores principalmente precisam ser atrativas, porém sem exageros. Uma outra característica é que o próprio usuário pode organizar o *lay out* da maneira que melhor lhe convém, tanto no SuperLogo quanto no MultiLogo.

Quanto ao critério *confiabilidade conceitual*, não foi verificada ausência de nenhum dos itens sugeridos na pesquisa: *fidedignidade* e *integridade*. O que quer dizer que suas especificações técnicas são confiáveis em 100%, assim como suas explicações filosóficas e pedagógicas que permearam o desenvolvimento do software. Além disso, é possível obter resultados positivos e exatos de acordo com a programação realizada, ou seja, se o programa sugere a possibilidade de construção de desenhos em perspectivas.

O que fica nítido é que, apesar da ausência de alguns itens importantes, não há comprometimento da eficiência do LOGO em nenhuma de suas versões analisadas. Acredita-se que a versão do

MultiLogo 3.0 possa ser mais atrativa por possuir maior facilidade de uso e, principalmente, pela presença de botões e ícones mais destacados no *software*. Recomenda-se, portando, o uso do MultiLogo nos primeiros contatos dos usuários com o *software* e, posteriormente, pode-se inserir o SuperLogo sem perda de qualidade técnica e pedagógica.

RESULTOS E DISCUSSÕES - ASPECTOS EDUCACIONAIS

É possível desenvolver todas as habilidades de orientação espacial utilizando a linguagem de programação LOGO. Observa-se que algumas habilidades, como as projetivas e euclidianas, podem ser desenvolvidas mais que outras (cf Figura 2)⁸.

As noções espaciais chamadas de *vizinhança* e *separação*, as primeiras que aparecem no espaço topológico, possuem baixo potencial de desenvolvimento na Linguagem LOGO e seu desempenho recebeu o índice de avaliação 0,25. Percebe-se que o *software* não utiliza comandos que abrangem referências qualitativas como perto, longe, separado e afastado, relacionadas às noções de vizinhança e separação. O usuário, pode utilizar referências ligadas à vizinhança e à separação na construção de seus projetos, posicionando e deslocando a tartaruga pela janela gráfica (cf. Figura 3).

Nessa figura, a menina de vestido rosa está próxima do menino. Utilizando comandos básicos da tartaruga, como Pd (*Paradireita*), Pf (*Parafrente*), Pe (*Paraesquerda*) e Pt (*Paratrás*), é possível transportar tanto o menino quanto a menina para algum ponto de referência do desenho como a casa, a lagoa, os animais, etc. Esse tipo de atividade ajuda o educando a despertar percepções que lhe auxiliarão na construção de noções de vizinhança e separação.

O desenvolvimento da habilidade de *sucessão espacial* em LOGO segue as de desenvolvimento das noções de *vizinhança* e *separação*. O que é possível quanto a essa habilidade é a construção de ambientes e situações, nos quais o indivíduo possa operar com as noções de ordem. Entretanto, tem-se a possibilidade de criação de diversos objetos com tamanhos variados, utilizando o comando *mudeproporção*.

A linguagem de programação LOGO possui mais recursos para o desenvolvimento da habilidade denominada *envolvimento*. Noções, como dentro, fora, limite e fronteira, podem ser trabalhadas através de jogos, atividades criadas no programa e utilizando comandos que auxiliam o desenvolvimento dessa noção. Esses comandos que podem ser: *tiralimite*, *amentaliminte*, *ponhalimite*, modificam o espaço na janela gráfica que a tartaruga poderá percorrer. Além disso, eles se relacionam com a noção de envolvimento tal como descrito por Piaget e Inhelder (1993).

