

Relato de experiência de alunos do curso de Licenciatura em Computação do IFMG - campus Ouro Branco na utilização de objetos de aprendizagem desplugados e do Scratch como instrumentos no ensino de programação

Dimas Antônio Silveira Gonçalves, Gislane Moura da Silva, Ronaldo Santos da Luz, Carlos Eduardo Paulino Silva

Curso de Licenciatura em Computação – Instituto Federal de Minas Gerais –
campus Ouro Branco – MG – Brasil

Rua Afonso Sardinha, 90 – Pioneiros – Ouro Branco/MG – CEP: 36.420-000

dimasasg@gmail.com, gismouralaine@hotmail.com, ronaldoluz67@gmail.com,
carlos.paulino@ifmg.edu.br

Abstract. *This article reports the experience of Graduation in Computer's students in using Scratch as a tool for teaching programming and the construction of a learning object unplugged, activity performed during the course Foundations and Methodologies of Teaching Computer Science in Education I.*

Resumo. *Este artigo relata a experiência de alunos do curso de Licenciatura em Computação na utilização do Scratch como ferramenta de ensino de programação e a construção de um objeto de aprendizagem desplugado, durante a disciplina Fundamentos e Metodologias do Ensino de Informática na Educação I.*

1. Introdução

A vida dos indivíduos no atual cenário mundial mostra-se cada vez mais ligada à informática e com os avanços tecnológicos o acesso está cada vez mais fácil e cotidiano. A informática está presente nas mais diferentes práticas do dia a dia, desde as relações de trabalho e consumo, até as relações sociais. Daí a importância de que os indivíduos saibam como operar e desfrutar desses avanços. Aqueles que ainda não estão familiarizados com essas novas tecnologias sentem uma enorme dificuldade em adaptar-se à nova realidade. Esse processo é conhecido como a exclusão digital. De acordo com (Brito *et al.*, 2005):

É neste contexto que se aplica o termo exclusão digital, privando — seja por motivos sociais, econômicos, políticos e/ou culturais — o acesso às vantagens e aos benefícios trazidos pelas novas tecnologias de informação e comunicação. A desigualdade registrada entre pobres e ricos entra agora na era digital e ameaça se expandir com a mesma rapidez da informática.

Uma das maneiras de tentar minimizar esse problema, além de garantir o acesso às tecnologias, é dar acesso à educação no âmbito tecnológico. Existe porém a necessidade de se ter profissionais qualificados para o ensino de informática. A formação desses profissionais mostra-se um desafio, já que sua atuação será tanto em níveis acadêmicos quanto em ambientes informais de educação. Os cursos tradicionais não contemplam a temática da educação e a maioria dos profissionais da área não está qualificada para atuar diretamente na área educacional.

Constata-se uma carência de profissionais de educação em computação, o que priva as escolas, do ensino de disciplinas desta área em cursos profissionalizantes; da introdução de fundamentos dessa ciência nos currículos regulares da educação fundamental e média; da interlocução em projetos interdisciplinares e transversais; da informatização escolar; da busca, análise e projeto de *softwares* educacionais e objetos de aprendizagem de qualidade, dentre muitos outros. (Prietch e Pazeto, 2009)

Quando falamos do ensino de informática, levamos em conta não só o ensino de matérias relacionadas à computação, mas também a orientação para a utilização cotidiana. Essa utilização consiste no uso corriqueiro da informática em atividades como utilizar terminais eletrônicos, manuseio de dispositivos eletroeletrônicos e assimilação de termos ligados a informática. Nesse aspecto faz-se necessário desmistificar o uso da informática visando dar fim à exclusão digital além de levar um maior número de pessoas a buscar formação na área de tecnologia.

Como reflexo dessa necessidade surge o curso de Licenciatura em Computação, que tem como objetivos preparar profissionais que atuem como mediadores entre os professores que já atuam no ensino de diversos conteúdos e não tem conhecimentos de tecnologia, e a tecnologia, e formar professores que saibam ensinar computação. A formação desse profissional, segundo o Currículo de Referência para Cursos de Licenciatura em Computação da Sociedade Brasileira de Computação (SBC, 2002), abrange um enfoque de formação especializada e multidisciplinar e o currículo deixa claro a caracterização do curso como de formação profissional docente em computação, independentemente do campo de atuação profissional. Portanto, o licenciado em computação aparece como uma opção para se tentar contornar os problemas observados do aprendizado na área de tecnologia e do aprendizado com tecnologia.

