

Material didático lúdico: uso da ferramenta Scratch para auxílio no aprendizado de lógica da programação

Manassés Vitorino de Oliveira¹, Luciene Cavalcanti Rodrigues^{1,2}, Ana Paula Garrido de Queiroga³

¹Instituto de Federal de Educação, Ciência e Tecnologia de São Paulo (IFSP)
Campus Votuporanga
Av. Jerônimo Figueira da Costa, 3014 – Pozzobon – Votuporanga, SP - Brasil
Grupo de Pesquisa CNPq: Tecnologias e práticas inovadoras aplicadas ao ensino

²Faculdade de Tecnologia de São José do Rio Preto (Fatec)
Centro Estadual de Educação Tecnológica Paula Souza
Rua Fernandópolis, 2510 – Eldorado – S. J. Rio Preto, SP – Brasil

³ União das Faculdades dos Grandes Lagos – UNILAGO

manavitorino@gmail.com, prof.luciene@ifsp.edu.br, anaproj.eng@gmail.com

Abstract. *The concept of programming logic is essential for most computer related courses, which is logical reasoning to create a sequence of steps to achieve a certain objective. According to surveys conducted with students in technical and technological courses computer area of the different institutions it is clear that the algorithm discipline and programming are cited as the most difficult, also presented high dropout and failure rate. Within this context, this paper aims to demonstrate the application of alternatives to facilitate student learning to reduce the failure and dropout. As an initial alternative was developed and tested material for the development of logical reasoning based on programming games and animations through blocks (Scratch)*

Resumo. *O conceito de lógica de programação é essencial para a maioria dos cursos da área de informática, que representa o raciocínio lógico para a criação de uma sequência de passos para atingir determinado objetivo. De acordo com pesquisas realizadas com alunos ingressantes em cursos técnicos e tecnológicos da área de informática de diferentes instituições é evidente que as disciplinas de algoritmo e programação são citadas como as mais difíceis, apresentado também alto índice de evasão e reprovação. Dentro deste contexto, este trabalho tem por objetivo demonstrar a aplicação de alternativas para facilitar o aprendizado do aluno que visam reduzir a reprovação e a evasão. Como alternativa inicial foi desenvolvido e testado material para desenvolvimento de raciocínio lógico baseado em programação de jogos e animações através de blocos (Scratch).*

1. Introdução

Segundo Mendonça Neto (2013) a lógica de programação é uma das mais importantes disciplinas para os alunos que ingressam em cursos na área de tecnologia da informação pois o desempenho do aluno na referida disciplina vai influenciar diretamente seu

desempenho durante o curso visto que tais disciplinas exigem dos alunos raciocínio lógico, resolução de problemas e capacidade de abstração da solução em uma representação formal e/ou em uma linguagem computacional (Jesus e Brito 2010).

Visando realizar um levantamento quantitativo a respeito das reprovações em cursos de tecnologia da informação, foi realizada pesquisa eletrônica com mais de 100 pessoas graduadas e graduandos. Na Figura 1 é possível verificar que quando perguntados sobre as reprovações em disciplinas, 42% dos alunos responderam que já reprovaram na disciplina de linguagem de programação, lógica da programação ou algoritmos, seguido por uma grande quantidade de reprovações em Banco de dados, que é uma disciplina correlacionada e que exige muito raciocínio lógico.



Figura 1. Quantidade de reprovações (Fonte: Pesquisa)

Para Pereira Junior et al (2005) o desenvolvimento de algoritmos nos cursos da área de computação é uma das competências mais difíceis de serem trabalhadas. Monteiro Neto (2013) considera que a maior dificuldade enfrentada em um curso na área da computação é ensino de algoritmos para ingressantes, este problema é amplificado quando se considera que as competências desenvolvidas neste período serão um dos fatores determinantes para o fracasso ou sucesso do aluno ao longo do restante do curso.

Por seu elevado índice de reprovação e desistência em instituições de ensino no Brasil, a disciplina de algoritmos tem sido motivo de grande preocupação para os professores, motivados na busca de estratégias para a melhoria no processo de ensino e da qualidade das aulas oferecidas aos alunos, mostrando assim a necessidade de algumas alterações tanto na parte didática como na metodológica da apresentação dos conceitos (Pereira Junior et al, 2005). Como mostra a Figura 2, muitos professores já utilizam materiais complementares como forma de ajudar no aprendizado do aluno, tendo em vista que os respondentes citaram que cerca de 35% dos professores utilizam vídeos e materiais complementares, bem como ferramentas como Greenfoot, Alice e Scratch (36%) que auxiliam o desenvolvimento de raciocínio lógico com a criação de jogos.