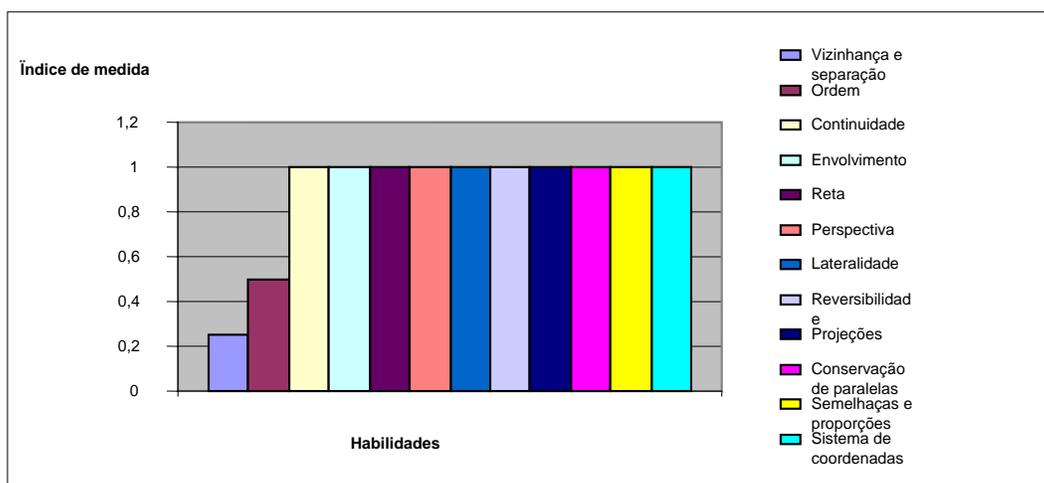


Figura 2 - Habilidades de orientação espacial

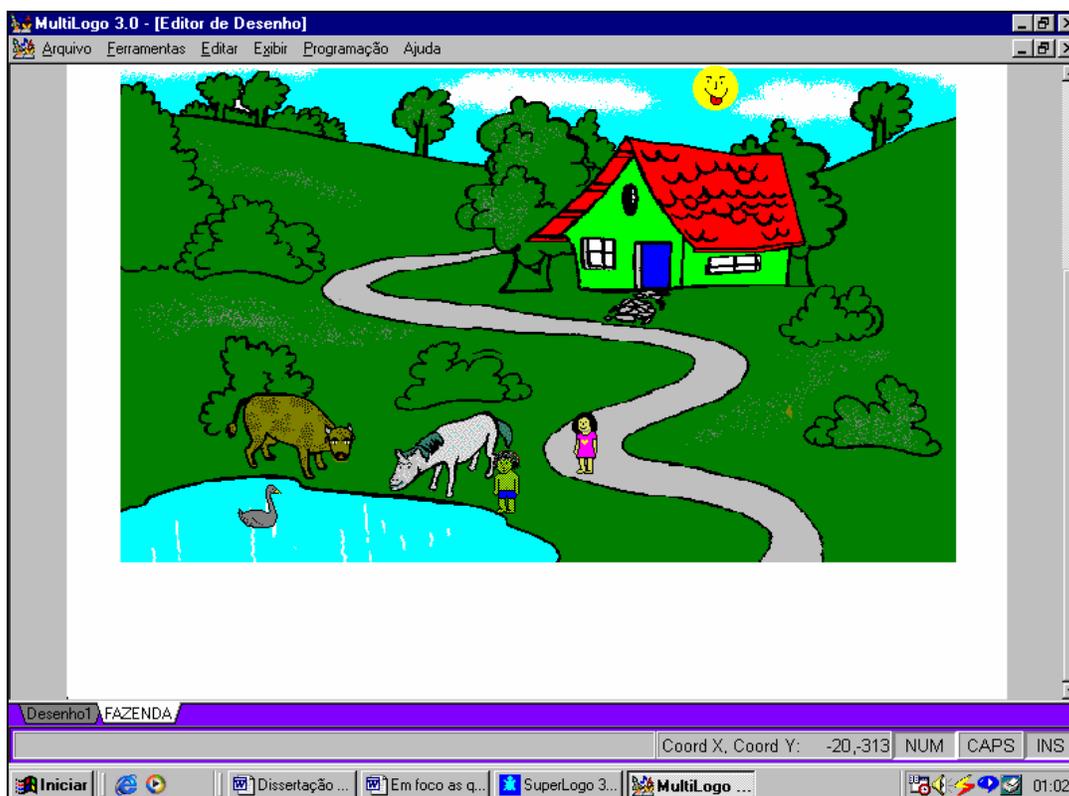


Figura 3 - Fazenda
Fonte: TECSO -Tecnologia de software - MultiLogo 3.0

A *continuidade*, última noção de espaço topológico, pode ser desenvolvida utilizando-se um pouco de cada noção construída anteriormente, pois representa a síntese de cada item das habilidades topológicas. Um ponto alto do LOGO, é a construção de figuras geométricas diversas.