2. Desafios na aprendizagem

O aprendizado de disciplinas ligadas à computação exige um alto nível de abstração, como nos aponta (Costa *et al.*, 2012):

A assimilação dos conceitos da grande área da Computação é um processo desafiador, sendo para muitos estudantes uma tarefa difícil compreender a abstração deste braço da Ciência. Disciplinas iniciais nessa modalidade de curso, como programação, revelam um elevado índice de abandono, que de acordo com Castro (2003), é devido a pouca habilidade que os estudantes possuem em construir algoritmos e estruturas de dados.

Essa abstração é pouco trabalhada no ensino médio. A maioria das disciplinas da área propedêutica não estimula essa forma de pensamento. Isso faz com que a maioria dos alunos só tenha contato com ela em cursos de nível técnico ou superior. Essa deficiência pode ser apontada como uma das causas para a grande evasão nos cursos superiores da área de tecnologia. Além da abstração, a lógica também não é contemplada nos currículos de ensino fundamental e médio, sendo que ela é de suma importância para que o aluno possa compreender os conceitos necessários da área de programação. Como nos apontam (Branco Neto e Schuvartz, 2007):

(...) os cursos da área de computação enfrentam um grande problema com as disciplinas de introdução à programação de computadores, as quais visam ensinar como utilizar o computador para solucionar problemas. Acadêmicos iniciantes, ao se depararem com a disciplina, sentem-se incapazes de programar, devido ao conjunto de habilidades que a programação exige como capacidade para solucionar problemas, raciocínio lógico, habilidade matemática, capacidade de abstração, entre outras.

Em seu trabalho, (Nunes, 2008) faz um levantamento a partir de dados do Censo da Educação Superior 2006, fornecidos pelo INEP/MEC apontando os resultados apresentados na Tabela 1.

Tabela 1: Percentual de evasão (Evasão/Total de matrículas) nos Cursos de Computação, em 2001 e 2006.

Cursos	Privada		Pública	
	2001	2006	2001	2006
Ciência da Computação (CC)	9,83	11,75	6,59	8,57
Engenharia da Computação (EC)	8,31	14,15	5,24	4,20
Sistemas de Informação (SI)	12,76	12,06	4,91	11,38
Licenciatura em Computação (LC)	13,79	14,39	9,09	8,49
Tecnólogo (Tec)	9,10	12,58	9,92	11,60
Outros Cursos (OC)	15,65	15,16	11,43	13,41

Conforme observado na tabela 1 há um alto nível de evasão dos cursos superiores da área de computação. Como instrumento de comparação citamos o trabalho de (Silva Filho *et al.*, 2007) no qual é demonstrada a taxa de desistência de cursos superiores, sendo os cursos separados em 8 áreas. As áreas de Ciências, Matemática e Computação apresentam 28% de desistência, abaixo somente do setor de Serviços, onde a taxa é de 29%, e acima de áreas como Ciências Sociais, Negócios e Direito, 25%, e Humanidades e Artes, 23%.

Para tentar minimizar a evasão de cursos ligados a computação faz-se necessário repensar os métodos de ensino para que sejam trabalhadas tanto a abstração quanto a lógica já no ensino básico, fazendo com que os alunos cheguem ao ensino superior com um conhecimento prévio na área.

Uma das maneiras de se trabalhar esses conteúdos é a utilização de objetos de aprendizagem e a introdução de *softwares* de programação para o ensino de informática nas escolas de nível médio. Por isso, como parte da disciplina de Fundamentos e Metodologias do Ensino de Informática na Educação I, ministrada pelo professor Carlos Eduardo Paulino Silva no campus Ouro Branco do Instituto Federal de Minas Gerais, houve o incentivo à utilização de um programa de ensino de programação lúdico diferente dos programas tradicionais, e a construção de um objeto de aprendizagem. Para isso os alunos foram divididos em grupos e cada grupo montou seu objeto de aprendizagem.