Figura 2. Uso de material (recursos) pelos professores (Fonte: Pesquisa)

Muitos educadores se empenham em experimentar novas metodologias ou ferramentas de possam minimizar este problema que está presente principalmente nos dias atuais tendo em vista que os alunos ingressantes no ensino médio e superior trazem posturas inadequadas para este tipo de aprendizagem, seja esta no ensino médio integrado ao técnico ou no ensino superior. Por exemplo, a maioria dos alunos traz consigo o medo de errar, atrapalhando o desenvolvimento de exercícios que precisam de inferências e experimentações para chegar ao resultado satisfatório.

Este projeto tem por objetivo criar material didático (apostila e vídeos) para uso em disciplinas como Algoritmos e Programação de Computadores dos cursos técnicos e superiores que demonstre os conceitos de lógica utilizando a ferramenta Scratch que possa ser utilizada em aulas práticas antes do primeiro contato do aluno com as linguagens de programação comerciais, almejando diminuir o índice de reprovação em tais disciplinas, bem como disponibilizar material didático a professores e alunos de maneira gratuita.

2. A dificuldade do aluno no aprendizado de lógica de programação

De acordo com pesquisas realizadas com alunos e ex-alunos dos cursos da área de informática ficou evidente que disciplinas ligadas à programação possuem alto índice de dificuldade e de reprovação. Segundo Friedrich et al (2012) a lógica em geral trata-se da correção do pensamento e a arte de pensar corretamente. Visto que a forma mais complexa do pensamento é o raciocínio, a lógica estuda ou tem em vista a “correção do raciocínio”. Tendo em vista o pensamento do autor, a falta de organização do pensamento para resolução de um determinado algoritmo leva muitos alunos a terem dificuldade e reprovarem na disciplina de lógica de programação e em consequência desistirem do curso. A lógica da programação é a técnica do uso correto do raciocínio para atingir um determinado objetivo, que resolve de forma ordenada a solução do problema, extremamente necessária para o desenvolvimento de programas, permitindo a definição de uma sequência lógica, ou seja, de passos a serem executados até atingir um determinado objetivo ou solução de um problema. A lógica de programação é parte do raciocínio lógico empregado no desenvolvimento de programas de computador, fazendo uso ordenado de elemento básico suportado por um dado estilo de programação. (Santos Junior 2009, p.8)

Segundo Santos Junior (2009) uma boa lógica de programação é desenvolvida a partir de um conjunto de elementos entre eles: Organização, Criatividade, Perseverança, Padronização e Otimização, desta forma, acredita-se que o uso do Scratch e o desenvolvimento de jogos que utilizam os mesmos conceitos e estruturas utilizadas para o desenvolvimento de algoritmos comerciais é de grande valor em séries iniciais dos cursos de tecnologia da informação, podendo desmistificar os conceitos e trazer maior engajamento dos alunos.

3. Scratch uma ferramenta para auxílio no aprendizado de lógica de programação x Visualg

Segundo Mendonça Neto (2013) o Scratch é uma linguagem gráfica de programação desenvolvida no MIT (*Instituto de Tecnologia de Massachusetts*) baseada em outras linguagens como o Logo e o Squeak, porém tem a proposta de ser mais simples, de fácil manuseio e intuitiva. Os projetos desenvolvidos no Scratch (Figura 3) podem ser compartilhados no site do próprio MIT (scratch.mit.edu) de forma muito simples, bastando apenas um pequeno cadastro. Após o upload, a execução do programa no próprio navegador (sem a presença do Scratch) fica disponível para todos, ou então, apenas para os demais usuários cadastrados.



Figura 3 - Tela inicial da ferramenta Scratch

Por possuir uma IDE (Ambiente Integrado para Desenvolvimento de Software) em que não é preciso digitar funções ou endereços, o Scratch caracteriza-se como uma linguagem de programação visual, tendo como ponto inovador o fato de usar novos paradigmas de programação, tornando o aprendizado mais fácil e mais divertido, contando com o uso de blocos gráficos para criar programas (Figura 4).