Trabalhar com modo tridimensional ativa a percepção do indivíduo para outros pontos de vistas, auxiliando o desenvolvimento da principal noção dentro do espaço projetivo que é a perspectiva. O LOGO possui comandos específicos para isso, como *logo3d*, *mude direção xyz*, *mude z*, *coordenada z*, *mude posição xyz*, *mude posição z*, *direção para xyz*, *direção de rolagem*, *direção de cabeceio*, *orientação*, *role para direita*, *role para esquerda*, *ative tartaruga -1*, *ative tartaruga -2*, *arco elipse* entre outros.

Não deixando dúvidas quanto a sua eficiência e excelência no desenvolvimento das perspectivas. O item perspectiva obteve índice de medida total igual a 1.0. Ao inserir-se um novo eixo nos desenhos, que é o Z, as construções passam a ganhar profundidade e a idéia de perspectiva (cf. Figura 4). O ideal é que os indivíduos construam desenhos no modo bidimensional e, posteriormente, no modo tridimensional para realizar comparações entre eles.

O item técnico *mensagem de erro*, é essencial para o desenvolvimento da atividade reversível nesse ambiente computacional. O erro faz com que o indivíduo volte à programação realizada, ativando o pensamento reversível que é extremamente complexo e construído após a noção lateralidade. As duas versões do software possuem, apresentados de maneira diferenciada, alguns comandos específicos que se baseiam na reversão. Comando, como o *paraesquerda* e *paradireita*, *vire para esquerda* e *vire para direita* (cf. Figura 5), auxiliam na construção da noção de *reversibilidade*.

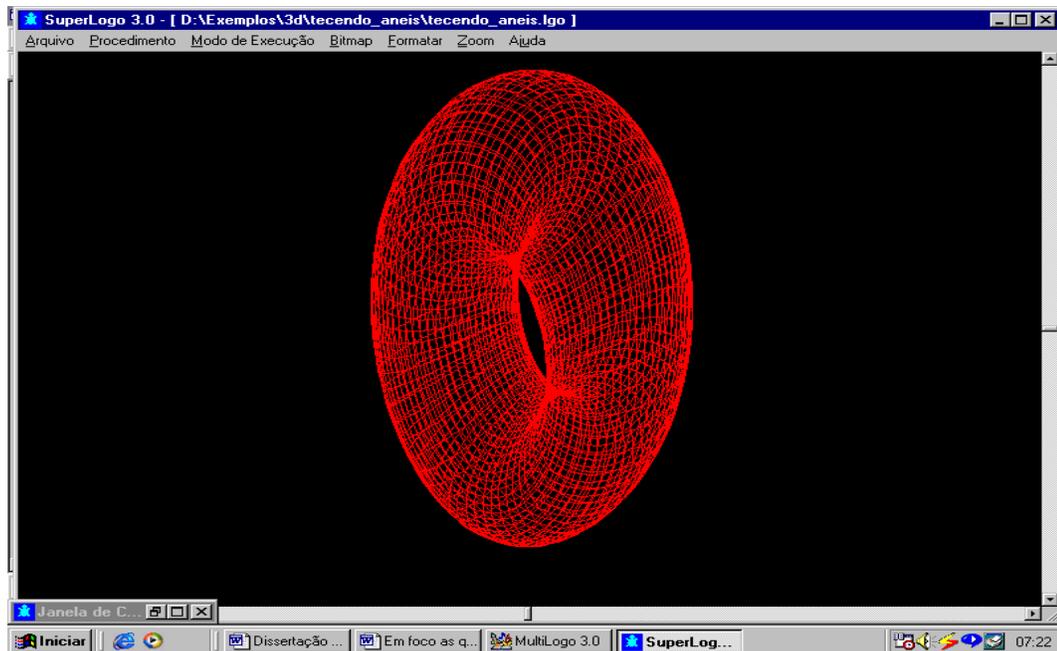


Figura 4 - Anel em perspectiva
Fonte: NIED/UNICAMP. Kit educacional SuperLogo 3.0