Dentre os diversos conteúdos que podem ser ensinados através de objetos de aprendizagem, foi escolhido pelo nosso grupo os conectivos lógicos. Já na área de programação existem diversos programas que facilitam o ensino e aprendizagem da programação. Entre eles temos SuperLogo (SuperLogo, 2013), Kodu (Kodu, 2013), RoboMind (RoboMind, 2013), Alice (Alice, 2013) e o Scratch (Scratch, 2013). No andamento da disciplina foi definido um desses programas para cada grupo, que deveria ser utilizado como apoio para explicação dos conteúdos de programação. Para o nosso grupo foi definido o Scratch.

3. Conjunção, Disjunção, Negação e Tabela verdade

Um dos conceitos primordiais para o estudo da lógica é o de conectivos lógicos, utilizados para formar proposições. As proposições devem ter apenas dois valores lógicos possíveis de resposta, verdadeiro ou falso, e seguem obrigatoriamente três princípios: o Princípio da identidade diz que uma proposição verdadeira é verdadeira, uma proposição falsa é falsa; o da Não-Contradição diz que nenhuma proposição poderá ser verdadeira e falsa ao mesmo tempo; e o terceiro fala que uma proposição ou será verdadeira, ou será falsa: não há outra possibilidade, este é o princípio do Terceiro Excluído (Alencar Filho, 2002).

Um dos conectivos lógicos é o “e”, conhecido como conjunção. Com ele a proposição será verdadeira apenas se ambas forem verdadeiras, sendo que basta que uma delas seja falsa para que toda ela se torne falsa, assim como se ambas forem falsas, o resultado também será falso. Já a disjunção, ou conectivo lógico “ou”, só tornará a proposição falsa se ambas as partes forem falsas. Nos demais casos ela será sempre verdadeira. Por fim temos o “Não”, com o qual podemos negar qualquer proposição. Existem outros tipos de conectores, a disjunção exclusiva, condicional e bicondicional, porém, não os trabalharemos aqui.

Existe uma maneira de se verificar se uma proposição composta é verdadeira ou falsa, baseando-se no valor verdade dos termos que a compõem. Ou seja, verificamos a veracidade do todo a partir de cada subparte através da tabela verdade.

Tabela 2 – Tabela Verdade

A	B	A e B	A ou B	Não A	Não B
V	V	V	V	F	F
V	F	F	V	F	V
F	V	F	V	V	F
F	F	F	F	V	V

4. Computação desplugada

Além do Scratch, durante a disciplina fomos incentivados a desenvolver um objeto de aprendizagem seguindo o conceito de computação desplugada (Bell e Witten, 1995), conceito esse que baseia-se na não utilização do computador para o ensino de fundamentos da Ciência da Computação. A vantagem desse método é a possibilidade de se trabalhar a computação em lugares que não disponham de infraestrutura (energia elétrica, *hardware* ou *software*) necessária para a utilização de computadores. Seguindo esse conceito deveríamos construir um objeto desplugado que pudesse ser uma ferramenta didática durante as aulas.

4.1. Construção do objeto

Existem diferentes tipos de objetos de aprendizagem, acreditamos que uma das maneiras de se incentivar a participação dos alunos é a aplicação de um jogo. Como ressalta (Sousa *et al.*, 2010):

As teorias da Epistemologia Genética de Piaget e Sócio-Cultural de Vygotsky motivam enfaticamente o uso do lúdico no processo de

aprendizagem. Piaget considerava o jogo essencial na vida das crianças e adolescentes por tratar-se do “berço obrigatório” das atividades intelectuais.

Com o jogo podemos incentivar a competição, a interatividade entre os alunos e a possibilidade de se reutilizar o objeto como maneira de testar o conhecimento obtido.

4.2. O jogo EvOU

O jogo foi desenvolvido utilizando-se a princípio um baralho convencional que foi dividido em partes iguais entre os jogadores, e um tabuleiro confeccionado em EVA. Depois do jogo apresentado ao professor e demais discentes da disciplina para a discussão de sua aplicabilidade, foram feitas melhorias. Foi proposto a não utilização de um baralho convencional, pois a sua utilização em sala de aula descaracterizaria o cunho educativo e pedagógico do jogo, além de requerer a associação entre cores e números. Dessa forma produzimos um baralho com cartas contendo os números 0 e 1 onde a associação é direta. Outra mudança feita no jogo, foi a inclusão de cartas representando a porta lógica NÃO, para complementar o conteúdo a ser trabalhado.

