



ANÁLISE DO SCRATCH COMO FERRAMENTA DE AUXÍLIO AO ENSINO DE PROGRAMAÇÃO DE COMPUTADORES

Pryscilla de Sousa Pereira – pryscillasousa@yahoo.com.br

Marcos Medeiros – marcosmedeiros31@gmail.com

José Wally Mendonça Menezes – wally@ifce.edu.br

Instituto Federal de Educação, Ciência e Tecnologia do Ceará (IFCE)

Endereço: Av. 13 de maio, Benfica, 2081

CEP: 60.050-531 – Fortaleza – CE

Resumo: *A disciplina de Algoritmos e Lógica de Programação tem apresentado um alto índice de reprovação dos alunos nos cursos em que está inserida. Os estudos sobre o baixo rendimento dos alunos nesta disciplina dão pistas das dificuldades encontradas por estes na criação de soluções computacionais dos problemas apresentados, por meio de algoritmos estruturados ou de alguma linguagem de programação. Este artigo tem por objetivo, mostrar uma ferramenta para auxiliar na didática dessa disciplina tornando-a mais atrativa, intuitiva e de fácil adaptação. Essa ferramenta é um ambiente de programação, conhecido como Scratch, que possui toda a estrutura básica e necessária de controle de fluxo para o aprendizado dessa disciplina.*

Palavras-chave: *Algoritmo, Scratch, Linguagem de Programação, Ensino.*

1. INTRODUÇÃO

Um dos maiores problemas das instituições de ensino superior brasileiras é o alto índice de evasão. Os cursos na área de Ciências, Matemática e Computação tem uma taxa em torno dos 28% de evasão. Esse valor é considerado alto, já que a média nacional é de 23%, segundo dados divulgados pelo INEP em 2005 (BRASIL, 2012). Nota-se que as disciplinas relacionadas a algoritmos ou linguagem de programação são umas das razões pela qual há evasão e reprovação nas primeiras fases dos cursos de informática (CABRAL, 2007).

O objetivo da disciplina citada é fazer o aluno compreender a forma de “raciocínio” do computador e dominar as técnicas para solucionar problemas propostos. Estas soluções podem ser um programa de computador ou só um pseudocódigo – o Algoritmo. Um algoritmo é um conjunto finito de comandos que resolverão um problema proposto (FARRER, 1989).

A dificuldade de adaptação dos alunos em desenvolver raciocínio lógico quando estão acostumados a decorar o conteúdo, e a falta de motivação do aluno gerada pelo despreparo, podem ser citadas como motivos para a desmotivação, reprovação ou evasão do aluno (GOIS & PINHO, 2007). A ausência de uma base lógico-matemática implica no aluno chegar despreparado na universidade, tornando o aprendizado em programação bastante complexo, chegando a ser desmotivador (DIJKSTRA, 1989). Além disso, em sua grande maioria, os cursos superiores na área de engenharia, segundo a própria ABENGE – Associação Brasileira de Ensino de Engenharia, são “baseados em conhecimento, com enfoque no conteúdo e



centrado no professor” (ABENGE, 1991). Faz-se necessária, então, uma mudança nos métodos de ensino e aprendizagem que permita desenvolver as competências necessárias de um bom profissional.

Portanto, apresenta-se como objetivo desse artigo mostrar que o Scratch pode ser usado no início da disciplina de Algoritmos e Lógica de Programação. Resumidamente, essa ferramenta não possui “regras de implementação”, em outras palavras, ela é voltada apenas para a lógica do programa. Dessa forma, o aluno pode desenvolver o raciocínio matemático antes de se deparar com a sintaxe da linguagem que o professor pretende ensinar.

