

Proposta de uma Metodologia para Apoiar os Processos de Ensino e de Aprendizagem de Lógica de Programação na Modalidade de Educação a Distância

Naidú Gasparetto de Souza¹, Sidnei Renato Silveira ², Fábio José Parreira³

¹Curso de Bacharelado em Sistemas de Informação – ²Departamento de Tecnologia da Informação (DTecInf) - Universidade Federal de Santa Maria (UFSM) – Campus Frederico Westphalen – RS – Brasil

naidu.gaspar@hotmail.com, sidneirenato.silveira@gmail.com,
fabiojparreira.ufsm@gmail.com

Resumo. Tendo consciência do grande desafio que envolve o processo de ensino-aprendizado dos conceitos de Lógica de Programação apresentado nos Cursos de Computação, o presente trabalho propôs desenvolver uma metodologia de ensino para a modalidade EaD, utilizando como intermédio AVA Moodle, e com base no sistema Problem Based Learning (PBL), utilizando como apoio a ferramenta VisuAlg para apresentação do conteúdo e a elaboração dos desafios. Para que a metodologia possa somar de forma concreta no aprendizado de cada aluno, dividimos a mesma em 9 etapas, onde cada etapa o aluno é colocado como foco principal, trabalhando seus potenciais, e desenvolvendo a postura de agente ativo na procura pelo conhecimento. Com ajuda da metodologia aqui proposta, busca-se amenizar a grande vazão dos alunos dessa área de estudo.

Palavras-Chave: Informática na Educação, Educação a Distância, Lógica de Programação

Abstract. *The present work proposes to develop a teaching methodology for the EaD modality, using AVA Moodle as an intermediate, and based on the Problem Based Learning (PBL), using the VisuAlg tool to present the content and the elaboration of the challenges. In order for the methodology to be able to sum up in a concrete way in the learning of each student, we divide it into 9 stages, where each step the student is placed as the main focus, working their potentials, and developing the active agent posture in search of knowledge. With the help of the methodology proposed here, the aim is to ease the flow of students from this area of study.*

Keywords: *Informatics in Education, Distance Education, Programming Logic.*

1.Introdução

É muito visível que a aplicação das Tecnologias da Informação e da Comunicação cresce cada dia mais, possibilitando o uso em todos os setores da sociedade atual. A

Informática faz parte diretamente de nossas vidas e amplia e facilita o acesso a muitas informações. Cada vez mais a sociedade demanda novas ferramentas tecnológicas e mais acesso à informação. Vivemos na sociedade do conhecimento. Todo esse desenvolvimento exige profissionais que consigam suprir esse processo acelerado. Nesse sentido, há uma preocupação na formação de excelentes profissionais na área da Informática, sendo debatida por vários pesquisadores mundialmente e, no Brasil, mais especificamente com o apoio da SBC (Sociedade Brasileira de Computação).

Para que as novas gerações de alunos que se dedicarem à Informática realmente consigam se desenvolver, não basta apresentar apenas uma simples aula expositiva. Faz-se necessário utilizar diferentes estratégias que possam estimular os processos de ensino e de aprendizagem. Para que os alunos consigam construir seu conhecimento é importante estarem totalmente concentrados naquele momento. Isso pode ser feito criando situações onde desafios sejam utilizados como estímulo, ou até mesmo vendo a aplicação direta do estudo teórico para a execução prática, transformando o conhecimento em aplicação e, assim, construindo gradativamente seu conhecimento.

A Informática, quando aplicada à Educação, gera uma soma de forças muito grande ao âmbito educacional, abrindo portas para que professores e alunos desenvolvam e apliquem diferentes métodos de ensino e de aprendizagem. A aplicação da Informática na Educação pode ser realizada de diferentes maneiras, tais como a utilização de ferramentas para apoiar atividades na modalidade de EaD (Educação a Distância), como é o foco deste trabalho.

Na UFSM (Universidade Federal de Santa Maira), para a realização das atividades relacionadas à EaD, o AVA (Ambiente Virtual de Aprendizagem) *Moodle* é utilizado como ambiente virtual, além de ser utilizado como ambiente de apoio às atividades desenvolvidas nas disciplinas presenciais. Destaca-se, também, que diversas instituições utilizam este ambiente como ferramenta de apoio aos professores e alunos. O *Moodle* tem, como principal objetivo, ser uma ferramenta didática *online*.

Nesse contexto, esse trabalho tem, como principal objetivo, o de desenvolver uma metodologia que gere um impacto positivo para os processos de ensino e de aprendizagem de Lógica de Programação, na modalidade de EaD. Para auxiliar no desenvolvimento dessa metodologia será proposta a utilização de ferramentas para apoiar o ensino e a aprendizagem de Lógica de Programação e sua aplicação será feita baseada no emprego do AVA *Moodle*.

Os processos de ensino e de aprendizagem de computação não são de conhecimento geral da população, já que estes conteúdos não são estudados nas Escolas de Educação Básica e Fundamental. Quando alunos que têm interesses na informática se deparam com a Lógica de Programação chega-se a um momento crítico, gerando um alto índice de desistências. A programação lógica exige um esforço real, o nível de dificuldade empregada é alto, e é um dos requisitos fundamentais na grande maioria dos cursos de computação (PEREIRA, et al., 2004).

Atualmente existe uma grande diversidade de ferramentas para auxiliar os processos de ensino e de aprendizagem de Lógica de Programação, já que se sabe que as disciplinas de programação são as que aumentam os índices de evasão de cursos de Informática (RODRIGUES, 2010). Ferramentas lúdicas são uma forma de atrair a atenção dos alunos, para que eles aprendam com mais facilidade (MACEDO et al., 2005)

A motivação para o desenvolvimento deste trabalho surgiu a partir do grande desafio que envolve os processos de ensino e de aprendizagem da Lógica de Programação (RODRIGUES 2002; SCHULTZ 2003; CHAVES DE CASTRO et al., 2003; DELGADO et al., 2004). Dentro desse contexto, observou-se a possibilidade de propor uma metodologia para apoiar esses processos, acreditando-se que, com paciência, concentração e dedicação, até mesmo a maior das dificuldades pode facilmente ser superada.

Para dar conta desta proposta, este artigo está estruturado da seguinte forma: a seção 2 apresenta um Referencial Teórico, conceituando os principais temas relacionados ao trabalho. A seção 3 apresenta alguns trabalhos relacionados, visando compor o estado da arte. A seção 4 apresenta o estudo de caso desenvolvido com os detalhes da metodologia proposta e ferramentas sugeridas. E, para encerrar o artigo são apresentadas as considerações finais e as referências empregadas.

2. Referencial Teórico

Esta seção apresenta o referencial teórico sobre as áreas envolvidas neste artigo, sendo elas: Informática aplicada à Educação, Educação a Distância, ensino de Lógica de Programação e ferramentas para apoiar os processos de ensino e de aprendizagem de Lógica de Programação.

2.1 Informática aplicada à Educação

O conceito de aprendizado vem de promover e compreender a importância do saber, é estimular a evolução do pensar, dentro das escolas o processo de aquisição do conhecimento é individual, cada um traça sua trajetória e busca percorrer sua própria história de conhecimento. A informática, quando aplicada à Educação, gera uma soma de forças muito grande ao âmbito educacional, abrindo portas para que professores e alunos desenvolvam e apliquem diferentes métodos de ensino e de aprendizagem (NÖVOA, 2009).

Conforme a Organização para a Cooperação e Desenvolvimento Econômico (OCDE), o processo educacional visando apenas a formação de profissionais prejudica o processo de aquisição de conhecimentos gerais que são necessárias para fortalecer o gosto pelo aprendizado, para que o processo de aprendizagem seja mais prazeroso, transformando em hábito para sua vida a busca pela evolução intelectual (OCDE 1995 apud HERNÁNDEZ, 1998).

Todo ato de ensinar envolve o ato de aprender, quando ensinamos estamos reafirmando aquele ensinamento a nós mesmos, e para que se potencialize a intenção é necessária uma total entrega do professor e do aluno (LUCKESI, 2000)

Os recursos oferecidos pela informática permitem renovar os conceitos de ensino e de aprendizado onde, anteriormente, professores agiam como agentes ativos e alunos passivos. Essa prática acaba por limitar a construção de conhecimento por parte do aluno, gerando um desequilíbrio em seus potenciais por estar agindo apenas como ouvinte, sem práticas de sujeito ativo durante o processo (MORAIS, 2003).

A aplicação de recursos da Informática pode fornecer o apoio necessário para as atividades dos professores. É necessário somente que o professor analise quais são as melhores ferramentas, qual o momento correto de aplicar, o contexto a usar, os conteúdos que virão a ser estudados e que objetivo deseja ter através disso. A Informática abre uma infinidade de possibilidades. As atividades mentais estão todas situadas em ambientes culturais, alguns de forma mais capacitador que a outra, que acabam sendo apoiadas pelo professor (BRUNER, 2001).

A computação gera uma motivação da parte do professor quando utilizada para auxílio ao aprendizado, proporcionando um ensino mais dinâmico aos alunos. É muito importante ressaltar que, para a utilização de computadores durante os processos de ensino e de aprendizagem, é necessário haver as especificações pedagógicas e curriculares sobre o uso de computadores na determinada disciplina, para que a aplicação dessa ferramenta não gere frustrações ao invés de benefícios (SILVA, et al., 2013).

Os recursos de Informática abrem a possibilidade de todos os indivíduos buscarem por conhecimento de forma simples e rápida. A Internet vem sendo o principal meio de acesso ao conhecimento, com capacidade de atingir a todos. O ser humano possui essa eterna busca pela evolução, e é necessário que o desejo de aprender e ensinar seja despertado ainda mais na humanidade, que nossas crianças e jovens possam ver e sentir que conhecimento só é conhecimento quando pode ser transmitido, e que não há velho suficiente que não possa aprender nem jovem que não tenha nada a ensinar, tudo é apenas uma questão de percepção.

2.2 Educação a Distância

A Educação a Distância, atualmente, envolve os processos de ensino e de aprendizagem intermediados pelo uso de tecnologias, proporcionando a conexão entre professores e alunos que estão separados fisicamente. A conectividade é feita por diversas tecnologias e ferramentas, tais como os AVAs. A Internet, atualmente, vem a ser o principal meio de conexão. Podemos citar diversos outros meios como o correio, rádio, vídeo e televisão. Existem estilos diferentes de educação, como presencial, semipresencial (parte das aulas são presenciais e a outra parte distância) e a educação a distância ou ainda chamada de educação virtual (MORAN, 1994).

O Instituto Nacional de Estudos e Pesquisas Educacionais Anísio Teixeira (Inep) divulgou dados do Censo de Educação Superior de 2016. De acordo com o levantamento, o número de matrículas nos cursos na modalidade de EaD cresceu 20% de 2015 para 2016. Já no ensino superior presencial, houve uma queda de 3,7% no número de alunos ingressantes (INEP, 2017).

Deve-se considerar que a EaD pode contar com aulas presenciais, mas é basicamente composta por alunos e professores separados fisicamente, seja no espaço ou no tempo, porém, o contato entre professor e aluno pode ser feito virtualmente e em tempo real. Infelizmente ainda existe uma certa ignorância de uma determinada parte da população em relação à modalidade de EaD, pois muitos consideram os cursos a distância com qualidade inferior aos cursos presenciais. Em um passado não tão

próximo, nas origens da EaD, rótulos de má qualidade ganharam grandes dimensões (LEMGRUBER, 2008).