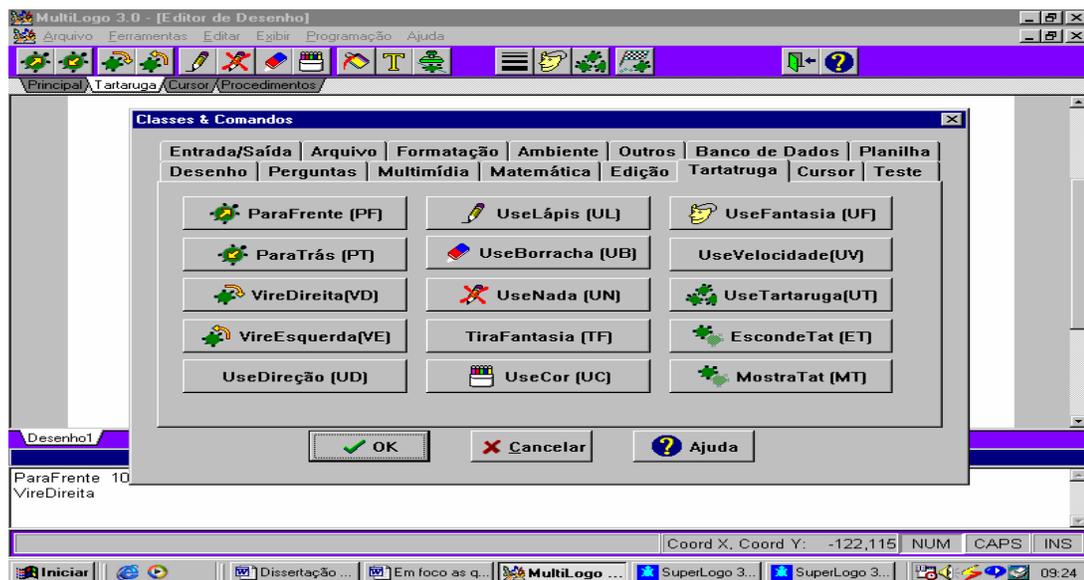


Figura 5 - Comandos reversíveis
Fonte: NIED/UNICAMP. Kit educacional MultiLogo 3.0

O sistema de coordenadas é a última característica a ser desenvolvida na habilidade de orientação espacial euclidiana. E os sistemas horizontais e verticais, que fazem parte das coordenadas, começam a ser desenvolvidas no LOGO sobretudo com as construções de retas de tamanhos e distâncias variados, passando por noções de perspectivas, laterais e reversíveis. Nos softwares o indivíduo pode manipular a tartaruga para que ela ocupe posições diferenciadas na janela gráfica. Essa posição no LOGO é chamada de estado e envolve um posicionamento, seguindo as coordenadas X, Y e a Z (quando utilizado no modo tridimensional), assim como outros comandos especificados na análise, como *paracima* e *parabaixo*(cf . Figura 6).