2. FUNDAMENTAÇÃO TEÓRICA

O aluno que está no ensino de informática tem muitas dificuldades durante a aprendizagem. Em comparação com outros cursos, podemos observar: um aluno de engenharia civil, quando se depara com o primeiro projeto a construir, a exemplo de uma casa, tem que formar seus conceitos e significados sobre técnicas de planejamento, cálculos, desenhos e construção, mas já passou pela experiência de ter entrado, morado e usufruído de uma residência e possui os principais conceitos sobre o objeto real: sabe o que são portas, janelas, pias, lâmpadas e paredes. Já um aluno de medicina, ao entrar para a faculdade, muito provavelmente, já teve a experiência de ter entrado em consultórios médicos para se consultar e sabe que é preciso haver uma consulta, e que dessa consulta provê um diagnóstico, que pode requisitar um ou mais exames, que podem resultar no uso de medicamentos e em consultas de retorno. No entanto, um aluno de Engenharia da área de Tecnologia da Informação, ao entrar para o curso técnico ou superior, contudo, não tem noção do que seja um sistema, uma linguagem de programação, ou um banco de dados, a não ser que já trabalhe na área de informática ou tenha feito algum tipo de curso de extensão.

A disciplina de Lógica de Programação ou Algoritmos é uma das mais importantes para os alunos que ingressam em diversos tipos e níveis de ensino, tais como os cursos superiores regulares, bacharelados, tecnólogos, cursos técnicos, e de extensão. Estes cursos, geralmente, são da área de Informática. Entretanto, o mesmo conteúdo, ou parte dele, é ministrado em cursos da área de Engenharia, Ciências Exatas e outras. Essa importância dar-se, pois os conhecimentos adquiridos nela vão influenciar diretamente no desempenho das demais disciplinas correlatas durante o restante do curso (RAABE *et al*, 2007).

O objetivo geral da disciplina é permitir que o aluno desenvolva o raciocínio lógico aplicado a solução de problemas em nível computacional, além de introduzir os conceitos básicos de desenvolvimento de algoritmos, de forma a propiciar aos alunos uma visão crítica e sistemática sobre resolução de problemas e prepará-los para a atividade de programação.

Os cursos de lógica de programação iniciam com uma média de 50 alunos e em poucos meses, constata-se que a taxa de reprovação, ou desistência, chega a 60% (ROCHA, 1993). Além disso, foi constatado que nessa disciplina o nível de aprendizagem é muito baixo, conforme diversos pesquisadores denunciam (SILVA *et al*, 2009) (SOUTO & DUDUCHI, 2011). Há vários fatores que podem levar a essa baixa aprendizagem, como uma base matemática precária, limitações de ensino do professor, material de apoio precário, entre outros.

Tornar o processo ensino-aprendizagem mais interessante é um desafio enfrentado pelo professor em qualquer nível de ensino. É necessário que os professores que trabalham com as disciplinas de algoritmos busquem soluções para minimizar o número de reprovações ou abandonos. Uma das formas de sanar esse problema é usar ferramentas ou ambientes



facilitadores que provoquem um aprendizado substancial das atividades didáticas.

3. SOFTWARES RELACIONADOS

A utilização de novas ferramentas, geralmente, tende a fazer com que as aulas tornem-se mais atrativas e participativas. Hoje há diversas ferramentas disponíveis, que visam facilitar o ensino da disciplina de algoritmos. Algumas delas são listadas abaixo.

- WebPortugol : criado pela Universidade do Vale do Itajaí, essa ferramenta, como o próprio nome sugere, permite construir online programa em português estruturado ou Portugol.
- Visualg: criado pela "Apoio Informática Ltda - Consultoria e Desenvolvimento de Sistemas", o Visualg é um "programa para edição e interpretação de algoritmos" em português estruturado ou português.
- ASA - Ambiente de Simulação e Animação de Algoritmos: desenvolvido pela equipe do SENAC. O ASA também conhecido como Construtor, é um software para criar algoritmos usando fluxogramas (MAYERHOFER,2012).
- LOGO: desenvolvida por pesquisadores do MIT-USA. É uma linguagem considerada bastante sofisticada, por possuir características pertencentes a 3 paradigmas computacionais distintos: procedural, orientado a objetos e funcional (GREGOLIN, 2012).
- Alice: programa desenvolvido pela Universidade Carnegie Mellon nos Estados Unidos. Alice é um ambiente de programação tridimensional de fácil utilização no qual podem ser criadas animações e interações entre personagens e objetos lembrando muito jogos de vídeo game (ALICE, 2012).
- SCRATCH: essa linguagem resulta de uma parceria entre a Portugal Telecom e a universidade MIT de Boston. Scratch permite a criação de histórias, animações, jogos e outras produções. Tudo pode ser feito a partir de comandos prontos que devem ser agrupados (SCRATCH, 2012).