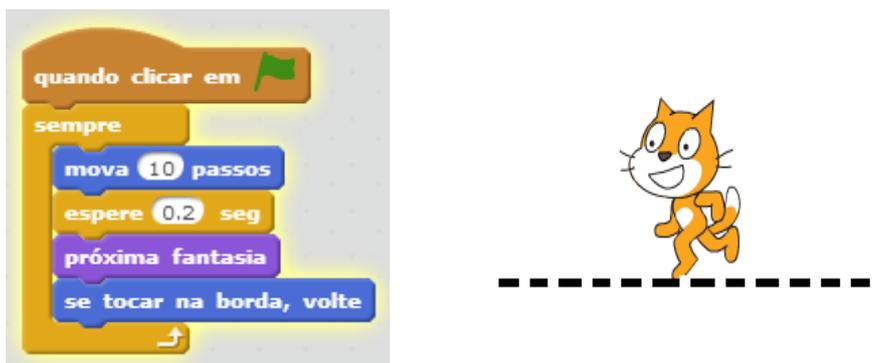


Figura 4. a) Representação bloco no Scratch b) Palco de execução

A ferramenta Scratch possui sintaxe que se assemelha com a lógica de programação, permitindo a criação de algoritmos básicos até jogos completos. Devido a esta estrutura, a ferramenta ajuda os alunos a aprender os principais conceitos de programação, tais como estruturas de decisão e repetição, variáveis, operadores e funções, tornando a aprendizagem de lógica de programação mais intuitiva e visualmente mais agradável, pois o próprio ambiente é voltado para computação criativa e design. Desta forma o aluno precisa somente se concentrar na organização dos blocos (Figura 5a), que não é o caso do desenvolvimento de algoritmo com a ferramenta VisualG (Figura 5b), na qual é necessário saber toda sintaxe da ferramenta e possuir experiência para construir um algoritmo com grande facilidade.

O Visualg é um aplicativo utilizado na maioria dos cursos de informática para ensino de algoritmo, com este aplicativo você pode digitar, executar e depurar o pseudocódigo para resolver problemas propostos, fornecendo também aos professores vários recursos didáticos para exemplificar os principais conceitos de lógica e aplicação prática de variáveis, estrutura de condição, repetição, funções, vetores e matrizes, sendo possível simular um ambiente de programação por meio do teste do pseudocódigo, similar ao DevC++ ou NetBeans (Souza, 2009).



Figura 5. a) Algoritmo no Scratch

b) Algoritmo no Visualg

Ao analisar as duas estruturas apresentadas na Figura 5 é evidente que arrastar e soltar os blocos de código é muito mais fácil inicialmente que a aprendizagem e escrita dos comandos em pseudocódigo, que usa uma codificação sensível a letras maiúsculas e minúsculas, podendo gerar erros de compilação, fato que não ocorre no Scratch.

Mélo et al (2011) destacam o Scratch como uma excelente ferramenta para o ensino de conceitos de Lógica de Programação por possuir uma interface visual amigável e simples, levando de forma intuitiva às principais estruturas de uma linguagem como: operadores (Figura 6a), variáveis (Figura 6b), estruturas de decisão (Figura 6c) e de repetição (Figura 6d), vetor (Figura 6e) e função (Figura 6f).

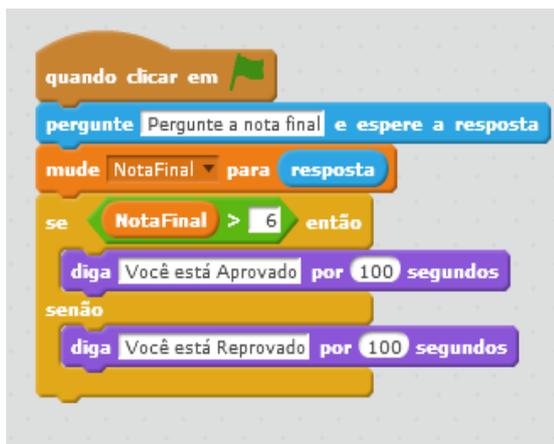
Como informa Almeida (2008 p.36) os operadores são meios de compararmos, incrementamos, decrementamos e avaliamos dados dentro do computador. No Scratch existem três tipos de operadores: os relacionais, os aritméticos e os lógicos como ilustra a Figura 6a. Conforme Fernandes (2010) variáveis são endereços de memória destinados a armazenar informações temporariamente, que também são utilizadas no Scratch (Figura 6b). Todo algoritmo ou programa deve ter variável, existem as variáveis de entrada, que armazenam informações fornecidas por um meio externo (geralmente pela digitação do usuário) e as variáveis de saída, que utilizam os dados processados como resultado.



