

Ensinando princípios básicos de programação utilizando jogos educativos em um programa de inclusão digital

Núbia dos Santos Rosa¹, Clevis Elena Rapkiewicz¹

¹UENF-Universidade Estadual Norte Fluminense Darcy Ribeiro

Abstract

This article describes the inclusion of basic programming activities in a digital program inclusion for students that participate in a pre-vestibular community course at UENF (Universidade Estadual Norte Fluminense Darcy Ribeiro) that is, high school graduated students not taking the higher level yet. The initial phase of the program used educational games as a resource. The resource used to introduce algorithms and programming has two objectives: the ludic use of the computer and at the same time motivating students to learn programming. The stage with the use of games showed that regardless the level of digital inclusion, the students felt motivated to follow activities of programming, or at least understand the logic behind the various programs they use.

Keywords: educational games, computational games, algorithms

Authors' contact :

{nubrosa,clevi}@uenf.com.br

1. Introdução

Apesar de vivermos na era da Informação, uma parte da população encontra-se excluída digitalmente. Segundo pesquisa divulgada pelo IBGE[2005] pouco mais de um terço dos estudantes brasileiros (36,3%) acessa a Internet pelo menos uma vez por dia. E a impossibilidade de acesso ao microcomputador foi o principal motivo alegado pelas pessoas que não utilizaram a Internet (37,2%).

De acordo com a definição de Fagundes [2004], a Inclusão Digital não é somente o acesso às tecnologias, mas sua apropriação na resolução de problemas. Ou seja, é fundamental que as pessoas tenham acesso as Tecnologias de Informação e Comunicação (TICs) e saibam como utilizá-las para resolver problemas. Entre os conteúdos que podem ser ensinados através do uso das TICs está um tema da própria área de informática: a lógica de programação envolvendo a capacidade de análise e resolução de problemas e algoritmos. A utilidade deste tipo de conhecimento não é restrita a profissionais de informática. Ao contrário, cada vez mais jovens tornam-se autores de *sites* e também de programas utilizando como recursos

pacotes fechados, sem muitas vezes entender a lógica de programação subjacente a várias partes do código. Isso se torna evidente quando por exemplo um estudante de ensino médio tenta adaptar um código em Java Script de uma página para a sua própria, e esbarra em conhecimentos básicos como o não entendimento da estrutura de um "for" que lhe permitiria mudar a quantidade de opções de um menu. Nesse contexto, entender a estrutura básica de algoritmos é fundamental para desenvolver a competência de ler algoritmos. E, porque não, de escrever os seus próprios? Como, porém, ensinar jovens de nível médio a programarem, sobretudo quando o público alvo tem dificuldades com o uso da própria informática, caracterizando grupo com nível baixo ou apenas médio de inclusão digital?

Na literatura podemos encontrar vários problemas e dificuldades relacionados à disciplina de algoritmos, entre os quais destacamos aspectos de motivação e de desenvolvimento de raciocínio lógico. Entre esses, mencionados por Rodrigues [2002], pode-se destacar: i) dificuldade de adaptação dos alunos desenvolverem raciocínio lógico quando estão acostumados a decorar o conteúdo; ii) falta de motivação do aluno gerada pelo despreparo e o desânimo quando ele acredita que a disciplina constitui um obstáculo extremamente difícil de ser superado. Segundo Pereira e Rapkiewicz [2004] durante o processo de ensino aprendizagem de algoritmos e programação nota-se que grande parte dos alunos apresenta dificuldades em assimilar as abstrações envolvidas, não conseguem desenvolver o raciocínio lógico necessário. Esse tipo de abstração é necessário quando o estudante precisa modificar um código (exemplo citado acima), para o qual o entendimento de como funciona um "for" seria necessário.

É neste contexto que estão sendo utilizados jogos educacionais em um Projeto de inclusão digital na Universidade Estadual do Norte Fluminense Darcy Ribeiro para alunos egressos do ensino médio, para inserir conceitos de programação. A motivação para isso foi dupla: o uso lúdico do computador e, ao mesmo tempo, motivação no aprendizado de programação. Ainda que nem todos os participantes do curso venham a cursar esta disciplina futuramente, entendemos que o desenvolvimento do raciocínio lógico é útil em diversas áreas do conhecimento.