Mesmo com diversos modelos, não há um software ideal, por seu uso depende da metodologia do professor e do nível de conhecimento e velocidade de aprendizado da turma. Nesse artigo iremos dar ênfase no uso do Scratch, tendo em vista que ele é um software gratuito, de fácil compreensão, que faz uso de um ambiente divertido, intuitivo e de fácil operação.

Como já foi dito anteriormente, o objetivo desse artigo é mostrar o uso do Scratch como uma ferramenta para auxiliar no início da disciplina de lógica de programação. Os alunos ingressantes nos cursos, de exatas, nasceram na era da informação, tem um contato estreito com diversas mídias, e seu raciocínio não é linear. Além de, em sua grande maioria, não terem o hábito de ler, nem de dedicar algumas horas do seu dia aos estudos e fazem parte da geração dos jogos. Assim o uso do Scratch seria ideal, já que o mesmo não possui nenhuma sintaxe, deixando o aluno focado apenas no aprendizado da lógica do programa, após o raciocínio lógico bem absorvido, o professor poderá iniciar o aprendizado de alguma linguagem.



4. SCRATCH – VISÃO GERAL

O Scratch é uma nova linguagem gráfica de programação criada no Media Lab do Instituto de Tecnologia de Massachusetts, inspirada nas linguagens Logo e Squeak, mas que pretende ser mais simples, fácil de utilizar e mais intuitiva (SCRATCH, 2012). Ela foi compartilhada com o mundo em 15 de maio de 2007. É um software gratuito, possui uma IDE onde não é preciso digitar funções, endereços, etc. Seu objetivo primário é facilitar a introdução de conceitos de matemática e de computação, enquanto também induzindo o pensamento criativo, o raciocínio sistemático e o trabalho colaborativo (SCRATCH, 2012)

O termo Scratch provém da técnica de *scratching* utilizada pelos Disco-Jockeys do Hip-Hop que giram os discos de vinil com as suas mãos para frente e para trás de modo a fazer misturas musicais de forma criativa e inesperada. Com o Scratch é possível fazer algo de semelhante, misturando diferentes tipos de trechos de mídia (gráficos, fotos, músicas, sons) de formas criativas. Veja a tela inicial da ferramenta na Figura 1.