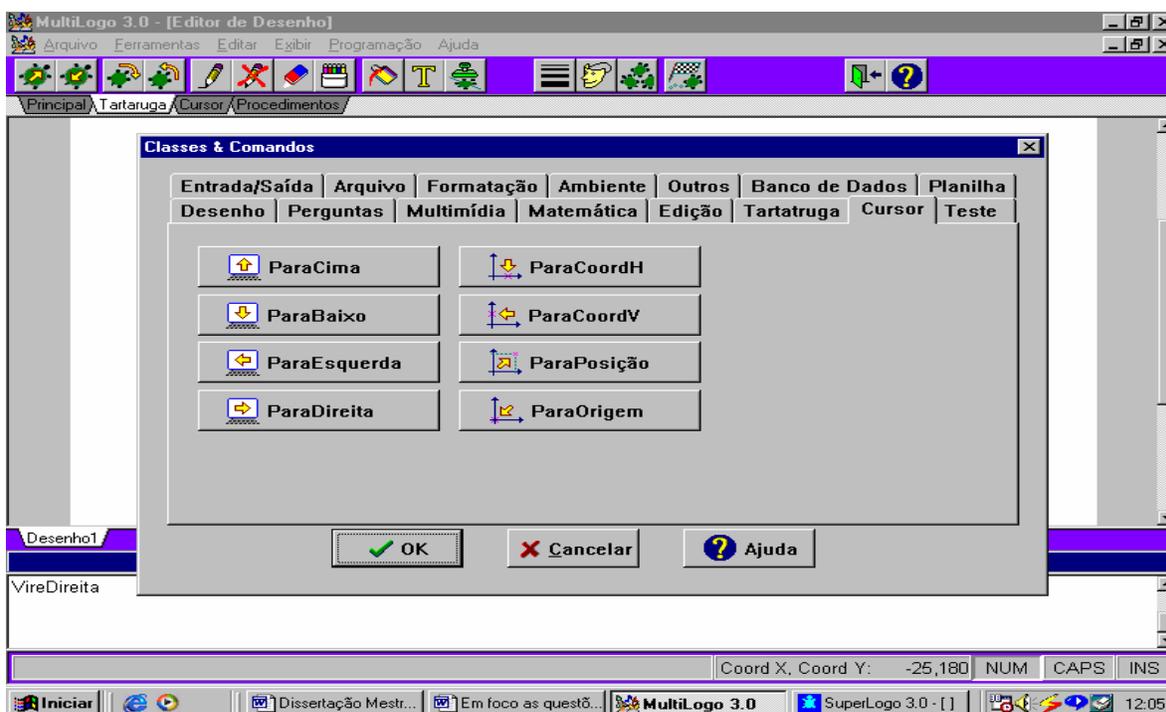


Figura 6 - Comandos dos sistemas de coordenadas em LOGO.
Fonte: TECSO - Tecnologia de softwares - MultiLogo 3.0

CONSIDERAÇÕES FINAIS

A percepção do espaço é essencial à Geografia nas suas múltiplas funções. As construções geométricas, por sua vez, auxiliam na estruturação do objeto de estudo geográfico (o espaço). Espaço esse que a Geografia é um fundo, uma estrutura para edificação dos elementos geométricos; lugar onde os corpos se movimentam com intencionalidade, dando vida e sentido à paisagem, ao lugar, ao território, às regiões, etc.

Para que os indivíduos possam pensar espacialmente e posicionar-se no espaço geográfico, é necessário desenvolver habilidades de orientação espacial, como noções topológicas, projetivas e euclidianas. Bicudo (2000, p.37) diz que:

(..) O vivido no mundo preenche de sentido as experiências de espaço e suas modalidades como lugar - em cima, embaixo, deste lado, daquele lado, na frente, atrás, dentro, fora -, distância -aqui, ali, longe, perto, antes e depois e

respectivos desdobramentos em termos espaciais.

Esses termos espaciais citados podem ser desenvolvidos no ambiente *LOGO*, como demonstrou o trabalho, cujo objetivo é comprovar que esse software possui comandos expressivos que auxiliam na construção das habilidades de orientação espacial.

No que se refere às noções topológicas, fica nítida a possibilidade de construção bem sucedida das noções espaciais de *envolvimento* e *continuidade*, pois os comandos para essa finalidade apresentam-se, no software, expostos de maneira clara e objetiva. Além disso, o programa *LOGO* pode ajudar também a desenvolver, de modo eficiente, as noções de ordem, separação e vizinhança.

Noções euclidianas, como os sistemas de coordenadas, semelhanças e proporções, por serem pautadas em construções matemáticas, podem ser desenvolvidas com bastante sucesso no *software*. No geral, o seu índice de aproveitamento corresponde a 1.0, o que significa excelência do *software* em desenvolver essa habilidade.

Contudo, o usuário precisa ter um conhecimento maior da linguagem de programação *LOGO*, um desenvolvimento cognitivo mais apurado, assim como todas as outras noções espaciais anteriores desenvolvidas para, ao utilizar o *software*, obter êxito em construções do espaço euclidiano.