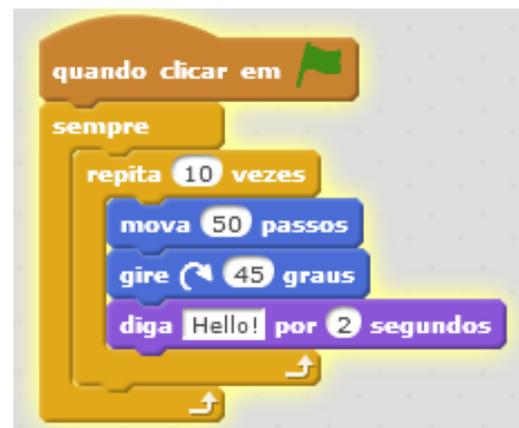
Figura 6 - a) Operadores no Scratch



b) Variáveis no Scratch



c) Estrutura de decisão no Scratch



d) Estrutura de repetição no Scratch



e) Vetores no Scratch



f) Funções (blocos) no Scratch

A estrutura de decisão, conhecida como seleção ou condição, é usada para tomar uma decisão e aceita escolher um conjunto de ações para ser executada a partir do resultado de uma condição (Figura 6c). Esta estrutura é representada por uma expressão lógica, sendo utilizada quando existe a necessidade de testar alguma condição em função de tomar alguma atitude. (Almeida, 2008, p.52)

As estruturas de repetição permitem que uma ou mais instruções sejam executadas por uma quantidade de vezes definida pelo algoritmo ou pelo usuário. As execuções repetitivas, também chamadas “*loops*”, são controladas por estruturas de repetição fornecidas pelas linguagens de programação. Como mostra a Figura 6d, é possível utilizar uma estrutura de repetição no Scratch, onde os blocos de comandos serão repetidos 10 vezes, movendo o personagem, girando-o no ângulo de 45 graus e dizendo (“Hello!”) a cada vez.

Um vetor, do ponto de vista de informática, é uma estrutura de dados homogênea, isto é, todos os elementos de um vetor são do mesmo tipo. A estrutura como um todo é identificada por um nome e um elemento na estrutura por um índice, esta estrutura também pode ser exemplificada no Scratch (Figura 6e).

4. Metodologia

Para este projeto foi realizado levantamento das principais fontes bibliográficas utilizadas por professores da área em suas aulas, bem como análise dos planos de ensino de duas instituições de ensino técnico e superior. Baseado neste levantamento foi desenvolvido o material didático com o conteúdo de lógica de programação e aplicado com os alunos da disciplina de Programação Estruturada da turma do 1º. Semestre do curso de Tecnologia em Análise e Desenvolvimento de Sistemas e do 1º. Ano do curso Técnico em Informática integrado ao Ensino Médio do Instituto Federal de Educação, Ciência e Tecnologia de São Paulo campus Votuporanga, totalizando 40 alunos. A ideia principal do projeto foi mostrar aos alunos a existência de uma ferramenta que possibilite auxílio no aprendizado de algoritmos, desta forma, antes da explicação da programação utilizando linguagem C++ eram realizados testes com a ferramenta Scratch e o material desenvolvido.

Como metodologia em sala de aula foram montados slides com os principais conceitos de algoritmos junto com a ferramenta Scratch. Para os alunos do ensino superior, a aplicação do slide durou quatro aulas com introdução, teoria e prática. Nas duas primeiras aulas foi realizada a introdução teórica abordando as principais vantagens do Scratch em relação ao desenvolvimento de algoritmo. Na segunda etapa foi proposto o desenvolvimento de um pequeno algoritmo como exercício e foi possível nota que com a pequena experiência adquirida os alunos foram capazes de solucionar o desafio proposto. O processo seguiu-se durante o semestre, tornando a aprendizagem mais lúdica e prática, desenvolvendo jogos antes da criação dos programas utilizando-se DevC++ e linguagem de programação C++ no curso superior. Já com os alunos do curso técnico integrado ao ensino médio, a cada novo conceito introduzido havia uma aula com o desenvolvimento de um jogo utilizando a aplicação deste conceito no Scratch. De acordo com as entrevistas os alunos participantes relataram que desta forma o conceito é assimilado mais facilmente, restando depois apenas a transferência da lógica para a linguagem de programação, seja ela qual for.