Critérios como a ausência de um professor presencial e material pedagógico limitado, por exemplo, acabavam por condenar a EaD. Ainda pode-se acrescentar a esses critérios de condenação que foram exercidos, uma grande diversidade de cursos EaD que foram disseminados possuindo uma baixa qualidade de conteúdo em geral. Uma série de empresas que acabava usufruindo negativamente da possibilidade de ensinar a distância para aumentar a lucratividade. Do outro lado, a EaD abriu um leque infinito de possibilidades de como e onde se adquirir o conhecimento, não mais estando preso apenas aos ensinamentos dados em sala de aula, assim, podendo o aluno buscar de diferentes formas, em diversas fontes, no horário e local que se sentir melhor para realizar o ato de aprender.

Pelas palavras de Moran (1994), a educação continuada é um importante conceito, que se dá na busca contínua pelo conhecimento e quando unimos a teoria e a prática, gera-se uma autorreflexão, ampliando sempre suas informações e relações. A EaD é aplicada aos mesmos níveis de ensino (ensino, médio, graduação e pós-graduação) e pode estimular a educação continuada. Todas as pessoas podem buscar pela EaD, é interessante que os interessados tenham alguma experiência consolidada de aprendizagem individual e de pesquisa, para que o contato de aprender virtualmente venha a ser mais fácil.

Segundo Belloni (2006), as funções dos docentes, na modalidade de EaD, podem ser divididas em grupos: primeiro os professores que recebem a responsabilidade pelo desenvolvimento dos cursos (conteudistas); segundo os que asseguram o planejamento e organização da distribuição de materiais e da administração acadêmica (formadores); e o terceiro grupo que fica responsável pelo acompanhamento do estudante durante os processos, agindo como tutores.

A EaD possui princípios que podem ser colocados como guias, sendo eles: flexibilidade, para professores e alunos durante os processos de ensino e aprendizagem; Contextualização, buscar sempre a agilidade para suprir as demandas educacionais; diversificação, possibilitando que o conhecimento possa ser apresentado e assimilado de diferentes formas, para estilos cognitivos variados; e abertura, permitindo que o aluno crie seus próprios hábitos de estudos a partir de seu tempo e suas necessidades (LEITE et al., 1998).

2.3 AVA Moodle

O *Modular Object Oriented Distance Learning (Moodle)* foi desenvolvido com o objetivo de ser um Sistema para Gerenciar Cursos (SGC), com a finalidade de ajudar os profissionais da área da Educação a elaborarem cursos *online*. Desenvolvido a partir de 1990, na Austrália, o *Moodle* tem como líder Martin Dougiamas. Os sistemas educacionais *online* são denominados, por sua vez, de Sistemas de Gerenciamento de Aprendizagem (SGA) ou Ambientes Virtuais de Aprendizagem (AVA). O *Moodle* é uma das principais plataformas utilizadas como Ambientes Virtual de Aprendizagem do mundo. Um dos fatores mais relevantes sobre o *Moodle* é seu código-fonte aberto (*Open Source Software*), permitindo que qualquer usuário possa utilizá-lo gratuitamente, por meio do *download* de uma de suas versões disponíveis na página *web*, podendo ser

usado, sem modificações, em diferentes Sistemas Operacionais, tais como *Unix*, *Linux*, *Windows*, *Mac OS* e outros sistemas de suportem PHP (linguagem de programação utilizada no desenvolvimento do AVA *Moodle*) (PULINO FILHO, 2004)

É possível modificá-lo, devido às condições da licença *GNU (General Public License)*¹. Devido à qualidade dos serviços oferecidos pelo AVA *Moodle*, um grande número de usuários vem se unindo a esse sistema, contribuindo com suas ideias e implementações, a partir de atividades tais como: correção de eventuais problemas ou *bugs*, modificação de seu código-fonte, acrescentando e compartilhando módulos específicos, experimentando novas perspectivas pedagógicas, ou simplesmente divulgando suas potencialidades (PULINO FILHO, 2004).

O AVA *Moodle* é utilizado, principalmente, no contexto de *e-learning*² ou *b-learning*³, onde o sistema permite a criação e desenvolvimento de cursos *online*, assim como páginas de disciplinas, grupos de trabalho e comunidades de aprendizagem. O *Moodle* está espalhado por todas as partes do mundo, chegando a ser utilizado em mais de 175 países (FREITAS, 2013).

2.4 Ensino de Lógica de Programação

Dentro da Lógica de Programação existe uma série de problemas que podem ser questionados, tanto destinados a professores quanto a alunos, é difícil atribuir a uma só causa, até mesmo as próprias metodologias que são aplicadas podem ser questionáveis, fica demasiadamente difícil apontar os fatores determinantes que dificultam o processo de aprendizagem da Lógica de Programação, por isso, cabe à os interessados o empenho em desvendar os mistérios relacionados a esse ensino, e com isso elaborar meios que facilitem e auxiliem o aprendiz.

Os processos de ensino e de aprendizagem de Lógica de Programação não são de conhecimento geral da população, pelo contrário, exigem conhecimentos específicos relacionados à linguagem de programação e à lógica necessária para desenvolver os programas e outros conhecimentos básicos relacionados à informática. Como a Lógica de Programação exige esses conhecimentos básicos para o seu aprendizado, nos cursos de Computação o que acontece é um grande número de desistentes devido às dificuldades encontradas, gerando reprovações e até mesmo energias negativas (baixa estima, ansiedade) nos alunos. Esse é um dos problemas que desafiam a Educação em Informática (FERREIRA et al., 2010)

Nos atuais currículos das escolas brasileiras ainda não existem conteúdos de computação, porém em alguns locais do Brasil essa prática já é aplicada, como as escolas de programação SuperGeek e MadCod, sediadas em São Paulo, e programas como Robótica na Escola, promovido no Estado do Recife (ARAÚJO et al., 2015).

¹ GNU *General Public License* (Licença Pública Geral), GNU GPL ou simplesmente GPL, é a designação da licença para *software* livre idealizada por Richard Matthew Stallman em 1989, no âmbito do projeto GNU da *Free Software Foundation* (FSF) (STALLMAN, 1989).

² *e-learning* (do inglês *electronic learning*, "aprendizagem eletrônica") (LEAL; AMARAL, 2004).

³ *Blended learning* refere-se a um sistema de formação onde a maior parte dos conteúdos é transmitido na modalidade a distância (FRIESEN, 2012).

Araújo et al., (2015) ainda falam sobre o equívoco ocorrido ao dizer que aulas de informática do Ensino Superior podem ser compartilhadas da mesma forma para crianças de séries iniciais. Para crianças, o ensino da computação na Educação Básica deve ter o objetivo de agregar conhecimento de raciocínio lógico-matemático, sendo diferente da abordagem do ensino superior.

Existem uma demanda maior que a oferta de profissionais de computação, porém a Educação Básica na sociedade brasileira ainda carece de ensino de computação. É de grande viabilidade a implementação do ensino de programação lógica para criar bons profissionais, observando que uma vasta parte das atividades econômicas envolvem atividades computacionais (BEZERRA; DIAS, 2014).

Grande parte dos alunos acaba passando por dificuldades, não conseguindo desenvolver o raciocínio lógico que é necessário para prosseguir no aprendizado de programação. A dificuldade encontrada pelos alunos pode chegar a tal nível que comece a afetá-los mentalmente e emocionalmente, o que acaba gerando reprovações e, quando pior, a evasão dos cursos (RODRIGUES, 2002).

A grande pauta é como gerar o interesse pela programação (motivação), pelos alunos, fazendo com que esse processo flua sem dificuldades, ajudando a superar qualquer dificuldade encontrada que o impeça de prosseguir Friedrich (FRIEDRICH, et al., 2012)

Na maioria das vezes, os conteúdos são apresentados de forma expositiva aos alunos, sendo esse o modo mais tradicional, a concentração é feita na busca pela solução do problema estabelecido pelo professor. O *Portugol* (pseudocódigo) é utilizado como ferramenta de apoio pelos professores, que podem aplicar exercícios ou demonstrar exemplos aos alunos (CASTRO, 2003).

Pelas palavras de BORGES (2000), atrair os alunos de uma forma a se interessar inteiramente pelas disciplinas é uma tarefa difícil, o modo tradicional não vem conseguindo realizar essa atração, o aluno acaba não tendo total clareza sobre a importância que as disciplinas de programação têm para a formação acadêmica. Acredita-se que o contato professor-aluno é uma parte importante da motivação. Ambientes mais descontraídos, com maior cooperação entre os participantes são interessantes ao gosto dos alunos do que ambientes mais fechados que visam mais a transmissão da informação, em uma visão instrucionista.

As metodologias realizam o apoio no processo de aprendizado do aluno, porém, não há uma metodologia que consiga atingir de forma concreta todos eles, até a mais famosa e consolidada metodologia não conseguem atingir a todos os alunos (SANTOS, 2003).

A metodologia instrucionista consiste em implementar no computador uma série de informações, e através dessas informações se é passado aos alunos na forma de um tutorial, simulação, exercício-e-prática ou jogo. É o paradigma que transfere para o computador a tarefa de ensinar, ou reforça as atividades realizadas em sala de aula. O ensino instrucionista é meio mais utilizado ainda nas escolas (VALENTE, 2005).

Autores como Marion (1999) dizem que paradigmas voltados a orientação a objeto, quando direcionados a disciplinas iniciais, acabam por não conseguindo atingir de maneira intensa os alunos, não trazendo uma ajuda significativa para os processos de ensino e de aprendizagem. Entretanto, alguns autores como Chaves (2003), afirmam que

paradigmas funcionais ajudam no processo de aprendizado. Outros autores defendem outros estilos de paradigmas, como o orientado a objeto ou procedural clássico (DELGADO et al., 2004).

Observando os desafios encontrados durante os processos de ensino e de aprendizagem de Lógica de Programação, é notável que a academia realiza uma mobilização em busca de uma maneira eficiente e confortável para conseguir atingir o estudante de forma positiva, amenizando as dificuldades e motivando-o (FERREIRA, et al 2010; PEREIRA, et al 2004; SCHULTZ 2003). A união de esforços, através de fatores para melhorar esse processo, seja através de metodologias pedagógicas adequadas, informações consistentes, ambientes adequados, ferramentas de apoio ou professores competentes, nunca serão o suficiente se os alunos não buscarem apresentar um interesse real em aprender, acredito que se torna extremamente mais difícil ensinar a quem não tem um real interesse em aprender. Os cursos devem alertar e reafirmar as dificuldades encontradas nas disciplinas de programação para que alunos inscritos em cursos de computação não se deparem com um “pesadelo” diante dessas disciplinas, que é o que acaba acontecendo, alunos que tem interesse ou afinidade com computadores, jogos e Internet em geral, e buscam cursos da área de Computação sem consciência da grade curricular que os compõem. É importante que exista consciência dos alunos nessa parte e clareza sobre os desafios encontrados durante o processo do conhecimento algorítmico (PEREIRA et al., 2004).

2.5 Ferramentas para ensinar Lógica de Programação

Muitos autores apontam que o uso de ferramentas lúdicas é uma forma de atrair a atenção dos alunos, para que eles aprendam com mais facilidade e consigam desenvolver mais facilmente suas habilidades. Para Macedo et al., (2005) a utilização de ferramentas lúdicas para apoiar os processos de ensino e de aprendizagem possuem as seguintes qualidades: tornam as tarefas prazerosas, são desafiadoras, possuem dimensão simbólica e não limitam as possibilidades.