O *LOGO*, também é um excelente *software* educacional para desenvolver as noções projetivas, sendo o seu índice de eficiência equivalente a 1.0. Essas noções destacam-se no *LOGO*, porque alguns dos seus comandos sugerem claramente operações projetivas, principalmente aquelas relacionadas a noções de perspectiva, reversibilidade e lateralidade. Verifica-se também que para se desenvolver qualquer projeto, simples ou complexo, é necessário utilizar comandos considerados projetivos, como *paraesquerda* e *paradireita* entre outros.

Por fim, constata-se que, para fins pedagógicos, não há variações relevantes entre MultiLogo e SuperLogo. Ambas as versões são convenientes ao desenvolvimento de habilidades espaciais, como as noções de envolvimento, ligadas ao espaço topológico; projeções, semelhanças, proporções e sistemas de coordenadas do espaço euclidiano e, com mais clareza, as noções de reversibilidade, lateralidade e perspectivas, as quais fazem parte do espaço projetivo.

A maior parte dos softwares educativos, não permitem ao usuário, uma reflexão dos seus erros. Verifica-se apenas estímulos, ou seja, caso o educando acerte, ele recebe palmas, caso erre, recebe vaias. Esse tipo de estímulo é muito comum e não deve ser utilizado, pois não se encaixa no objetivo de um processo de ensino e a aprendizagem e condizente com um novo espaço de aprendizagem. O software *LOGO* apresenta uma concepção contrária à exposta, pois permite trabalhar com ênfase a questão do erro e as possibilidades de aumento de esquemas de assimilação e acomodação, temas tão expostos por Jean Piaget ao longo de suas obras.

Os resultados teóricos alcançados deixam a sensação de que há muito a se fazer. Primeiramente, a pesquisa é objetiva, pois não realizou um experimento empírico com educandos. É pertinente analisar o comportamento dos usuários frente a esse recurso e sua subjetividade diante dessa nova tecnologia. O ideal seria a conjugação de uma análise teórica com uma empírica.

REFERÊNCIAS

ALMEIDA, Fernando José. *Educação e informática: os computadores em sala de aula*. São Paulo: Cortez. 1987.

BICUDO, Maria Aparecida Viggiani. *Fenomenologia- confrontos e avanços*. São Paulo: Cortez, 2000.

BOARDMAN, David. *Graphicacy and Geography Teaching*. London:Croom Helm, 1986.

BRAÚLIO, Bernadete T. Lemos. *Informática na Educação: O uso da Linguagem computacional LOGO no Estudo das Operações Cognitivas da Fase Lógico-Formal, Tal como Descritas por Piaget*. Dissertação de Mestrado, FAE-UFMG, 1989.

- CAMPOS, Gilda Helena Bernardino de. *Metodologia para avaliação da qualidade de software educacional. Diretrizes para desenvolvedores e usuários*. Tese de Doutorado, COPE-UFRJ, 1994.
- FREIRE, Fernanda Maria Pereira. *Avaliação lingüística-cognitiva baseada no uso do LOGO: um caso de paralisia cerebral*. (IN) VALENTE, José Armando (org.) *Aprendendo para a vida: Os computadores na sala de aula*. São Paulo: Cortez, 2001.
- GOODYER, Peter. *LOGO- Introdução ao poder do ensino através da programação*. Rio de Janeiro:Campus, 1986.
- LE SANN, Janine. *Elaboration d'un Matériel Pédagogique pour l'apprentissage de notions Géographiques de base, dans les classes primaires, au Brésil:une proposition à partir des apports théoriques de la Géographie, de la Pédagogie, de la Psychologie et de la Graphique*. Thèse de Doctorat, EHESS, Paris, 1989.
- OLIVEIRA, Celina Couto. *Ambientes informatizados de aprendizagem-Produção e avaliação de software educativo*. Campinas: Papirus, 2001.
- PAPERT, Seymour. *LOGO: Computadores e educação*. São Paulo: Brasiliense, 1985.
- PFUHL, Dulce Madalena Autran Von & TUCCI, Wilson José. *Logo-programação & aprendizado*. São Paulo: Nobel, 1985.
- PIAGET, Jean e INHELDER, Barbel. *A representação do espaço na criança*. Porto Alegre: Artes Médicas, 5ª ed. 1993.
- SISTO, Firmino Fernandes. *A construção do espaço cognitivo em Jean Piaget*. (IN)
- MIGUEL, Antônio e ZAINBONE, Ernest. *Representações do espaço: multidisciplinaridade na educação*. Campinas: Autores Associados, 1996.
- VALENTE, Ann Berger e VALENTE, José Armando. *LOGO: conceitos, aplicações e projetos*. São Paulo: MacGraw-Hill, 1988.