Dentre os jogos desenvolvidos por alunos está o labirinto, contendo a contagem de vidas, pontos, uso de movimentação por meio de setas, bloco de comandos sempre, se – então, uso de variáveis e diversos personagens (Figuras 7 e 8).

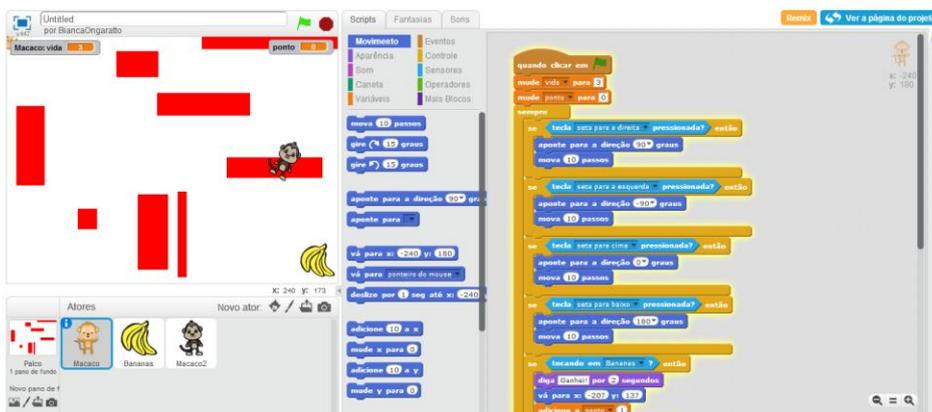


Figura 7. Jogo de Labirinto (Aluna Bianca Ongarotto)

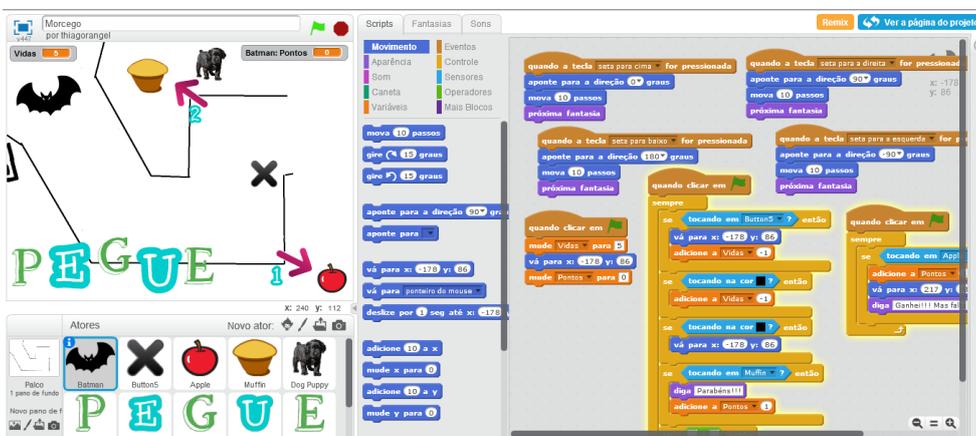


Figura 8. Jogo de Labirinto (Aluno Thiago Rangel)

5. Conclusão

De acordo com as pesquisas e relatos dos alunos, as disciplinas que possuem maior grau de dificuldade são as que envolvem lógica e programação, acarretando também, maior índice de reprovação em cursos técnicos e superiores da área de informática. Nota-se também que poucos professores utilizam materiais diferenciados, tornando então o aprendizado monótono e baseado em leitura, testes e avaliações. Após a aula aplicada foi possível detectar que os alunos realizaram as atividades com sucesso desenvolvendo algoritmos na ferramenta Scratch, sendo possível também entender os principais conceitos de algoritmo e aplicar no Scratch, onde notadamente evidenciou-se que o interesse por descobrir novos comandos e novas ferramentas foi aumentando gradualmente com o decorrer do curso.

Segundo relatos dos alunos de ambos os cursos, a aprendizagem do conceito de variáveis, uso de estruturas condicionais e laços de repetição tornou-se bem mais fácil devido à estrutura visual do ambiente do Scratch, visto que os movimentos dos personagens precisam ser programados de forma a repetir-se muitas vezes, a contagem de pontos, vidas e emissão de sons também só é possível com uso de variáveis e estruturas de se – então – senão, além é claro de fixação dos conceitos relativos à direção, ângulos, operadores matemáticos, relacionais e lógicos.