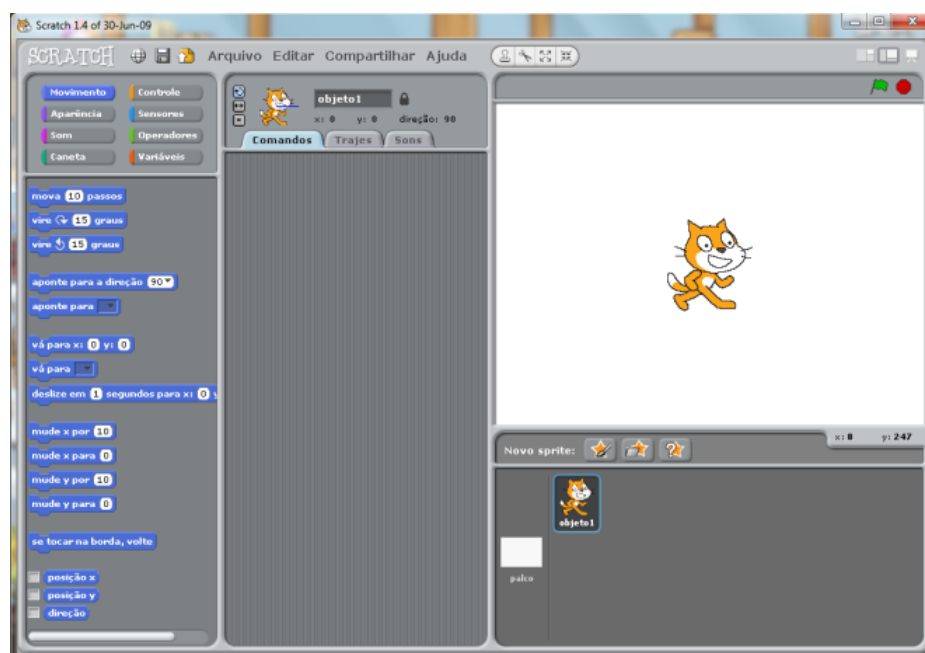


Figura 1 – Tela inicial do Scratch

A programação do Scratch é feita através de blocos de comandos que são encaixados uns aos outros, formando a sequência de comandos que se deseja. Os blocos são concebidos para se encaixar apenas de uma única forma, fazendo sentido sintaticamente, não ocorrendo, assim, erros de sintaxe. É uma maneira de trazer conceitos de informática de alto-nível para seus usuários, pois os comandos presentes nos blocos são praticamente os mesmos.

Esse software permite a construção de programas que controlam e misturam gráficos, animação, texto, música e som. O mesmo pode interagir com objetos exteriores de vários tipos. A página de internet do Scratch fornece inspiração e audiência, lá se podem experimentar os projetos de outros, reutilizar e adaptar as suas imagens e scripts.

É muito simples se criar jogos, animações, histórias interativas, arte e outros, o que aumenta a interatividade entre o aluno e a linguagem. Apesar de não ser muito robusto, esse



software, possui estruturas básicas, porém importantes, para um programador como estruturas de controle de fluxo.

O software é dividido em 3 blocos, como visto na Figura 1. O primeiro bloco contém os comandos que serão adicionados ao seu programa. Comandos como: de controle, movimentos, operações, aparência, sons e outros. No segundo bloco é onde ficará o seu programa, esta tela conterà os blocos de comandos, os trajés de seus sprites (os desenhos que aparecem na tela) e os sons que acompanham os sprites. E o terceiro bloco contém a tela de animação, onde o programa é executado.

5. APLICAÇÃO

Como é visada à utilização do Scratch como uma alternativa para o ensino de algoritmos e linguagem de programação, então será mostrado um exemplo típico dessa disciplina: a média aritmética. Logo abaixo, será feita a descrição do problema.

O problema consiste de fazer um programa que calcule a média final dos alunos, sendo que os mesmos fizeram 3 provas. Ao apresentar a média final, o programa deverá mostrar a situação do aluno:

- Aprovado: se média maior ou igual a 7,0.
- Recuperação: se maior ou igual a 5,0 e menor que 7,0.
- Reprovado: se menor que 5,0.

Antes de iniciar, é necessário identificar nesse problema quem são os dados de entrada, os dados de saída e o que será processado. Então, para montar o algoritmo proposto, três perguntas são feitas. A primeira corresponde as notas de cada aluno, a segunda refere-se ao processo de cálculo da média e a última define a apresentação da situação do aluno.

a) Quais são os dados de entrada?

R: Os dados de entrada são $N1$, $N2$ e $N3$.

b) Qual será o processamento a ser utilizado?

R: O procedimento será somar todos os dados de entrada e dividi-los por 3 (três).

$$M = \frac{N1 + N2 + N3}{3}$$

Comparação: $M > 7$, aprovado

$M < 7$ e $M \geq 5$, recuperação

$M < 5$, reprovado

c) Quais serão os dados de saída?

R: O dado de saída será a média final e a situação do aluno

Se média ≥ 7 , mostre aprovado

Se média < 7 e média ≥ 5 , mostre recuperação

Se média < 5 , mostre reprovado

Depois de todas essas etapas finalizadas pode-se iniciar a construção e execução do



programa em Scratch. Na Figura 2, é mostrada a inicialização do programa: A caixa laranja é onde todas as variáveis serão declaradas.