Existem inúmeras ferramentas para apoiar os processos de ensino e de aprendizagem de Lógica de Programação, tais como o *Ambap*. O *Ambap* (Ambiente de Aprendizado de Programação) foi desenvolvido com o objetivo de auxiliar alunos iniciantes durante os processos de ensino e de aprendizagem de programação. Entre seus objetivos estão o conjunto de ferramentas computacionais que auxiliam o ensino de programação, atuando diretamente no auxílio à resolução de problemas, trabalhando diretamente na compreensão do problema e a construção da forma algorítmica necessária para solucioná-la. O *Ambap* permite que os usuários desenvolvam seus programas por meio de linguagem algorítmica, e ainda possibilita o entendimento de conceitos relacionados a variáveis, comandos, recursão, através de um processo de simulação, deixando de lado as preocupações inerentes relacionadas à implementação com características específicas da máquina, o que normalmente é encontrado em linguagens de programação convencionais (ALMEIDA; COSTA; SILVA, 2002). Podemos observar a interface simples e dedutiva que o *Ambap* possui na Figura 1.

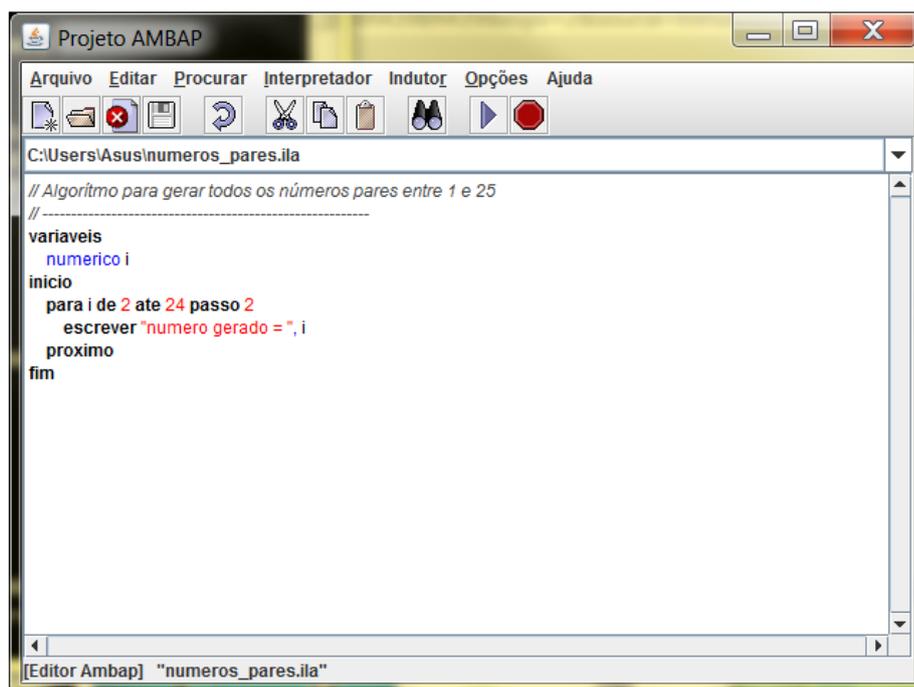


Figura 1. Interface inicial Ambap (ALMEIDA; COSTA; SILVA, 2002).

Outra ferramenta é o *VisuAlg*. Esta é uma ferramenta que permite o ensino de lógica de programação a alunos iniciantes, por meio de um ambiente próximo à realidade. Alunos iniciantes passam por grandes dificuldades de abstração, ou seja, executar um programa somente por escrita em papel, sendo uma das barreiras necessárias a se quebrar no aprendizado das técnicas de elaboração de algoritmos. Vendo por outro ângulo, alunos iniciantes podem encontrar grandes dificuldades utilizando linguagens de programação como Pascal ou C, por exemplo, devido a sua linguagem complexa. Algo próximo ao ideal seria iniciar os estudantes em uma linguagem mais simples, de fácil entendimento, como o “Portugol”, que possui reconhecimento acadêmico, em diversos livros, possuindo ainda os conceitos básicos da estrutura de programação, dessa forma, fazendo com que a jornada de aprendizagem da programação se torne mais prazerosa (RODRIGUES, 2010).

Com uma metodologia simples, o *VisuAlg* é uma ferramenta capaz de simular diversos comandos na tela do computador, possibilitar verificações de valores e variáveis, e todo o acompanhamento da execução do algoritmo passo a passo, devido ao seu grande apelo didático. Ainda é capaz de executar depurações simples. Possui ainda um editor de texto simples e muito eficiente e, por fim, dispõe de todos os principais recursos necessários para trabalhar em um ambiente gráfico (RODRIGUES, 2010).

O *VisuAlg* é uma das ferramentas que são apresentadas aos alunos do curso de Sistemas de Informação na UFSM-FW para apoiar os processos de ensino e de aprendizagem dos alunos. A tela do *VisuAlg* compõe-se da barra de tarefas, do editor de textos (que toma toda a sua metade superior), do quadro de variáveis (no lado esquerdo da metade inferior), do simulador de saída (no correspondente lado direito) e da barra de *status*, como mostra a Figura 2. Quando o programa é carregado, já apresenta no editor um "esqueleto" de pseudocódigo, com a intenção de poupar trabalho ao usuário e de mostrar o formato básico que deve ser seguido.

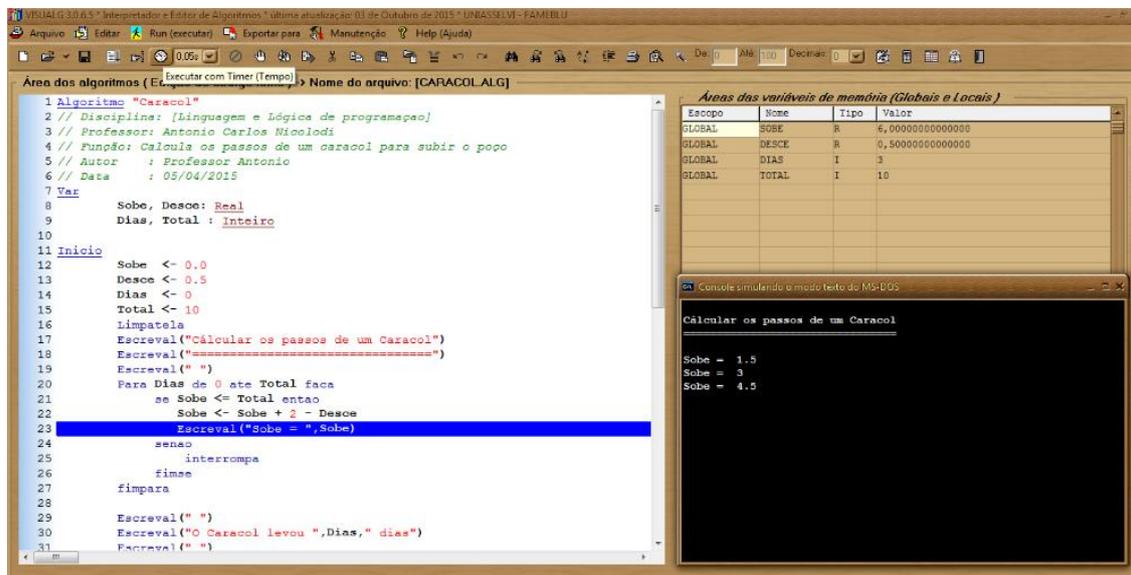


Figura 2. Interface inicial da ferramenta *VisuAlg* (Fonte: visualg3.com.br, 2017).

A ferramenta *Scratch* foi desenvolvida em 2012 pelo *Media Lab* do Instituto de Tecnologia de *Massachusetts*. Em seus princípios básicos buscou inspiração nas linguagens *Logo* e *Squeak*, porém, seu objetivo é ser uma linguagem de maior simplicidade, utilização fácil e intuitiva (SCRATCH, 2017).

É um *software* distribuído gratuitamente, contém uma IDE (*Integrated Development Environment*) que descarta a digitação de funções, endereços, etc. O objetivo do *Scratch* é estimular os processos de ensino e de aprendizagem de maneira fácil, utilizando conceitos matemáticos e de computação, de forma que o desenvolvimento individual da parte lógica de cada um dos alunos seja estimulado.

O termo *Scratch* vem da técnica de *scratching* (arranhar), que é utilizada pelos *Disco-Jockeys* (DJs) no *Hip Hop* e músicas eletrônicas que, por meio do movimento de discos de vinil com as mãos para frente e para trás cria, assim, uma sinfonia misturando o som já gravado com a forma criativa e espontânea de suas artes. O *Scratch* trabalha de forma semelhante, permitindo que o usuário misture diversos tipos de trechos de mídia (fotos, músicas, gráficos, sons) conforme a criatividade individual (SCRATCH, 2017).

A interface do *Scratch* é dividida da seguinte maneira: 1. Os Blocos são os elementos para construção dos programas. São usados para organizar os comandos a serem executados. Portanto, cada bloco possui uma função relacionada. 2. Na parte superior da área central encontram-se três abas: *scripts*, *costumes* e *sounds*. Em cada uma existem os blocos de funções correspondentes à cada categoria. Cada bloco pode ser testado separadamente bastando, para isso, clicar duas vezes sobre o objeto. 3. Na coluna mais à direita encontra-se a área de comunicação dos comandos (Tela). Basta arrastar o comando até esta área para que ele seja incorporado ao código. A Figura 3 apresenta a interface detalhada da ferramenta *Scratch*.

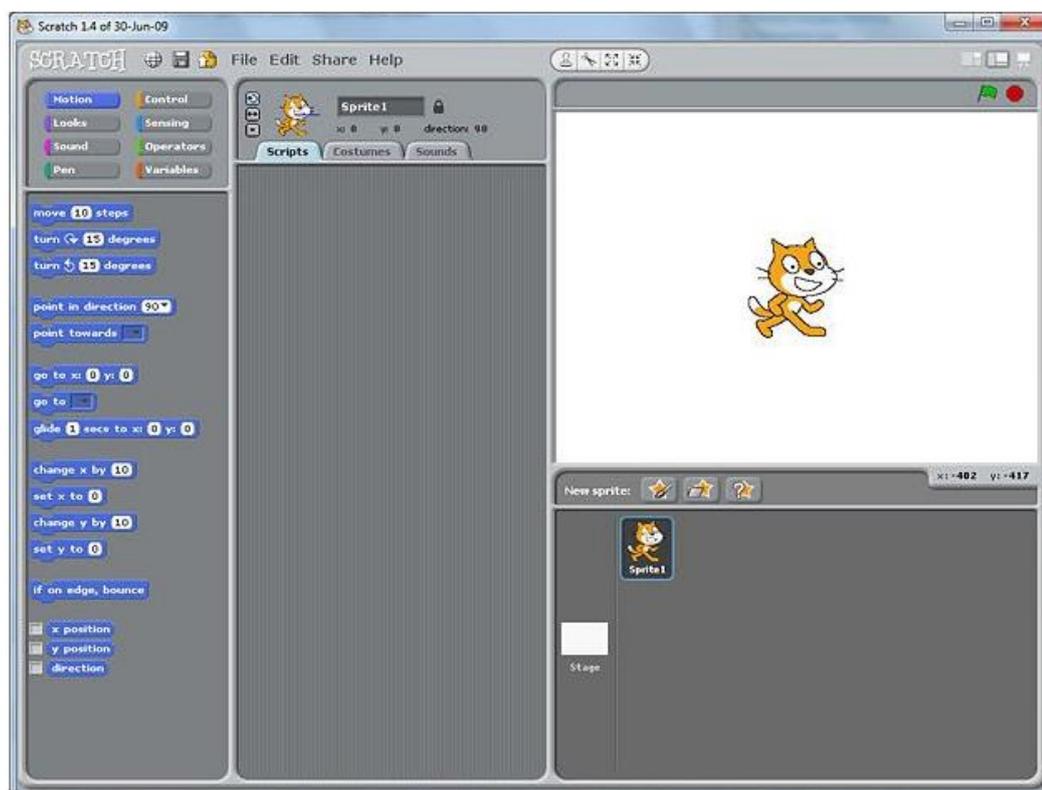


Figura 3. Interface inicial da ferramenta Scratch (Fonte: scratch.mit.edu, 2017).