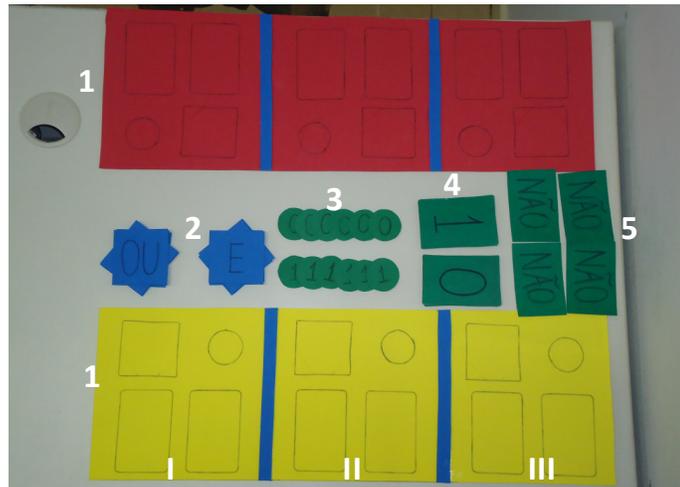


Figura 1 – Peças do jogo EvOU

Observando a Figura 1 destacamos as seguintes peças do jogo EvOU: dois tabuleiros (1), divididos cada um em 3 casas I, II e III, onde são encaixadas fichas representando as portas “E” e “OU” (2), além de fichas com os números “0” e “1” (3), representando os números binários de saída. Essa arquitetura traz como vantagem a variabilidade na montagem do jogo, com isso o jogador sorteia ou escolhe as peças a serem utilizadas. Além de um baralho com vinte cartas com o número “1” e vinte com o número “0” (4), e quatro cartas com a palavra “NÃO” (5).

A escolha das peças para montagem do tabuleiro pode ser feita através de sorteio ou definida pelos jogadores. Das doze cartelas que representam o “E” e o “OU”, três são colocadas em cada tabuleiro. As fichas contendo os números “0” e “1” também são distribuídas dessa mesma maneira. Depois de embaralhadas, as cartas “0” e “1” são distribuídas em número igual entre os jogadores e devem permanecer viradas para baixo. Além disso cada um recebe duas cartas “NÃO”. Depois do tabuleiro montado e das cartas “0” e “1” serem distribuídas, inicia-se a partida.



Figura 2 – Jogo EvOU montado

Os alunos iniciam a partida, jogando duas cartas sobre o tabuleiro. O objetivo é fazer com que a combinação de cartas resulte na saída desejada.

O tabuleiro é dividido em três casas, sendo que para mudar de uma para outra o jogador deve jogar as cartas que resultem na combinação correta. Caso isso não ocorra ele continua jogando as cartas na mesma casa, até que consiga passar para a próxima. Para finalizar o jogo podem ser definidas duas formas de vitória, ou o jogador que completar as três casas primeiro, ou o que conseguir acabar o baralho passando por mais casas. Cada jogador tem direito a duas cartas “NÃO”, que podem ser usadas no momento que o jogador preferir, sendo que ela modifica o valor da carta que foi jogada, devendo ser utilizada antes que o adversário inicie a jogada. Na figura 3 temos uma representação da dinâmica do jogo em forma de fluxograma.