Este artigo está estruturado em 5 seções. A primeira seção é a introdução ao tema do trabalho desenvolvido. A segunda seção apresenta algumas referências em

relação ao uso de jogos. A seção 3 mostra alguns jogos desenvolvidos e na quarta seção vemos como os jogos podem ser utilizados de forma eficiente, através de estratégias pedagógicas adequadas e ainda, como utilizar estas estratégias nos jogos aqui apresentados tornando os úteis para o ensino de algoritmos. E, por último, a seção 5 apresenta as conclusões.

2. Jogos educacionais

Os jogos educacionais são ferramentas que apóiam o processo de ensino-aprendizagem. Segundo Tarouco [2005] os jogos podem ser ferramentas eficientes, pois eles divertem enquanto motivam, facilitam o aprendizado e aumentam a capacidade de retenção do que é ensinado, exercitando as funções mentais e intelectuais do jogador. Algumas características que fazem com que os jogos sejam atrativos são descritas pela autora, como: os jogos possuem regras (fornecem uma estrutura), têm metas (produzem motivação), são interativos (permitem agir), têm resultados e realimentação (favorecem a aprendizagem) e são divertidos (tornam o processo agradável).

De acordo com Johnson [2005] os jogos forçam o aluno a decidir, a escolher, a priorizar e que todos os benefícios intelectuais do jogo resultam dessa virtude fundamental, porque aprender como pensar, em última análise, tem a ver com aprender longo prazo e, então, decidir. Dessa forma, os jogos não servem apenas para entretenimento. Podemos notar uma relação entre programadores e jogadores quando Johnson [2005] ainda compara a maneira como os jogadores pensam com a forma como os programadores pensam quando escrevem um código: uma série de instruções aninhadas com múltiplas camadas.

Pode-se dizer que o algoritmo está presente implicitamente no jogo e no modo como os alunos pensam no jogo, mas os mesmos ainda não conseguem enxergá-lo explicitamente. Através dos jogos os alunos se sentem motivados no desenvolvimento das tarefas e, conseqüentemente, desenvolvem o raciocínio. Isso porque estes jogos agregam um conjunto de elementos multimídia que prendem mais a atenção do que questões em papel.

A simulação é outro fator importante presente nos jogos. A simulação é um benefício que o jogo oferece, uma vez que o aluno consegue visualizar sua ação sobre o jogo. Através da simulação o aluno consegue visualizar os passos necessários para alcançar um resultado quando, por exemplo, visualiza, seleciona e transporta personagens no jogo. O jogo permite tornar real apenas o que seria imaginário no caso de enunciados de problemas apresentados no quadro ou em listas de exercício em papel. O ensino de algoritmos por meio dos jogos digitais pode promover, de forma lúdica, a formação de novas atitudes a fim de diminuir as dificuldades encontradas pelos alunos e facilitar o processo de aprendizagem.

3. O uso dos jogos no desenvolvimento do raciocínio lógico

Com o intuito de utilizar os jogos para auxiliar o desenvolvimento do raciocínio lógico dos alunos tanto para leitura de algoritmos quanto para posterior elaboração de algoritmos simples, alguns jogos educacionais e lógicos foram desenvolvidos. Os jogos possuem a seguinte estrutura: Tela da História (narração de uma história coerente com o jogo), Tela de Regras (mostra as regras a serem seguidas no jogo), Tela Principal (tela do jogo em si), Tela de Perda (aparece ao usuário quando ele não acerta e, possui texto de estímulo), Tela de Vitória (aparece quando o usuário ganha o jogo) e Tela de Créditos (contém os nomes dos integrantes da equipe).

A seguir são apresentadas algumas telas, nomes e a descrição de alguns dos jogos desenvolvidos.



Figura 1: Jogo "Sobrevivência"

- Jogo “Sobrevivência” - O jogador tem que transportar cinco pessoas de um lado para o outro do rio em 30 minutos. A jangada suporta no máximo 2 pessoas. O tempo (12, 8, 6, 3 e 1 minutos) de travessia de cada pessoa é de acordo com o seu peso. No caso de duas pessoas na jangada, o tempo será determinado levando-se em conta o maior peso.