Figura 2 - Inicialização do programa.

Na próxima etapa, é feita a entrada dos dados. É nesse momento em que as notas serão digitadas para em seguida serem processadas. Veja Figura 3.

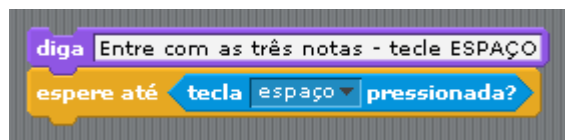


Figura 3 - Entrada de dados

Após, a entrada de dados é possível calcular a média e fazer o tratamento para saber se o aluno passou, ficou de recuperação ou se foi reprovado. Veja Figura 4.

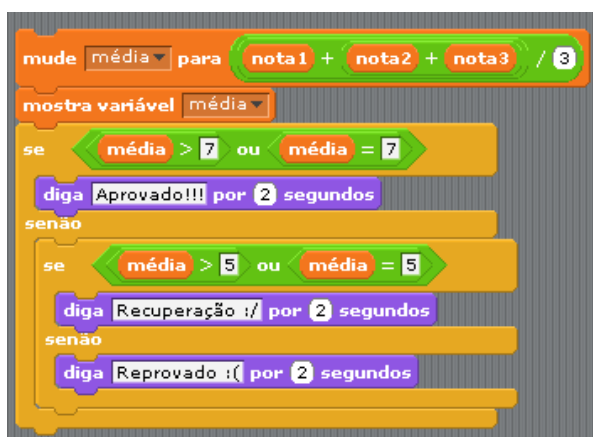


Figura 4 - Processamento de dados.

Na Figura 5 é mostrado o programa por completo, com todas as suas etapas. Logo em seguida, na Figura 6, é demonstrada a execução do programa.

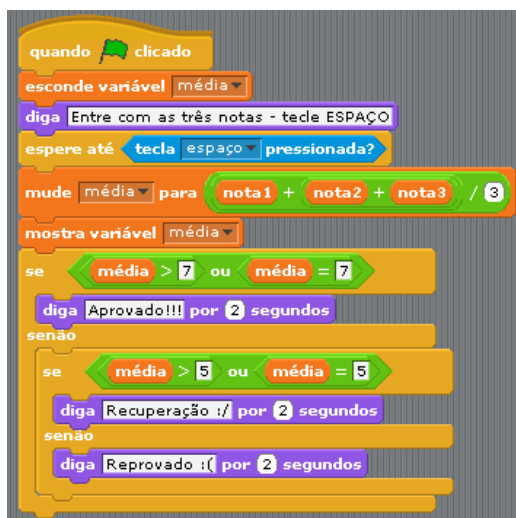


Figura 5 –Estrutura do programa média



Figura 6 – Execução do programa média

6. CONSIDERAÇÕES FINAIS

Ao final desse artigo, tendo apresentado um modo de utilização da ferramenta citada, pode-se perceber que o Scratch é uma ferramenta que pode auxiliar os alunos da disciplina de Linguagem de Programação e Algoritmos que possuem dificuldade em organizar pensamentos, em construir conceitos matemáticos, ou problemas com informática. Com o Scratch a programação torna-se mais divertida e simples, pois ela é efetuada através da criação de sequências de comandos simples, que correspondem a blocos de várias categorias, encaixados e encadeados de forma a produzirem as ações desejadas. O aluno que usar esse software ao início da referida disciplina, terá melhor compreensão dos conceitos de programação como, estruturas de decisão e repetição, variáveis, operadores, etc. Este é um programa inovador, que permite exercitar a criatividade e o raciocínio científico lógico e matemático, tornando o aluno apto a solucionar qualquer problema computacional que lhe seja apresentado.