A ferramenta Alice tem a capacidade de fornecer animações 3D e manipular de maneira direta a linguagem de programação e seus elementos. No sistema Alice o aluno não tem a necessidade de escrever o código de programação, assim removendo a necessidade de o aluno desenvolver a sintaxe do programa, permitindo que eles foquem sua atenção somente na parte que condiz ao entendimento dos conceitos de aprendizagem. A ferramenta possui uma linguagem gráfica, que introduz conceitos de animações para ajudar a desenvolver o aprendizado e construir programas (COOPER et al., 2003).

Com a utilização do *software* Alice é possível criar mundos virtuais 3D. Esses mundos são criados com objetos gráficos que mudam seus comportamentos com o passar do tempo. Os objetos na ferramenta Alice podem ser trabalhos de diversas formas, girar, mover, cortar, colorir, andar e correr por exemplo. Alice se enquadra no estilo de ferramenta CAD (*Computer Aided Design* ou Projeto auxiliado por Computador) que são ferramentas que trabalham com modelagem de objetos 3D. A ferramenta possui um banco de dados com diversos grupos de objetos estabelecidos pelos próprios desenvolvedores de Alice. Também é possível adicionar objetos de terceiros ou da própria Internet, a qual oferece uma infinidade de objetos modelados. A Figura 4 apresenta a interface da ferramenta Alice.

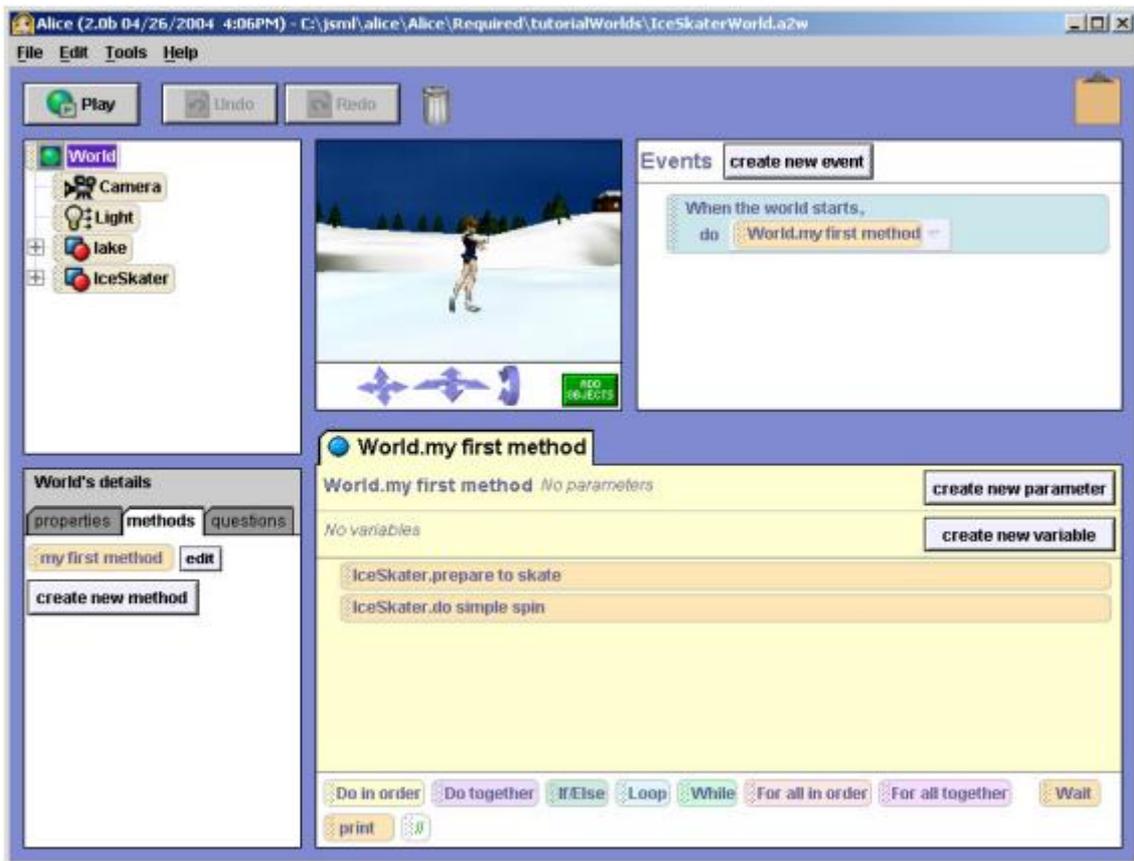


Figura 4. Interface inicial ferramenta Alice (Fonte: www.alice.org, 2017).

A última ferramenta aqui utilizada como referência é a *KTurtle*, que foi desenvolvida para dar assistência ao ensino inicial da programação da linguagem “Logo”. Foi desenvolvida para aplicação em escolas, com intuito de atingir a educação básica, sendo uma linguagem simples de aprender, o que acaba facilitando esse contato inicial com a programação (SOUSA et al., 2015).

O *KTurtle* é baseado na linguagem “Logo”, sendo uma das únicas linguagens de programação que faz o papel inverso, onde os comandos da programação são traduzidos para o idioma do programador, transformando descrições em código. Foi desenvolvida pela *KDE Software Compilation* e lançada sob a Licença Pública Geral GNU.

Quando a ferramenta foi lançada, foi disponibilizada para diversas plataformas de Linux, tais como *Red Hat*, *openSUSE*, *Debian* e *Ubuntu*. O *KTurtle* possui pacotes com separações, módulos simples e outros mais completos, sendo o *Kdeedu*, o qual já vem com o *KTurtle* incluso. Também existem versões da ferramenta disponibilizada para o Sistema Operacional *Windows*.

Quando executado, o *KTurtle* tem o objetivo de ensinar programação utilizando apenas gráficos de tartaruga e não possuindo maiores recursos, já que é voltado para o ensino inicial. A Figura 5 apresenta a interface da ferramenta *KTurtle*.

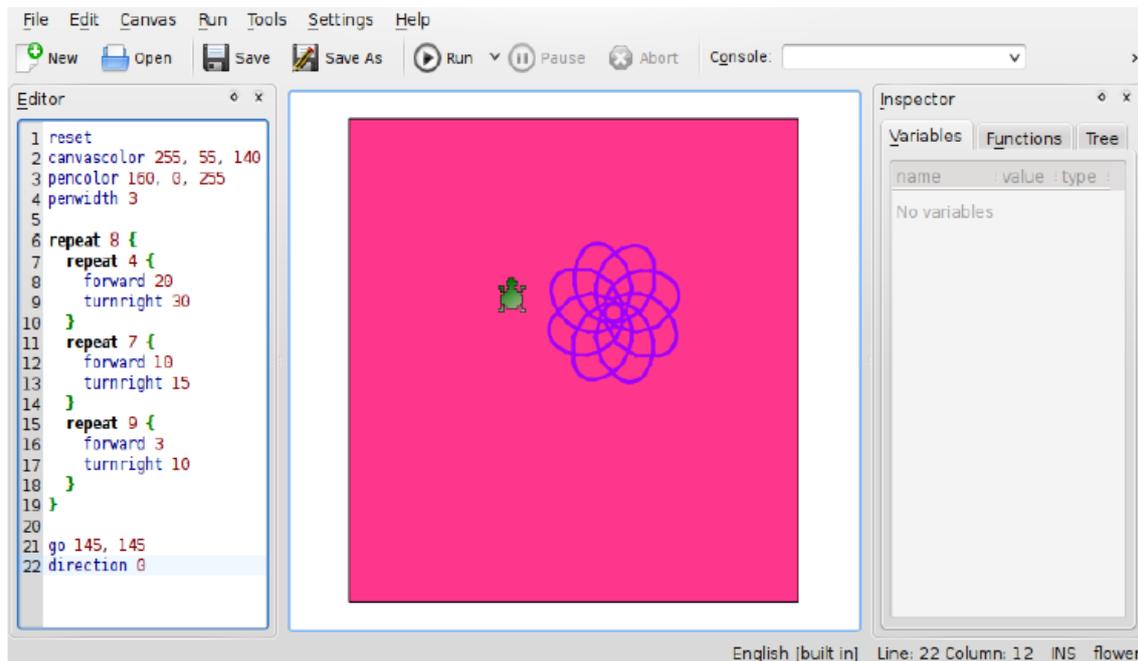


Figura 5. Interface inicial da ferramenta *KTurtle* (Fonte: edu.kde.org/KTurtle, 2017).

3. Estado da Arte

Esta seção apresenta alguns trabalhos relacionados e que foram utilizados como base para a elaboração da metodologia proposta. Ao final da seção apresenta-se um estudo comparativo entre os trabalhos estudados e o trabalho aqui apresentado.

3.1 Uma Proposta para o Ensino de Programação de Computadores na Educação Básica

No trabalho desenvolvido por Garlet, Bigolin e Silveira (2016) foi proposta a definição de um modelo que permita a implantação do ensino de Lógica de Programação na Educação Básica Brasileira, especialmente nos últimos três anos do Ensino Fundamental (7º, 8º e 9º anos).

Foi realizado um estudo de caso, que seguiu um roteiro de observações, onde foi analisada a motivação dos alunos em relação ao conteúdo ministrado em aula, a capacidade de aprendizagem, a diferença no comportamento de ambos os sexos e também a participação individual de cada aluno que esteve presente projeto. Foram realizadas aulas de introdução à Lógica de Programação para alunos do 7º ao 9º ano do Ensino Fundamental da Escola Estadual de Ensino Médio Cardeal Roncalli de Frederico Westphalen – RS e da Escola Estadual de Ensino Médio São Gabriel de Ametista do Sul - RS.

O trabalho teve, como objetivo geral, o de comprovar a importância do ensino de Lógica de Programação na educação básica em nosso país, além de mostrar os

benefícios para os alunos que buscam pelo Ensino Superior, fazendo com que os profissionais sejam cada vez mais qualificados para o mercado de trabalho.

A ferramenta utilizada durante as aulas foi o *VisuAlg*, que tinha o intuito de ensinar Lógica de Programação. Com o uso da ferramenta os conheceram os comandos principais de programação, tais como comandos de entrada, saída, atribuição, desvio condicional e de seleção múltipla. Os autores afirmam que o *VisuAlg* foi muito útil para os alunos, e que teve uma grande aceitação por ter uma fácil usabilidade e ser em português. O *VisuAlg* foi a ferramenta escolhida por ser mais direcionada aos processos de ensino e de aprendizagem da parte da programação lógica, que é o objetivo encontrado nesse trabalho.

O contato para o levantamento dos dados e validação do trabalho foi realizado em nove encontros de 2 horas cada no período de tempo de um semestre. As aulas eram ministradas em um laboratório de informática em máquinas equipadas com sistema operacional *Microsoft Windows*.