¹ Os recursos computacionais podem ser definidos como Softwares dos tipos: tutoriais, simulação, abertos, gráficos, programação e autoria.

² Logo é uma linguagem de programação que foi desenvolvida no Massachusetts Institute of Technology (MIT), Boston, Estados Unidos, por um grupo de pesquisadores liderados pelo professor Seymour Papert, como linguagem de programação serve para estabelecer um processo de comunicação com o computador.

³ Espaço é um conceito que é difícil definir. A natureza de espaço tem sido debatida muito tempo pelos psicólogos, filósofos e os geógrafos usam o conceito em contextos diferentes: 'espaço de um quarto', 'espaço por construir', 'espaço entre cidades' e "espaço territorial", ilustrando usos diversos da palavra. Espaço é uma idéia na mente que permite a estruturação de relações entre objetos. Espaço é subjetivo e relativo, dependendo do modo como é estruturado pela mente em uma ocasião particular. (...) Existem variações na idéia de espaço segurada por pessoas diferentes em tempos diferentes. Há variações consideráveis na idéia de espaço segurada por crianças em fases sucessivas de desenvolvimento intelectual.

⁴ Interativo diz respeito aos sistemas de programas e procedimentos onde o sujeito interfere continuamente no curso das atividades no computador, fornecendo dados ou comandos e o computador, por sua vez oferece uma resposta a esses comandos. O sujeito então interfere no resultado final do processo, personalizando ambientes e modificando-os a qualquer momento.

⁵ A Transdisciplinaridade na visão de Edgar Morin não nega a interdisciplinaridade. A Transdisciplinaridade, como a própria palavra indica na sua etimologia, carrega o prefixo *trans*, que quer dizer, aquilo que atravessa. Nesse sentido, a Transdisciplinaridade na Educação implica um movimento pedagógico por parte dos professores e alunos compreendendo a abordagem de temas que ao atravessarem o campo das disciplinas resgata o sentido do conhecimento, até então, fragmentado, disjuntivo, descontextualizado e não pertinente. Ao criar conexões entre saberes a Transdisciplinaridade como unidade orgânica entre esferas do

conhecimento nos ajuda na cimentação processual através, por exemplo, do trabalho pedagógico por projetos e com projetos em que o conhecimento do aluno é construído através da investigação, da reflexão, da compreensão e explicação no âmbito dos grupos e do coletivo. Assim, os conteúdos e os processos cognitivos, que antes dirigiam a aprendizagem, agora são requisitados através do desenvolvimento dos temas por projetos. O tema transversal é que provoca o movimento transdisciplinar no processo pedagógico. SCALCO, Gildo e UDE, Walter. Transdisciplinaridade e complexidade. Revista Presença Pedagógica. Belo Horizonte: Editora Dimensão, Vol.9 nº 52, Jul/Agos, 2003.

⁶ Software educativo.

⁷ Os softwares chamados programas de comunicação tem ganhado cada vez mais difusão tanto no setor educacional quanto administrativo. São recursos que permitem comunicação à distância em rede como a internet e a intranet, transmitindo informações, mensagens, videoconferências, realizando pesquisas, etc. Problemas relacionados à ilegalidade, privacidade e veracidade da informação transmitida merecem uma revisão desse recurso computacional por parte dos usuários.

⁸ Quanto mais próximo do índice 1, maior é a capacidade do software em desenvolver a habilidade de orientação espacial analisada.