REFERÊNCIAS

ABENGE - Associação Brasileira de Ensino de Engenharia. **Perfil do Engenheiro do Século XXI**. Brasília, 1998.

ALICE. Disponível em: <<http://www.alice.org/>>. Acesso em: 20 maio 2012.

BRASIL. Ministério da Educação. Instituto Nacional de Estudos e Pesquisas Educacionais. **Sinopses do ensino superior. Censos do ensino superior**. Comunicações pessoais. Disponível em: www.inep.gov.br. Acessado em: 12 jan. 2012.

CABRAL, M. I. C. *et al.* **Perfil dos cursos de computação e informática no Brasil**, XXVII Congresso da SBC - XV WEI, Rio de Janeiro, 2007.

DIJKSTRA, E. W. **On the Cruelty of Really Teaching Computing Science**. Communication of ACM, 1398-1404, 1989.

FARRER, H. **Algoritmos Estruturados**, Editora LTC, p. 23, Rio de Janeiro, 1989.

GOIS, A.; PINHO, A. **Brasil é reprovado, de novo, em matemática e leitura**. Disponível em: <<http://www1.folha.uol.com.br/folha/educacao/ult305u351481.shtml>>. Acesso em: 13 jan. 2012.

GREGOLIN, V. R. **Linguagem LOGO: Explorando conceitos matemáticos**. Disponível em: <<http://tecnologiasnaeducacao.pro.br/revista/a1n1/art8.pdf>>. Acesso em: 20 maio 2012.

MAYERHOFER, M. A. **Um paradigma para o Desenvolvimento de Software Educacional**. Disponível em: <<http://www.senac.br/informativo/bts/222/boltec222d.htm>>. Acesso em: 20 maio 2012.

RAABE, A. L. A.; SANTIAGO, R.; DAZZI, R. L. S. **Adquirindo experiência na construção de ferramentas de apoio a aprendizagem de algoritmos**. Workshop de Ambientes de Apoio a Aprendizagem de Algoritmos e Programação. Simpósio Brasileiro de Informática na Educação, 2007. Anais dos XVIII Simpósio Brasileiro de Informática na Educação, 2007.

ROCHA, H. V. **Representações Computacionais Auxiliares de Conceitos de Programação**. In: "Computadores e Conhecimento: Repensando a Educação". Livro organizado por Valente, J. A. Editora Unicamp, 1993.

SCRATCH. **ABOUT Scratch (Scratch Documentation Site)**. Disponível em: <http://info.scratch.mit.edu/About_Scratch>. Acesso em: 20 maio 2012.

SILVA, I. F. A.; SILVA, I. M. M.; SANTOS, M. S. **Análise de problemas e soluções aplicadas ao ensino de disciplinas introdutórias de programação**. Universidade Federal Rural de Pernambuco, Recife – PE. 03 pp. Disponível em <<http://www.eventosufrpe.com.br/jepex2009/cd/resumos/R1479-1.pdf>>. Acesso em: 12 jan.



2012.

SOUTO, A. V. M.; DUDUCHI, M. **Um processo de avaliação baseado em ferramenta computadorizada para o apoio ao ensino de programação de computadores.** Disponível em <<http://bibliotecadigital.sbc.org.br/download.php?paper=1363>>. Acesso em: 12 jan. 2012.

ANALYSIS OF SCRATCH AS A TEACHING AID TOOL FOR COMPUTER PROGRAMMING

Abstract: *The discipline of Algorithms and Logic Programming has shown a high rate of failing students in the courses in which it operates. Studies about low efficiency of students in this course show the difficulties encountered on the creation of computational solutions of the problems presented by using algorithms or any structured programming language. The objective of this article is to present a tool to assist in teaching this discipline by making it more attractive, intuitive and easy to adapt. This tool is a programming environment, called Scratch, which has all the basic structure and flow control required for learning this discipline.*

Keywords: Algorithm, Scratch, programming language, education.

Realização:

 **ABENGE**

Organização:



**O ENGENHEIRO
PROFESSOR E O
DESAFIO DE EDUCAR**