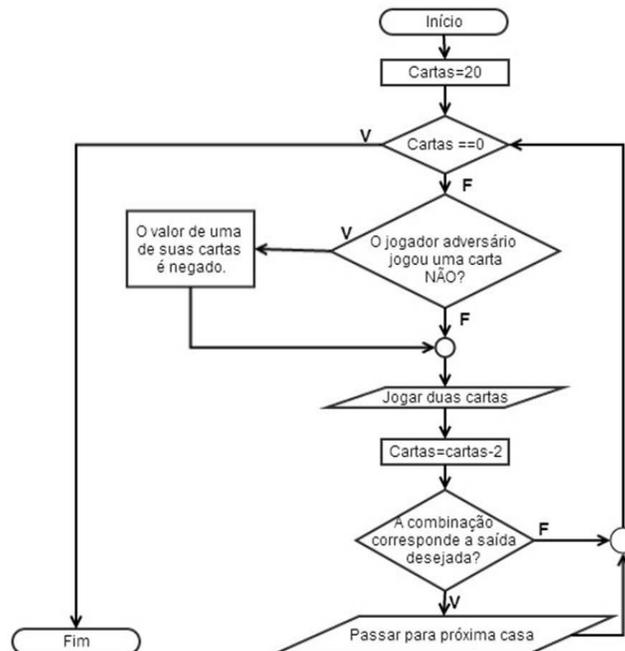


Figura 3 – Fluxograma da dinâmica do jogo

5. Programa Scratch

O Scratch é uma linguagem de programação desenvolvida por Lifelong Kindergarten Group no Media Lab, MIT, com financiamento da National Science Foundation, Intel Foundation, Nokia e do consórcio de pesquisa do MIT Media Lab. Ele possibilita a criação de programas interativos, jogos e animações além de contar com um sítio de compartilhamento das criações na *Web*. O programa funciona de forma bem intuitiva e visual, lembrando os blocos de montar semelhantes ao LEGO. Os comandos já vêm prontos, separados em categorias, que são diferenciadas pela coloração de cada função e devem ser agrupados de modo lógico. Existe a possibilidade de trabalhar com inúmeros objetos, alguns já presentes no programa e outros que podem ser adicionados pelo usuário.

No decorrer da disciplina, com posse da ferramenta Scratch, semanalmente apresentávamos uma miniaula com conteúdo proposto pelo professor. Cada conteúdo deveria ser trabalhado dentro do programa e as aulas ministradas aos demais discentes e ao professor. Durante a utilização do programa foram constatados alguns problemas de usabilidade.

5.1 Dificuldades apresentadas na utilização do Scratch

O Scratch funciona através de um palco, onde você escolhe o cenário para o programa. Nele é possível adicionar os objetos e utilizar diferentes estruturas de controle e de decisão, o que possibilita trabalhar diferentes conteúdos. Durante a utilização do programa tivemos algumas dificuldades. Uma das dificuldades enfrentadas foi a interação entre objetos diferentes. Quando adicionamos aos objetos alguns comandos, eles devem ser copiados para todos, sendo que não conseguimos fazer uma única sequência de comandos que fosse automaticamente replicada a todos os objetos. Dessa forma a interação entre objetos diferentes fica prejudicada, levando o programa a ser executado com alguns erros. Além disso ele não possui um debugador que aponte os erros na programação. O código sempre é executado, independente de estar certo ou não. Essa dificuldade ocorre principalmente quando utiliza-se múltiplos objetos. Diante dessas dificuldades sentimos falta de um comando para que o programa voltasse ao estado inicial. Esse comando não existe, e por isso é preciso resetar manualmente, desfazendo os movimentos realizados.

Vale ressaltar que tivemos pouco tempo de utilização do programa. Acreditamos que um maior contato com o Scratch diminuiria a ocorrência das dificuldades citadas anteriormente. Existem manuais de utilização do Scratch disponíveis na *Web*, porém, os que encontramos detalham muito mais a interface do que a programação em si. O que auxilia bastante é a existência de inúmeros projetos compartilhados na internet por usuários.

6. Execução da aula

Como atividade final da disciplina preparamos uma aula de modo a integrar o Scratch com o objeto de aprendizagem desplugado desenvolvido, aplicada aos alunos do curso superior em Licenciatura em Computação e com 50 minutos de duração. O principal objetivo da aula foi ensinar os conectivos lógicos E, OU e NÃO, pertencentes ao conteúdo de lógica computacional, e a montagem de suas expressões.

Durante a aula foi apresentada aos alunos uma introdução sobre os conectivos lógicos E, OU e NÃO com auxílio do *data show*. Os alunos foram separados em

grupos, formando times que disputaram o jogo com o intuito de verificar o aprendizado. Depois da partida, foi apresentada uma pequena introdução ao programa Scratch e os alunos utilizaram o programa para resolução de um problema proposto, relacionado aos conectivos.