Nota-se nos exemplos das figuras 7 e 8 que os alunos possuem liberdade para a criação dos jogos, não havendo padrão específico de programação entre os códigos dos personagens de cada aluno, porém ambos atingiram o objetivo da atividade, que era aprender a trabalhar com o laço de repetição “sempre”, as variáveis e a estrutura de decisão “se”. Em relação ao conteúdo da disciplina, notou-se que é completamente possível o uso da ferramenta Scratch em conjunto com a ferramenta VisualG, sem atrapalhar o andamento do curso e o cumprimento do plano de ensino, adicionando-se ainda maior diversão, espontaneidade e participação dos alunos envolvidos. No tocante às notas, para as referidas turmas foi obtido um índice de 92% de aprovação na turma do ensino médio e 85% na turma de ensino superior, obtendo assim um aumento na aprovação de 5% em relação à média de aprovações em anos anteriores. Evidencia-se que no tocante à análise dos resultados o aumento de 5% é um resultado simples quando comparado apenas quantitativamente, porém em relação ao grau de satisfação, confiança, o aumento da criatividade, liberdade de expressão e satisfação dos alunos com o curso foi possível evidenciar um aumento considerável, levando até à maior permanência dos alunos em horário extra aula e participação em eventos.

A apostila desenvolvida está disponível gratuitamente em www.luciene.pro.br e pode ser utilizada por professores, alunos e a comunidade em geral. Trabalhos futuros estão sendo implementados agora com acompanhamento anual dos alunos do primeiro ano do ensino médio integrado ao técnico em informática, onde foram adicionados o uso dos aplicativos do Code.org, desenvolvimento de jogos com Scratch em sala de aula e como trabalhos extras, uso da ferramenta AppInventor para a criação de aplicativos móveis e do software Flowgorithm para a criação de fluxogramas. O que pode ser notado até o momento tomando-se como base uma turma de 20 alunos que utilizam o material desenvolvido é que 3 alunos tiveram rapidamente a percepção de que não gostam da área e pediram transferência no primeiro bimestre, há uma média de 5 alunos que possuem média bimestral abaixo de 6,0 e que possuem certa dificuldade com a

expressão de comandos utilizando pseudocódigo mas quando submetidos ao desenvolvimento com a ferramenta Scratch não apresentam dificuldade de raciocínio.

Referências

- ALMEIDA, M. (2008). Curso essência de lógica da programação. Digerati Book São Paulo.
- FERNANDES, N.M.M, FERNANDES, W. L. Lógica De Programação: Pseudocódigo. Clube dos Autores, 2010.
- 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: <<http://www.br-ie.org/pub/index.php/sbie/article/view/1762>> Acesso em: 18 maio 2015.
- MÉLO, F. E. N., et. al (2011). Do Scratch ao Arduino: Uma proposta para o ensino introdutório de programação para cursos superiores de tecnologia. XXXIX Congresso Brasileiro de Educação em Engenharia, Blumenau, SC. Disponível em <<http://www.abenge.org.br/CobengeAnteriores/2011/sextoestec/art1886.pdf>>. Acesso em: 14 maio 2015.
- MENDONÇA NETO, V. S. (2013). A utilização da ferramenta Scratch como auxílio na aprendizagem de lógica de programação. Anais do II Congresso Brasileiro de Informática na Educação (CBIE 2013). Workshops (WCBIE 2013) . Disponível em: < [br-ie.org/pub/index.php/wcbie/article/viewFile/2675/2329](http://www.br-ie.org/pub/index.php/wcbie/article/viewFile/2675/2329)> Acesso em: 18 maio 2015.
- PEREIRA JUNIOR, R.C.J. et al (2005). Ensino de Algoritmos e programação: Uma experiência no nível médio. XXV Congresso de Sociedade Brasileira de Computação, XIII Workshop de informática na escola. São Leopoldo, RS. Disponível em <<http://www.lbd.dcc.ufmg.br/colecoes/wei/2005/008.pdf>> Acesso em: 18 maio 2015.
- SANTOS JUNIOR, C. I. (2009) Introdução a Lógica de Programação. Clube do Autores.
- SOUZA, C. M. (2009). Visualg -Ferramenta de Apoio ao Ensino de Programação. Revista TECCEN – volume 2 – número 2 - setembro de 2009 – ISSN 1984-0993. Disponível em <<http://www.uss.br/pages/revistas/revistateccen/V2N22009/ArtigoVisuAlgSOUZA.pdf>> Acesso em: 14 maio 2015.