Analisando os resultados obtidos com o estudo, foi observado que os alunos tiveram, inicialmente, dificuldades na aprendizagem do conceito e uso das variáveis. Porém, passadas estas dificuldades, o desempenho das aulas melhorou e os alunos começaram a desenvolver adequadamente as atividades propostas, permitindo com que se cumprisse com o plano de ensino.

O objetivo principal do trabalho, que era fazer com que os alunos estudassem Lógica de Programação foi atingido, pelo que afirmam os autores, pois os alunos conseguiram desenvolver sozinhos, todas as atividades propostas. A maior dificuldade encontrada foi o grande número de desistentes da parte dos alunos, número que chegou a 85% em uma das escolas e de 48% na outra. Entretanto, essa desistência não impossibilitou a conclusão da validação do estudo.

3.2. Desenvolvendo o Raciocínio Lógico no Ensino Médio: uma proposta utilizando a ferramenta *Scratch*

O trabalho desenvolvido por Mota, Ribeiro e Emmendorfer (2014) propôs a aplicação de um curso de Lógica de Programação por meio da ferramenta *Scratch* para adolescentes do ensino médio em uma escola pública, ainda contando com a participação de alguns professores. A proposta do trabalho consistiu na execução de uma sequência de etapas, que se iniciou com oficinas de Lógica de Programação por meio da ferramenta *Scratch*. Essas oficinas ocorreram na Universidade Federal do Rio Grande (FURG) de forma gradual, onde os alunos desenvolveram os conhecimentos de Lógica de Programação de forma lúdica e criativa e implementaram desafios a partir de um conjunto de condicionais, sem a necessidade do uso da sintaxe exigida em linguagens de programação, tais como Java e C. Assim os alunos tinham somente o objetivo de desenvolver a lógica da atividade proposta, sem se preocupar com os detalhes de implementação.

A oficina teve duração de 24 horas/aula, realizadas no laboratório de informática de uma escola pública, com a frequência de duas vezes na semana. Os alunos atendidos pelas oficinas foram das 7^o e 8^o serie. Essas oficinas foram realizadas nas disciplinas de matemática e física e envolveram problemas resolvidos relacionados ao conteúdo visto

em aula, com o fim de despertar nos alunos o interesse pela solução de problemas de forma não tradicional, e sim por meio de jogos e programas desenvolvidos no *Scratch*.

Para realizar o acompanhamento e monitoramento dos alunos, foram aplicados questionários antes e depois da realização das oficinas, com o intuito de identificar possíveis mudanças de postura, por parte dos alunos, com relação à resistência quanto à Lógica de Programação, uma vez que os alunos julgavam esse conteúdo complicado. Também foi avaliado nos alunos o interesse por ciências exatas e da Terra antes e após as oficinas de *Scratch*.

O estudo em questão foi realizado em um grupo pequeno, foi aplicado a um grupo de 8 alunos que já realizou a oficina, onde todos demonstraram muito interesse em continuar o estudo de Lógica de Programação e da ferramenta *Scratch*. O próximo passo para esse trabalho descrito pelos autores é a aplicação dos estudos em turmas maiores, utilizando uma análise qualitativa e quantitativa, baseada em questionário com questões abertas e fechadas.

3.3. Ensino de Lógica de Programação através do Jogo Defense of the Ancients 2

Em seu trabalho Cataldo et. al., (2017) propuseram um minicurso que pudesse servir como auxílio para alunos do curso de Informática, que estivessem passando por dificuldades no desenvolvimento de algoritmos, visaram incentivá-los a formular novos *scripts* para o jogo chamado *Defense of the Ancients 2* através da programação em Lua.

A linguagem de programação Lua trabalha com *script* de multiparadigma onde a ideia é fornecer um framework no qual o programador possa trabalhar com vários estilos, misturando livremente construtores de diferentes paradigmas, sendo pequena, reflexiva e leve. Foi projetada com o objetivo de expandir aplicações em geral, é uma linguagem extensível (agrupa partes de diferentes linguagens), para prototipagem e para ser embarcada em jogos, que são considerados *softwares* com maior complexidade.

O trabalho foi desenvolvido no curso de Ciência da Computação no Instituto Federal de Educação, Ciência e Tecnologia do Sudeste de Minas Gerais - Campus Rio Pomba-MG.

A metodologia utilizada no desenvolvimento do minicurso foi dividida em três passos: primeiro a iniciação lúdica baseada em resolução de problemas; segundo a solução de problemas com linguagem natural; terceiro a construção de programas e traduções de algoritmos com a linguagem Lua.

Durante o minicurso foi explicado sobre a linguagem Lua, e projetos em que ela foi utilizada. Projetos bem conhecidos, como as ferramentas *CryEngine 3* e *Corona SDK* e jogos bem famosos como *Angry Birds*, *Street Fighter IV*, *Tibia* e *DOTA 2*. O minicurso seguiu a ordem de ensinar variáveis e operadores, e suas respectivas sintaxes da linguagem Lua, e pós isso foi realizado a explicação da semântica.

Depois que receberam as instruções sobre o funcionamento da programação em Lua foi abordada a utilização dos *scripts* direto no jogo DOTA 2.

A avaliação foi realizada por meio de um formulário respondido pelos alunos, que em sua verificação possibilitou identificar a satisfação dos mesmos com relação ao minicurso. O minicurso teve um pequeno público de 7 alunos do curso de Informática

com idade de 16 a 17 anos e onde todos possuíam o conhecimento básico de lógica e programação. Ao todo 85,7% dos alunos não conheciam Lua, e 57,1% já tinha o conhecimento sobre a existência do jogo DOTA 2. Dos 7 alunos que participaram do minicurso, 6 obtiveram uma fixação promissora, onde os estudantes responderam a um questionário que continha 8 questões, com objetivo de saber se o jogo demonstrou ser um ótimo exemplo visual da programação realizada e se acreditaram ser eficiente a união de jogos ao ensino. O estudo foi realizado na matéria de Algoritmo e Programação.

Os autores concluíram que foi possível observar ao longo do trabalho que os indivíduos acabam demonstrando interesse pelo método de ensino da Lógica de Programação utilizando o meio de jogos eletrônicos, onde foi notável a motivação por parte dos alunos, incentivando-os a explorar o conteúdo visto em sala de aula em jogos, podendo assim observar na prática o funcionamento da estrutura de dados em um ambiente onde já possuem familiaridade como um jogo de computador.

Para trabalhos futuros, deixam em aberto uma possível continuação do desenvolvimento de *scripts* para aplicações no DOTA 2. Criando a possibilidade para um próximo minicurso com a ideia de instruir os alunos com conceitos de Inteligência artificial (IA) com a criação de técnicas de Inteligência Artificial para os *BOTs* (Personagens controlados pelo computador).

3.4 Estudo Comparativo

Após a apresentação de trabalhos relacionados ao proposto, traçou-se um comparativo entre os mesmos e a solução proposta, originando um quadro destacando as características mais relevantes de cada um (Quadro 1).

Quadro 1 – Estudo Comparativo

Características	Trabalho 1 (Garlet et al., 2016)	Trabalho 2 (Mota et al., 2014)	Trabalho 3 (Cataldo et, al., 2017)	Solução Proposta
Nível de Ensino	Ensino Médio	Ensino Médio	Ensino Superior	Ensino Superior
Série/Turma/Semestre	7º 8º e 9º anos	7º e 8º	***	1º e 2º
Ambiente de Programação Utilizado	<i>VisuAlg</i>	<i>SCRATCH</i>	<i>Jogo Defense of the Ancients 2</i>	<i>VisuAlg</i>
Ligado a alguma disciplina existente?	Não	Física e Matemática	Introdução a Algoritmos	Introdução a Algoritmos
Objetivos	Relatar a importância do ensino da	Desenvolver a capacidade lógica e de	Auxiliar no processo de aprendizado de	Desenvolver uma metodologia

	Lógica de Programação desde o ensino fundamental.	raciocínio através da Lógica de Programação com auxílio de ferramentas	Lógica de Programação com utilização de jogo eletrônico e programação em Lua.	para ensinar Lógica de Programação para a modalidade de EaD com auxílio de ferramentas de apoio.
Forma de Avaliação/Validação	Estudo de Caso	Questionário	Formulário	***

Com base nas características destacadas no Quadro 1, pode-se observar as diferentes abordagens realizadas pelos autores e as técnicas utilizadas durante o desenvolvimento dos trabalhos, alguns optando por ferramentas simples com linguagem próxima do português estruturado, outros em contexto de programação visual e multimídia e um deles opta por utilizar jogos eletrônicos como ferramenta de apoio.

O trabalho aqui em questão buscou abordar as dificuldades de ensino de Lógica de Programação, as ferramentas utilizadas nos processos de ensino e de aprendizagem, quais metodologias vêm sendo aplicadas, a EaD e o ensino de Lógica de Programação, para que com essas importantes informações nos apoiem no desenvolvimento de uma metodologia de ensino voltada a auxiliar no aprendizado de Lógica de Programação aplicada em EaD, que futuramente será direcionada ao Ambiente Virtual de Aprendizagem (AVA) *Moodle*.

No nosso caso, a ferramenta escolhida para ser utilizada nesse trabalho foi o *VisuAlg* devido a sua objetividade e clareza para quem está iniciando no mundo dos algoritmos. O *VisuAlg* possui uma linguagem simples, é muito prática a elaboração de pequenos sistemas e ainda permite melhor entender a execução por meio de um visualizador de variáveis, assim facilitando os contatos iniciais com a programação e consequentemente fortalecendo o processo de aprendizado de futuras linguagens como C, Java e Delphi por exemplo.

4. Estudo de Caso Proposto

A solução apresentada neste trabalho envolveu a proposta de uma metodologia para apoiar os processos de ensino e de aprendizagem de Lógica de Programação, para aplicação no AVA *Moodle*, que é o ambiente utilizado pela UFSM como apoio às atividades dos cursos presenciais e, também, nos cursos ministrados na modalidade de EaD, por meio da UAB (Universidade Aberta do Brasil).

A ideia principal se baseou na criação de uma metodologia que auxilie os processos de ensino e de aprendizagem de Lógica de Programação para ser aplicada nas disciplinas iniciais de ensino de programação do Curso de Licenciatura em Computação da UFSM.

Além disso, foram estudadas e definidas formas de apresentar os conteúdos de Lógica de Programação na modalidade de EaD, tais como vídeo aulas, tutoriais e/ou simulações.

A metodologia de pesquisa utilizada para o desenvolvimento deste trabalho foi o Estudo de Caso. Pelas palavras de Yin (2001), estudos de caso são indicados em trabalhos que envolvam questões do tipo “por que” e “como”, que são questionamentos que fazem parte da metodologia de pesquisa dos estudos de caso. Como o objetivo principal desse trabalho trata-se em elaborar um método para apoiar os processos de ensino e de aprendizagem de Lógica de Programação Lógica na modalidade EaD, a metodologia do estudo de caso será utilizada como base. Yin (2001) ainda diz que a metodologia dos estudos de caso pode ser utilizada em diversos campos políticos, econômicos, sociais e administrativas.

Tendo-se em vista que esta proposta foi desenvolvida em apenas um semestre, não foi possível realizar a validação da metodologia proposta, podendo ser realizada como trabalhos futuros.

4.1 Definição do Método e da Ferramenta

O método proposto será aplicado com o apoio da ferramenta *VisuAlg* que permite que os alunos tenham contato com a linguagem de programação de maneira simplificada, aprendendo a trabalhar as principais funções que a ferramenta disponibiliza e entendendo comandos de entrada, saída, atribuição, desvio condicional e de seleção múltipla, sempre visando o melhor aprendizado dos alunos.