Os conhecimentos prévios necessários aos alunos eram o conceito de números binários, noções básicas de informática para utilização do programa, além de conceitos básicos de programação. A fim de testar os conhecimentos adquiridos na aula, como atividade final, foi proposto aos alunos que elaborassem algumas expressões a partir de sentenças dadas pelos professores, além de utilizar o Scratch para a elaboração de um pequeno programa onde foram empregados os conectivos lógicos. Na figura 4 temos o programa produzido pelos alunos durante a aula com o Scratch.

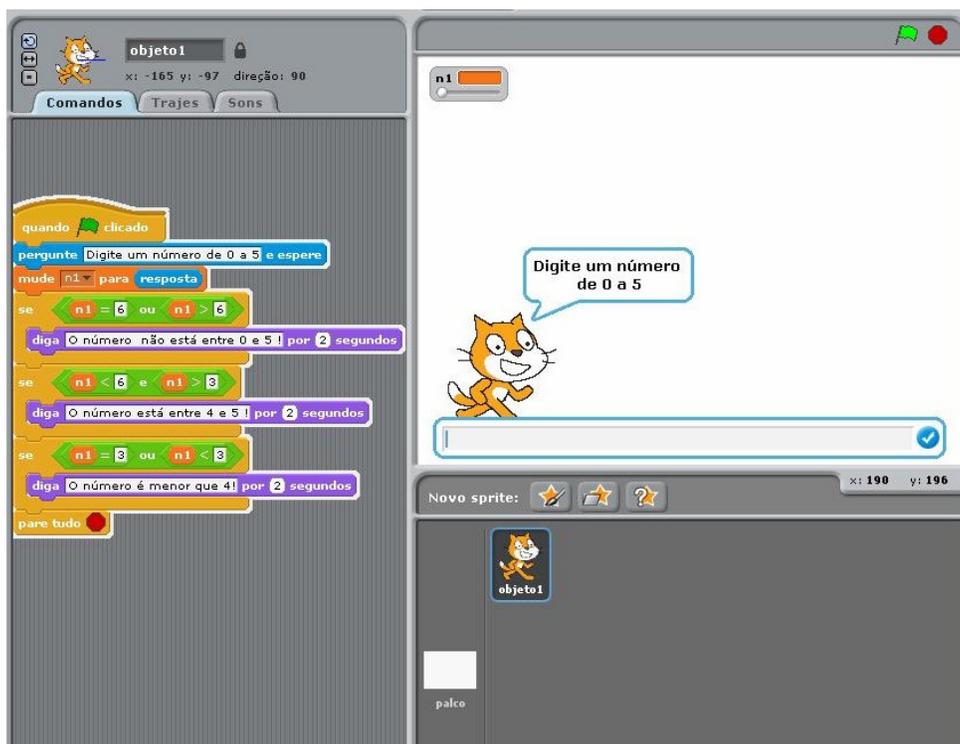


Figura 4 – Programa desenvolvido no Scratch

7. Considerações finais

Essa disciplina nos proporcionou um contato maior com a realidade do ensino da informática, adquirindo experiência docente. Durante o andamento da disciplina, através da construção do objeto de aprendizagem e da utilização do Scratch, ampliamos nossa visão das maneiras de se ensinar a programação. Além de perceber que o ensino da programação não está somente vinculado à utilização do computador, mas sim a uma ampla gama de objetos que podem ser utilizados. O jogo EvOU é apenas uma dessas ferramentas, assim como o Scratch.

Acreditamos que a aplicação em sala de um objeto de aprendizagem seja uma forma mais estimulante aos alunos, pois foge à maneira tradicional de como a lógica

computacional é ensinada na sala de aula. O uso de programas lúdicos como o Scratch, também facilita, ao mesmo tempo, o interesse e a aprendizagem.

Apesar de não termos ministrado a aula para alunos do ensino médio, a aula foi apresentada aos alunos da disciplina Fundamentos e Metodologias do Ensino de Informática na Educação 1 do 3º período do curso de Licenciatura em Computação do IFMG – campus Ouro Branco. Essa troca de experiências dentro da sala foi de fundamental importância para a aquisição de conhecimento que poderá ser aplicado nas aulas que serão ministradas aos alunos do ensino médio. Além disso, a possibilidade de poder criar o objeto contribuiu de maneira primordial para uma melhor compreensão do ensino de informática.