O *VisuAlg* apresenta uma metodologia simples, sendo uma ferramenta capaz de simular diversos comandos na tela do computador, o que acaba por ajudar nas associações e entendimento lógico do aluno. Outro benefício é a possibilidade de verificar os valores das variáveis, e acompanhar a execução do algoritmo em todos os seus passos. A ferramenta *VisuAlg* visa o apoio didático, dispondo dos principais recursos necessários para se estudar Lógica de Programação.

A escolha do *VisuAlg* foi realizada com base em alguns fatores. Um deles é o fato da ferramenta *VisuAlg* ser uma das ferramentas apresentadas aos alunos dos cursos de Bacharelado em Sistemas de Informação e de Licenciatura em Computação na UFSM-FW para apoiar os processos de ensino e de aprendizagem. Outro fator é a praticidade de uso da ferramenta e seu fácil entendimento e manuseio, que acabam por apoiar de forma concreta alunos iniciantes da programação. É uma ferramenta que tem como foco principal o de auxiliar o processo de aprendizado da Lógica de Programação, que vem a ser principal objetivo de nosso trabalho também.

A principal motivação para utilizar o *VisuAlg* como ferramenta de apoio à metodologia, foi a experiência dos autores deste trabalho. Um dos autores deste trabalho destaca que, logo nos primeiros semestres, ao cursar Sistemas de Informação na UFSM-FW, teve o primeiro contato com as disciplinas de programação (Lógica e Algoritmos, Laboratório de Programação, Estrutura de Dados, entre outras), encontrando dificuldades durante o decorrer das aulas. Neste momento o *VisuAlg* passou a auxiliar em seu processo de aprendizado, permitindo que o mesmo obtivesse uma melhora considerável no aprendizado de linguagens de programação mais complexas, como a Linguagem C, que é uma das linguagens apresentadas no curso de SI.

Esta escolha é reforçada por Souza (2009) que, em seu trabalho, diz que pelo fato de ser totalmente em português e por utilizar um modelo intuitivo de editor de texto o *VisuAlg* pode ser utilizado logo no início, nas primeiras aulas de programação e que esse contato não causará impactos negativos ao aluno, que poderia encontrar dificuldades relacionadas à complexidade da programação.

Souza (2009) ainda coloca que a experiência mostra que o *VisuAlg* facilita o entendimento de como um programa de computador funciona, possibilitando *feedback* em tempo real sobre a exatidão e correção do pseudocódigo digitado, o que acaba estimulando o aluno a ver os resultados de suas modificações imediatamente.

O público alvo desse trabalho são os alunos do curso Licenciatura em Computação da UFSM-FW, que estiverem realizando as disciplinas iniciais de programação. O trabalho é voltado à modalidade de EaD e sua aplicação será no AVA *Moodle*.

A metodologia de ensino empregada nesse projeto foi a Aprendizagem Baseada em Problemas (ABP) ou *Problem Based Learning* (PBL). O PBL é um método de aprendizado que tem como centro o próprio aluno, tendo o problema como elemento motivador do estudo e integrador do conhecimento.

Leite e Esteves (2005) definem a ABP como um caminho que conduz o aluno para a aprendizagem. Nesse caminho, o aluno busca resolver problemas inerentes a sua área de conhecimento, com o foco na aprendizagem, tendo em vista desempenhar um papel ativo no processo de investigação, na análise e síntese do conhecimento investigado.

A PBL tem como foco a construção de conhecimento com base na investigação que pode envolver perguntas complexas, problemas ou desafios. Com base nessas questões que são abordadas inicialmente, os alunos passam a se envolver com os processos de pesquisa referente ao conteúdo, a elaboração de hipóteses, busca por recursos e aplicações práticas da informação, visando chegar a elaboração de uma solução referente ao desafio ou problema proposto.

O PBL traz um grande benefício aos alunos, tornando inseparável o aprender e o fazer. Para aplicação do PBL é necessário que os alunos trabalhem sua capacidade de raciocínio lógico e a capacidade de cooperação. O PBL tem total ligação com a exploração do contexto, a comunicação entre pares e a criação a partir do conhecimento. Na produção de resultados, que é a etapa final que envolve a Aprendizagem Baseada em Problemas (ou Projetos), os alunos podem organizar suas descobertas de diversas formas, em formato multimídia, com o uso de gráficos e tabelas, ferramentas, vídeos ou aplicativos.

Dentro dessa metodologia, o conteúdo não é apresentado por completo pelo professor e é com base nisso que os alunos começam a trabalhar. Os próprios alunos realizam a busca pelo conhecimento necessário para atingir os objetivos, onde o professor atua como orientador. No PBL é comum que a mesma atividade proposta tenha resultados diferentes podendo, também, agregar aprendizados diferentes.

4.2 O Estudo de Lógica de Programação no Curso de Licenciatura em Computação e a Aplicação da Metodologia PBL

No segundo semestre do currículo do Curso de Licenciatura em Computação da UFSM existe a disciplina que introduz os conceitos de Lógica de Programação, denominada Introdução a Algoritmos. As disciplinas são ministradas na modalidade de EaD, por meio do AVA *Moodle*, seguindo um cronograma bimestral. Nesse cronograma bimestral, cada disciplina possui 10 semanas de atividades: 6 semanas de aulas e 4 semanas de avaliações (avaliação 1, avaliação 2, avaliação substitutiva e exame final), o conteúdo das atividades foi adaptado para que se encaixasse corretamente nesse período de tempo, como podemos observar no Quadro 2, que apresenta o conteúdo que será abordado.

Quadro 2 – Cronograma das aulas

Semana	Conteúdo Abordado
1	Conceitos de Programação em Linguagem Natural.
2	Algoritmos Sequenciais.
3	Algoritmos com seleção (simples, encadeada e composta).
4	Primeira avaliação (realizada a distância).
5	Algoritmos com repetição contada e indeterminada.
6	Estrutura de Dados (vetores e matrizes).
7	Métodos: Procedimentos e Funções.
8	Segunda avaliação (realizada presencialmente).
9	Avaliação substitutiva (recuperação das notas).
10	Exame Final.

Para aplicação da metodologia PBL foram estipulados aspectos essenciais que devem ser levados em consideração (CLARO, 2016):

- Conteúdo significativo: O conteúdo abordado no decorrer do projeto deve ser de total relevância para o aluno, sua profissionalização e evolução;
- Perguntas Instigantes: É importante que as perguntas gerem dúvidas complexas;
- Aluno atuando como sujeito ativo da aprendizagem: Os alunos recebem voz de decisão para desenvolver seus projetos, tomando decisões ativas sobre seu aprendizado;
- Habilidades do Século XXI: Trabalhar o raciocínio lógico, manuseio de tecnologias digitais, a comunicação, o controle de tempo e a criatividade;
- Pesquisa e Inovação: Deve ser estimulada a busca da informação em diversas fontes: vídeos, sites, fóruns, livros e revistas por exemplo. A criatividade deve fazer parte das soluções encontradas pelos alunos;

- *Feedback* e Revisão: O professor ou orientador deve prestar ajuda ao longo do projeto, de forma a oferecer *feedbacks* que venham a enriquecer o aprendizado;
- Apresentação de Resultados: Demonstrar os resultados para os demais colegas é um meio de estimular os alunos em diversos aspectos, tais como: trabalhar a liderança, comunicação interpessoal, criatividade, autoconfiança, além de prepará-los para o mercado de trabalho.

A base da metodologia PBL usada neste projeto foi proposta por Nuutila et al. (2005), sendo composta por nove etapas: 1) Atenção Plena; 2) Introdução do Conteúdo; 3) Pergunta Motivadora; 4) Desafio Proposto; 5) Pesquisa do Conteúdo; 6) Cumprimento do Desafio; 7) Reflexão e *Feedback*; 8). Respondendo à Questão Inicial e 9) Avaliação do Aprendizado. Nuutila et al. (2005) propuseram as 7 últimas etapas destacadas. Neste trabalho incluímos mais duas etapas 1: *Atenção Plena* e 2: *Introdução do Conteúdo*, pois acreditamos ser importante que o aluno tenha uma explicação prévia do conteúdo antes de iniciar as atividades.

Na primeira etapa o aluno deve buscar o que chamamos de atenção plena ou consciência plena, que designa um estado mental que se caracteriza pela retomada da consciência sobre os pensamentos, trazendo a atenção para a experiência do exato momento presente (o aqui e agora). Dessa forma passamos a vigiar nossos pensamentos em busca da aceitação de nossa situação por meio de uma atitude aberta, de curiosidade, ampla e tolerante, dirigida a todos os fenômenos que se manifestam na mente consciente, ou seja, todo tipo de pensamentos, fantasias, recordações, sensações e emoções percebidas no campo de atenção, precisam ser percebidas e aceitas como elas são, deixando de lado qualquer sentimento derivado do medo (*stress*, angústia, nervosismo, tristeza ou preguiça), para assim ter um momento de atenção plena (BISHOP, et al., 2004).

Na segunda etapa é realizada a introdução do conteúdo que será abordado. Essa introdução tem, como objetivo, abordar o tema proposto para a semana de estudos, para que o conhecimento seja compreendido por todos os alunos da turma. Com isso é possível garantir que todos tenham o conhecimento mínimo necessário para iniciar as atividades.

Na terceira etapa são formuladas questões que possuam uma dificuldade relevante, para que o assunto inicial instigue a turma. Dessa forma o tutor pode argumentar com os alunos, a fim de identificar o conhecimento individual de cada um sobre o tema. Esses encontros serão realizados via *Moodle* por meio dos fóruns e *chats*, que podem ser individuais ou em grupo. Nos cursos ministrados na modalidade de EaD, na UFSM, os alunos têm contato com tutores virtuais (que auxiliam os alunos por meio das ferramentas disponíveis no AVA *Moodle*), tutores presenciais (que acompanham as avaliações presenciais nos polos de apoio presencial) e professores formadores (que definem os conteúdos, instrumentos de avaliação e criam as salas de aula virtuais).

Na quarta etapa é lançado o desafio para a turma. O desafio pode ser composto de apresentação, objeto ou pesquisa onde possa se demonstrar o conteúdo e as habilidades adquiridas pelos alunos ao longo do desenvolvimento do processo.

Na quinta etapa é quando os alunos devem buscar se especializar no assunto que envolve o desafio. Os alunos devem buscar expandir seu conhecimento utilizando de

todos os meios possíveis, tais como Internet, livros, jornais, revistas e vídeos por exemplo, sempre visando concluir o objetivo.

Na sexta etapa é quando os alunos concluem o desafio que havia sido proposto, e com isso criam uma forma de colocar os conhecimentos e potenciais que se desenvolveram durante a conclusão do desafio em prática. Caso o projeto envolva maior grau de complexidade é importante o acompanhamento do tutor ou orientador durante o processo.

Na sétima etapa é aberto um momento de reflexão para que os alunos possam debater sobre o tema, esse debate ainda pode ser acompanhado de exercícios. Nessa fase é quando o tutor ou orientador devem estar disponíveis para esclarecimentos e fortalecimento do conhecimento que foi adquirido pela turma. É importante que o tutor siga questionando e instigando os alunos sobre o tema, e que forneça *feedback* sobre os projetos que foram desenvolvidos.

Na oitava etapa é lançada novamente a questão que havia sido proposta à turma na primeira etapa. Nesse momento os alunos devem respondê-la com base nos novos conhecimentos que foram adquiridos.

Na nona etapa é realizada uma avaliação que deve ser conduzida pelo professor. Essa avaliação tem o objetivo de averiguar se os alunos conseguiram atingir o propósito que havia sido planejado.

4.3 Proposta de Aplicação do Método

Com base nas ferramentas definidas e nos passos propostos para aplicar a metodologia PBL, nesta seção apresentamos uma proposta de aplicação, de acordo com um dos conteúdos propostos para a disciplina de Introdução a Algoritmos.

Inicialmente o aluno deve realizar o treinamento de atenção plena que integra mente e corpo, permitindo que o mesmo mude a maneira como experiência seu dia a dia, aumentando sua percepção e como se sente sobre as próprias experiências. A prática exige entrega, disciplina e perseverança para trazer benefícios palpáveis. É proposta uma volta a si mesmo, visando aumentar a consciência e atenção aos nossos atos, mudando a maneira como agimos, vivemos e nos relacionamos. É importante que o aluno reserve pelo menos 2 minutos para a realização de cada um dos exercícios apresentados no Quadro 3.

Quadro 3 – Prática para a Atenção Plena

Exercícios para a Atenção Plena	
1	Primeiro sente-se confortavelmente em um ambiente tranquilo e de seu agrado. Traga a atenção para a sua respiração e nas sensações em volta do seu estômago, a maneira que ele infla quando inspiramos, a sua volta à posição original quando expiramos. Respire de forma totalmente consciente, de forma lenta e sutil durante todo o processo.
2	Deixe sua respiração fluir normalmente agora, apenas sinta as sensações corporais que ocorrem nesse momento, sinta seu corpo e mantenha sua mente vazia, caso

	sinta qualquer desconforto em seu corpo durante a prática, tente identificar o incômodo, porém não tente modificá-lo, apenas tome nota da sua sensação.
3	Faça o mesmo procedimento para sua mente, observe-a, tome consciência a respeito das sensações que você esteja experienciando, por exemplo: “Estou agitado”. Não tente modificá-la, aceite-a.
4	Traga a atenção de sua mente para as sensações que está experienciando. Conecte-se com elas, sem julgar ou comentar, apenas respire e absorva as sensações que está sentido, de forma gentil vá se acalmando e relaxando, diminuindo o fluxo de pensamentos e se tomando cada vez mais consciente do aqui e agora.

Fonte: dos autores, 2017

Na segunda etapa, deve ser realizada a publicação do material de introdução do conteúdo referente a conceitos de algoritmos sequenciais e tutoriais de utilização do *software VisuAlg* no Ambiente Virtual de Aprendizado *Moodle*, onde o conteúdo disposto pode ser composto por vídeos, *links*, apresentações em *PowerPoint*, livros e artigos. A introdução do conteúdo deve levar em consideração os aspectos essenciais citados anteriormente, ofertando apenas conteúdos de real relevância sobre o tema.

O exemplo inicial será o de um algoritmo que, a partir de duas notas, calcule a média aritmética de um aluno, apresentado na Figura 6.

```

1 Algoritmo "Cálculo de média aritmética"
2 // Professor : Sidnei Renato Silveira
3 // Descrição : A partir de duas notas, calcule a média aritmética.
4 // Autor(a) : Naidú Gasparetto de Souza
5 // Data atual : 22/11/2017
6 Var
7 // Seção de Declarações das variáveis
8 nota1, nota2, media: real
9
10 Inicio
11 // Seção de Comandos, procedimento, funções, operadores, etc...
12 escreval ("Digite a primeira nota do aluno:")
13 leia (nota1)
14 escreval ("Digite a segunda nota do aluno:")
15 leia (nota2)
16 media <- (nota1+nota2)/2
17 escreval ("A média do aluno é:", media)
18 Fimalgoritmo

```

```

CA Console simulando o modo texto do MS-DOS
Digite a primeira nota do aluno:
8,5
Digite a segunda nota do aluno:
9,3
A média do aluno é: 8.9
>>> Fim da execução do programa !

```

Figura 6. Exemplo do exercício de média aritmética (Fonte: dos autores, 2017).

Na terceira etapa é onde os alunos serão questionados de forma a gerar um impacto significativo em relação ao conteúdo apresentado sobre algoritmos sequenciais. Nesse momento novamente será publicado como atividade via *Moodle* a questão onde os alunos devem utilizar a ferramenta *VisuAlg* e a partir disso elaborar um algoritmo, a

partir de uma questão motivadora: Como calcular a média ponderada, com diferentes pesos para diferentes instrumentos de avaliação?

Na quarta etapa é lançado o desafio, derivado do algoritmo apresentado como exemplo. Propõe-se que seja elaborado um algoritmo que calcule a média ponderada de um aluno, com base em duas avaliações. A primeira avaliação será composta por duas notas: uma prova (com peso 6) e um trabalho (com peso 4). A segunda avaliação será composta por três notas: 2 trabalhos (cada um com peso 2,5) e uma prova com peso 5. Durante essa atividade a turma deve ser acompanhada pelo tutor, que deve abrir o debate via *chat* ou fórum no ambiente *Moodle*, para que os alunos possam expor suas dúvidas e compartilhar o conhecimento que possuem sobre o tema e suas possíveis soluções com o restante da turma. Essa é uma das principais etapas, onde o tutor deve identificar possíveis dificuldades por parte dos alunos e trabalhar a cooperação entre a turma.

Na quinta etapa é o momento onde os alunos buscam se especializar no tema proposto, criar afinidade com a ferramenta *VisuAlg* e expandir seu conhecimento com base no desafio proposto anteriormente. Como auxílio serão publicados no *Moodle* uma série de conteúdos de apoio (artigos, livros, vídeos aulas, *sites*). Cabe lembrar que a pesquisa individual por parte do aluno deve ser totalmente incentivada por parte do tutor.

Quando chegamos à sexta etapa deve-se iniciar novamente um *chat* ou fórum via *Moodle* para que o aluno possa compartilhar o conhecimento que adquiriu com os demais colegas e, com isso, buscar concluir o desafio que havia sido proposto na quarta etapa. Nesse momento deve-se criar uma *tarefa* no *Moodle* onde os alunos devem enviar os algoritmos que foram criados utilizando o *VisuAlg*.

Na etapa de número sete deve-se utilizar um *chat* ou fórum via *Moodle* onde será iniciado um debate sobre o conteúdo que foi abordado (Algoritmos Sequenciais). Nessa fase é importante que o tutor realize o acompanhamento individual sobre o nível de conhecimento que foi atingido pelos alunos, visando estimular os que encontraram maior dificuldade e uni-los com os que demonstraram maior entendimento do conteúdo. O tutor ainda pode optar por aplicar outro exercício como forma de esclarecimento de dúvidas e fixação do conteúdo, exercícios que também devem ser realizados utilizando a ferramenta *VisuAlg*. Outro papel que o tutor deve manter é o ato de instigar os alunos a prosseguirem na busca pelo conhecimento do conteúdo abordado. Por fim, nessa etapa o tutor deve comunicar sobre os alunos sobre o desempenho obtido no desafio que foi desenvolvido.

Quando chegamos a oitava etapa é novamente abordado o tema inicial do desafio. Nesse momento o tutor deve utilizar o fórum para que os alunos possam responder à questão inicial com base nos conhecimentos que foram adquiridos durante o processo de conclusão do desafio. O aluno deve explanar sua opinião sobre o desafio, dar sugestões de possíveis alterações que podem ser realizadas utilizando como base o algoritmo inicial feito por eles, possíveis continuações do algoritmo que possam vir a ser implementadas e, também, sanar qualquer dúvida que possa ter restado com ajuda do tutor.

Na nona e última etapa é proposto um instrumento de avaliação para medir o desempenho dos alunos. O Quadro 4 apresenta sugestões de exercícios que podem ser utilizados neste instrumento.

Quadro 4 – Avaliação de desempenho

Questões de Avaliação utilizando o <i>VisuAlg</i>	
1	Faça um algoritmo que calcule a média aritmética, média ponderada e ao final apresente se o aluno foi aprovado, reprovado ou ficou em exame.
2	Numa eleição existem três candidatos identificados pelos números 1, 2 e 3. Faça um algoritmo que compute o resultado de uma eleição. Inicialmente o programa deve pedir o número total de votantes. Em seguida, deve pedir para cada votante votar (informando o número do candidato) e ao final mostrar o número de votos de cada candidato.
3	Fazer um algoritmo para calcular e escrever a soma dos cubos dos números pares compreendidos entre B e A ($B > A$). B e A são lidos pelo teclado.
4	Faça um algoritmo que lê uma temperatura em <i>Fahrenheit</i> e calcula a temperatura correspondente em <i>Celsius</i> . Ao final o programa deve exibir as duas temperaturas. – Usar a fórmula: $C = (5 * (F-32) / 9)$.
5	Desenvolva um algoritmo que efetue a leitura de três valores numéricos representando os lados de um triângulo. O algoritmo deverá verificar e informar se os lados fornecidos formam realmente um triângulo (cada lado é menor que a soma dos outros dois lados). Se esta condição for verdadeira, deverá ser indicado qual tipo de triângulo foi formado: isósceles (dois lados iguais e um diferente), escaleno (todos os lados diferentes) ou equilátero (todos os lados são iguais).

Fonte: os autores, 2017

Além do instrumento de avaliação, o debate entre os tutores e alunos pode gerar um novo desafio. Esse desafio deve manter a progressão de dificuldade. Por exemplo, além de realizar o cálculo da média e média ponderada, deve realizar a implementação de um algoritmo que possa dizer se o aluno foi aprovado, reprovado ou ficou em exame. Assim o aluno começa a ser instigado a estudar o próximo conteúdo da disciplina, que são os algoritmos com seleção. Com essa avaliação será possível averiguar se os alunos conseguiram atingir o propósito que havia sido planejado.

5. Considerações Finais

Nesse trabalho que utilizou o método de estudo de caso e teve como base a metodologia PBL, visou-se desenvolver uma metodologia consistente para ensinar lógica de programação para turmas iniciais do Curso de Licenciatura em Computação da UFSM onde as disciplinas são ministradas na modalidade de EaD, por meio do AVA *Moodle*. A metodologia foi elaborada para possível aplicação na disciplina que introduz os conceitos de Lógica de Programação, denominada Introdução a Algoritmos. Acreditamos que dentro da metodologia aqui elaborada, foram atingidos os objetivos propostos inicialmente.

A dificuldade que foi encontrada durante o processo de desenvolvimento desse trabalho é exatamente o desafio que envolve elaborar uma metodologia com capacidade

de proporcionar um real benefício aos processos de ensino e de aprendizagem dos alunos. Muito já foi falado sobre as diversas maneiras de se ensinar Lógica de Programação, reconhecendo as dificuldades que o tema possui. Nesse trabalho a proposta é de colocar o aluno como agente propulsor de seu processo de aprendizado, tendo como objetivo estimulá-lo de diferentes maneiras para que sua mente se torne receptiva aos conteúdos, de forma atrativa e instigante.

Para os trabalhos futuros, fica a possibilidade de validação da metodologia desenvolvida. Essa validação deverá ser feita em uma turma ministrada na modalidade de EaD com base no estudo aqui elaborado. Ainda visando a melhoria do trabalho, para sua continuidade pode-se explorar novas ferramentas que auxiliam no processo de aprendizado da Lógica de Programação e as demais metodologias que venha a somar para o trabalho.