Este tipo de ferramenta possibilita um maior envolvimento com o campo da educação. Para o licenciando em computação é de suma importância que se façam trabalhos como esse, pois, incentivam o uso da criatividade em relação a metodologias didáticas e pedagógicas construindo assim um melhor percurso profissional. Além disso, é um sinal de mudança na própria área da computação, essa abertura para a educação, mostra-se um fator positivo, que pode auxiliar os profissionais da área a lidarem e enfrentarem as dificuldades encontradas até então.

8. Trabalhos futuros

Pretendemos ministrar a aula sobre conectivos lógicos para as turmas do curso integrado em informática do IFMG – campus Ouro Branco para que possamos avaliar o uso do Scratch com o objeto de aprendizagem produzido como ferramentas didáticas.

Como maneira de expandir os conteúdos a serem trabalhados no jogo pretendemos incluir a possibilidade de utilizar expressões lógicas, e além disso, pensar novas formas de interação entre os tabuleiros e utilizar o conceito de objetos tangíveis, para que o jogo não se torne monótono.

Referências Bibliográficas

- Alencar Filho, E. (2002). “Iniciação à Lógica Matemática”. Editora Nobel.
- Alice. (2013). Disponível em: <http://www.alice.org/index.php>. Acesso em 20/10/2013.
- Bell, T. C. G.; Witten, I. (1995). “Computer Science Unplugged: Capturing the interest of the uninterested”. Anais do NZ Computer Conference, Wellington, Nova Zelândia.
- Branco Neto, W. C.; Schuvartz, A. A. (2007). “Ferramenta Computacional de Apoio ao Processo de Ensino-Aprendizagem dos Fundamentos de Programação de Computadores”. *In*: Simpósio Brasileiro de Informática na Educação (SBIE).
- Brito, S. C., Masiero, R. F. C., Magalhães, D. G. B. S., Vilas Boas, A. A. (2005). “Impacto da Exclusão Digital na Sociedade e no Mercado de Trabalho”. *In*: II Simpósio de Excelência em Gestão e Tecnologia, p. 986-994.
- Costa T.; Batista A.; Maia M.; Almeida L.; Farias A. (2012). “Trabalhando Fundamentos de Computação no Nível Fundamental: experiência de licenciandos em Computação da Universidade Federal da Paraíba”. *In*: XX Workshop sobre Educação em Computação (WEI).

- Kodu. (2013). Disponível em: <http://www.kodugamelab.com>. Acesso em 20/10/2013.
- Nunes, D. J. (2008). “Estatísticas da Educação Superior: Área da Computação”. Disponível em: <http://www.sbc.org.br/>. Acesso em 11/08/2013.
- Prietch, S. S., Pazeto, T. A. (2009). “Análise, Sugestões e Perspectivas de um Curso de Licenciatura em Informática”. *In: XVII Workshop sobre Educação em Computação (WEI)*.
- RoboMind. (2013). Disponível em: <http://www.robomind.net/pt/>. Acesso em 20/10/2013.
- SBC. (2002). “Referências dos Cursos da Área de Computação”. Disponível em: <http://www.sbc.org.br/>. Acesso em 11/08/2013.
- Scratch. (2013). Disponível em: <http://scratch.mit.edu>. Acesso em 20/10/2013.
- Silva Filho, R. L. L.; Motejunas, P. R.; Hipólito, O.; Lobo, M. B. C. M. (2007). “A evasão no ensino superior brasileiro”. *Cadernos de Pesquisa*, v. 37, n. 132, p. 641-659.
- Sousa, R. V.; Barreto, L. P.; Andrade, A.; Abdalla, D. (2010). “Ensinando e aprendendo conceitos sobre ciência da computação sem o uso do computador: Computação Unplugged!!!”. *Práticas em Informática na Educação: Minicursos do Congresso Brasileiro de Informática na Educação*, volume 1, número 1.
- SuperLogo. (2013). Disponível em: <http://projetologo.webs.com/slogo.html>. Acesso em 20/10/2013.