Referências

- ALMEIDA, E. S. de; COSTA, E. B.; SILVA, K. S. **AMBAP: Um Ambiente de Apoio ao Aprendizado de Programação**. 2002. 10 f. TCC (Graduação) - Departamento de Tecnologia da Informação, Universidade Federal de Alagoas (tci/ufal), Campus A. C. Simões Br 101 Norte Km 14 Bl 12 – 57072-970 – Maceió, Al, 2002. Disponível em: <<http://www.lbd.dcc.ufmg.br/colecoes/wei/2002/006.pdf>>. Acesso em: 09 set. 2017.
- ARAÚJO, D. C.; RODRIGUES, A. N.; SILVA, C. V. de A.; SOARES, L. S. (2015) **O Ensino da Computação na Educação Básica Apoiado por Problemas: Práticas de Licenciados em Computação**. In: **Anais do XXIII WEI (Workshop sobre Educação em Computação) Garanhuns**. Disponível em:<<http://www.lbd.dcc.ufmg.br/colecoes/wei/2015/014.pdf>> Acesso em 04 de setembro de 2017.
- BEZERRA, F.; DIAS, K. (2014). **Programação de Computadores no Ensino Fundamental: Experiências com Logo e Scratch em escola pública**. In XXII Workshop sobre Educação em Informática, Brasília, DF: SBC.
- BELLONI, M. L. (2006). **Educação a Distância**. Campinas, Autores Associados.
- BISHOP, S.R., LAU, M., SHAPIRO, S., CARLSON, L., et al. (2004). **Mindfulness: A Proposed Operational Definition**. *Clin Psychol Sci Prac* 11:230–241.
- BORGES, M. A. F. (2000) “**Avaliação de uma Metodologia Alternativa para a Aprendizagem de Programação**”. *VIII Workshop de Educação em Computação – WEI, 2000*. Curitiba, PR, Brasil.
- BRUNER, J. (2001). **A Cultura da educação**. Porto Alegre: Artmed.
- CATALDO, W., Junior, R., Lavorato, Â. S., & Camargos, D. (2017). **Ensino de Lógica de Programação através do Jogo Defense of the Ancients 2**, (Cbie), 648–657. <https://doi.org/10.5753/cbie.wie.2017.648>
- COOPER, S.; DANN, W.; PAUSCH, R. (2003). **Using animated 3d graphics to prepare novices for cs1**. *Computer Science Education*.

- CHAVES DE CASTRO, T., CASTRO JÚNIOR, A., MENEZES, C., BOERES, M. e RAUBER, M. (2003) **Utilizando Programação Funcional em Disciplinas Introdutórias de Computação**. XI Workshop de Educação em Computação – WEI 2003. Campinas, SP, Brasil.
- CLARO, M. (2016) **Aprendizagem Baseada em Projetos (PBL)**. Disponível em: <<https://www.moodlelivre.com.br/1927-aprendizagem-baseada-em-projetos-pbl-em-7-passos-infografico>> acesso: novembro de 2017.
- DELGADO, C., XEXEO, J. A. M., SOUZA, I. F., CAMPOS, M., RAPKIEWICZ, C. E. (2004) “**Uma Abordagem Pedagógica para a Iniciação ao Estudo de Algoritmos**”. XII Workshop de Educação em Computação (WEI'2004). Salvador, BA, Brasil.
- FERREIRA, C.; GONZAGA, F.; SANTOS, S. (2010) **Um Estudo sobre a Aprendizagem de Lógica de Programação por Demonstração**. In XVIII WEI (Workshop sobre Educação em Computação) Belo Horizonte. Disponível em: <http://www.inf.pucminas.br/sbc2010/anais/pdf/wei/st06_03.pdf>. Acesso em: 05 de setembro de 2017.
- FREITAS, E. (2013) **Por dentro da EaD**. Disponível em: <<http://mundogeo.com/blog/2013/11/15/por-dentro-da-ead/>>. Acesso em: 25 out. 2017.
- FRIESEN, N. (2012) Report: **Defining Blended Learning**. Disponível em: <http://learningspaces.org/papers/Defining_Blended_Learning_NF.pdf>. Acesso em: 21 jun. 2016.
- FRIEDRICH, R. V. et al (2012). **Proposta Metodológica para a Inserção ao Ensino de Lógica de Programação** com Logo e Lego Mindstorms. Anais do SBIE 2012. Disponível em: Acesso em: 18 maio 2015.
- GARLET, D.; BIGOLIN, N. M.; SILVEIRA, S. R. **Uma Proposta para o Ensino de Programação de Computadores na Educação Básica**. 2016. Trabalho de Conclusão - Curso de Sistemas de Informação, Departamento de Tecnologia da Informação, Universidade Federal de Santa Maria, Frederico Westphalen, 2016.
- HERNÁNDEZ, F. (1998). **Transgressão e mudança na educação**. Porto Alegre: Artmed.
- LEAL, D.; AMARAL, L. (2004) **Do ensino em sala ao e-Learning**. Disponível em: <http://www.campusvirtual.uminho.pt/uploads/celda_av04.pdf>. Acesso em: 21 jun. 2016.
- LEITE, L. S., VIEIRA, M. L. S e SAMPAIO, M. N. (1997 ou 1998? No texto aparece 1998) Atividades não presenciais: **preparando o aluno para a autonomia** In Tecnologia Educacional. Rio de Janeiro, ABT. Ano XXVI. Nº 141. Abr/Mai/Jun/1997. P. 36-40.
- LEITE, L.; ESTEVES, E. **Ensino orientado para a Aprendizagem Baseada na Resolução de Problemas** na Licenciatura em Ensino da Física e Química. In: Bento Silva e Leandro Almeida (Eds.). Comunicação apresentada no VIII Congresso Galaico-Português de Psicopedagogia. Braga: CIED - Universidade do Minho, p. 1751-1768, 2005

- LEMGRUBER, M. S. (2008). **Educação a Distância: para além dos caixas eletrônicos**. Anais Do 2o Simpósio Hipertexto E Tecnologias Na Educação. Retrieved from http://portal.mec.gov.br/arquivos/conferencia/documentos/marcio_lemgruber.pdf
- LUCKESI, C. (2000) **Avaliação da aprendizagem escolar: estudos e proposições**. São Paulo: Cortez.
- MACEDO, L., PETTY, A. L. S., PASSOS, N. C. (2005). Os jogos e o lúdico na aprendizagem escolar.
- MARION, W. (1999) "**CS1: What Should We Be Teaching?**", Annual Joint Conference Integrating Technology into Computer Science Education ,Working group reports from ITiCSE on Innovation and technology in computer science education, pp 35-38, Polônia.
- MORAIS, R. X. T. (2003) **Software Educacional: a importância de sua avaliação e do seu uso nas salas de aula**. Faculdade Lourenço Filho. Monografia. Disponível em: <http://www.flf.edu.br/revista-flf/monografias-computacao/monografia-rommel-xenofonte.pdf>. Acesso setembro, 2017.
- MORAN, J. (1994) **Novos caminhos do ensino a distância**. 1994. 3 f. Tese (Doutorado) - Curso de Projetos Inovadores A Educação, CEAD - Centro de Educação A Distância. Senai, Rio de Janeiro, 1994 (atualizado em 2002). Disponível em: <>. Acesso em: 07 set. 2017.
- MOTA, F. P.; RIBEIRO, N. F. A.; EMMENDORFER, L. (2014). **Desenvolvendo o Raciocínio Lógico no Ensino Médio: uma proposta utilizando a ferramenta Scratch**. Universidade Federal do Rio Grande (furg), Carreiros. Disponível em: <http://www.br-ie.org/pub/index.php/sbie/article/view/2964/2698>. Acesso em: 26 out. 2017.
- NÓVOA, A. **Professores imagens do futuro presente**. Lisboa: Educa, 2009.
- NUUTILA E; TOMAS S., MALMI L (2005). **PBL and Computer Programming – The Seven Steps Method with Adaptations**. Computer Science Education, 15(2):123–142.with Adaptations. Computer Science Education, 15(2):123–142.
- PEREIRA JÚNIOR, J.C.R., Rapkiewicz, C. (2004) “**O Processo de Ensino-Aprendizagem de Fundamentos de Programação: Uma Visão Crítica da Pesquisa no Brasil**”, WEI RJES.
- PULINO FILHO, A. R. (2004). **Introdução ao Moodle: ambiente de aprendizagem: Módulo 1**. Disponível em: http://ava.bahia.fiocruz.br/pluginfile.php/704/mod_resource/content/1/Manual_Moodle_UNB_-_Modulo_1.pdf. Acesso em: 25 out. 2017.
- RODRIGUES, A. **Manual do VisuAlg**. (2010). Curso de Curso Técnico em Informática, Instituto Federal de Educação, Ciência e Tecnologia, Instituto Federal de Educação, Ciência e Tecnologia, Ceará - Campus Iguatu, 2012.
- RODRIGUES, M. C. (2002) “**Como Ensinar Programação?**”. Informática – Boletim Informativo Ano I n° 01, ULBRA. Canoas, RS, Brasil.
- SANTOS, G., DIRENE, A. I., GUEDES, A. L. P. (2003) “**Autoria e Interpretação Tutorial de Soluções Alternativas para Promover o Ensino de Programação de**

Computadores”. XIV Simpósio Brasileiro de Informática na Educação – SBIE 2003. Rio de Janeiro, RJ, Brasil.

SILVA, M. F; CORTEZ, R. C. C.; OLIVEIRA, V. B. (2013) **Software Educativo como auxílio na aprendizagem da matemática**: uma experiência utilizando as quatro operações com alunos do 4º Ano do Ensino Fundamental I. Educação, Cultura e Comunicação, Fatea, V.4, n.7. Disponível em: <<http://publicacoes.fatea.br/index.php/eccom/article/viewFile/594/424>>. Acesso em abril, 2016.

SCHULTZ, M. R. O. (2003) “**Metodologias para Ensino de Lógica de Programação de Computadores**”. Monografia de Especialização (Ciência da Computação). Universidade Federal de Santa Catarina (UFSC), Florianópolis, SC, Brasil.

SCRATCH (2017). **ABOUT Scratch** (Scratch Documentation Site). Disponível em: <http://info.scratch.mit.edu/About_Scratch>. Acesso em: 07 set. 2017.

SOUSA, A., SILVA, S., RAIOL, A.A.C., SARGES, J., BEZERRA, F. (2015) **O Universo Lúdico da Programação com Logo no Ensino Fundamental**. In: 23º WEI – Workshop sobre Educação em Computação – CSBC 2015.

SOUZA, C. M. (2009). **VisuAlg - Ferramenta de Apoio ao Ensino de Programação**. Revista TECEN, 2, 1–9. Retrieved from <http://www.uss.br/pages/revistas/revistateccen/V2N22009/ArtigoVisuAlgSOUZA.pdf>

STALLMAN, R. M. (1989) **GNU: General Public License**. Disponível em: <<http://www.gnu.org/licenses/old-licenses/gpl-1.0.html>>. Acesso em: 21 jun. 2016.

VALENTE, J. A. (2005). **Informática na Educação - O Computador auxiliando o processo de mudança na escola**. Nied-Unicamp E Ced-Pucsp, 4. Retrieved from http://www.aedi.ufpa.br/parfor/letras/images/pdf/at_distancia/castanhal_1.2013/castanhal_2010.020/1.2013_castanhal_2010-020_tecn_ed_e_ens_do_port_texto_i_prof_williane_santos_cpia.pdf

YIN, R. K. (2001) **Estudo de Caso: planejamento e métodos**. 2. ed. Porto Alegre: Bookman.