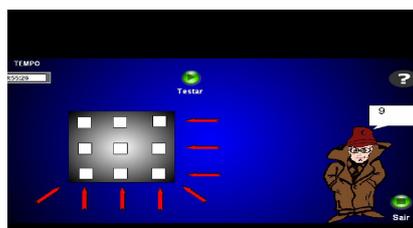


Figura 2: Jogo "Código Secreto"

- Jogo “Código Secreto” - O jogador deve decifrar o código que abre a porta para o detetive entrar na casa. A combinação é formada pela soma das linhas, das colunas e das diagonais. Para o alarme não disparar é necessário que a soma de cada uma das linhas, colunas e diagonais, seja exatamente igual a 15. Todos os algarismos de 1 a 9 devem ser usados, mas sem repetição, senão o alarme dispara.

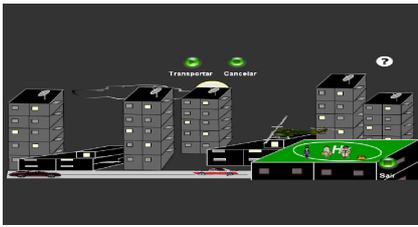


Figura 3: Jogo “A Grande Fuga”

- Jogo “A Grande Fuga”-O jogador deve transportar o chefeão, o piloto, o guarda-costas e o dinheiro para o outro lado da cidade utilizando um helicóptero que suporta no máximo 170 quilos e as três pessoas, inclusive o dinheiro, pesam 270 quilos. Só o guarda-costas pode estar sozinho com o dinheiro.



Figura 4: Jogo "Vasos"

- Jogo “Vasos” - O jogador deve dividir 8 litros de vinho em duas porções de 4 litros. O problema é que existem três vasos, com capacidade de 8, 5 e 3 litros, respectivamente, e os três vasos devem ser usados para fazer a divisão.

4. Estratégias associadas aos jogos

O jogar em si, porém, pode ser um mero divertimento, sem levar necessariamente ao resultado almejado pelo professor. É fundamental, pois, associar ao recurso do jogo um conjunto de estratégias que permita ao professor, como mediador do processo de ensino e aprendizagem, conduzir o aluno a construir sua própria solução e representá-la na forma de algoritmos.

As estratégias associadas são:

1ª estratégia – Problematização, por meio de um problema contextualizado: A apresentação do problema como se fosse uma história, normalmente motiva o aluno, que se sente participante e comprometido com as regras para chegar à solução. Nesta etapa, o aluno pode realizar ações concretas – interagir com a máquina, e abstratas – observar os resultados de suas ações, refletir sobre estes resultados e propor novas soluções, pois há um *feedback* imediato, permitindo ao usuário refletir sobre os erros e fazer novas tentativas. Há, pois, amplo espaço para testar hipóteses e para a experimentação. Se o aluno não consegue resolver o problema, é apresentada uma tela com convite para novas tentativas e, caso contrário, é apresentada uma tela de cumprimentos. Assim, por detrás do aparente “jogar” observa-se o processo de interação, descrito por Piaget [1983] entre o sujeito e o objeto (o próprio problema), onde o

sujeito adapta suas ações de acordo com a resposta observada no objeto.

2ª estratégia – Representação lógica e construção de hipóteses: Tendo vencido o jogo (ou solucionado o problema) o professor solicita que cada aluno escreva a sua solução, ou seja, a seqüência de interações que levaram à vitória. Nas primeiras tentativas, aproxima-se bastante de uma solução em linguagem natural, mas à medida que novos jogos são apresentados, a tendência do aluno é procurar uma forma mais sintética e rápida de fazer estes registros. Observe-se que registrar uma seqüência de ações é, em última instância, escrever um conjunto de passos que leva a um algoritmo, ou ainda, a formalização de um modelo.

3ª estratégia – parte-se agora para a leitura e teste dos modelos, por meio de uma decodificação das representações, testando sua clareza e viabilidade de execução, ou seja, sua lógica. Nesta etapa, solicita-se que os alunos troquem entre si as anotações com os registros feitos na etapa anterior, de modo que cada aluno execute no computador aquilo que outro colega escreveu. O objetivo é, inicialmente, verificar se cada aluno conseguiu escrever algo que um outro colega entenda e, por sua vez, se o aluno consegue entender a representação construída por um outro (e estará, portanto, lendo um algoritmo). A seguir a seqüência é executada em um teste de mesa ou chinês, verificando se é adequada, por meio da experimentação.

4ª estratégia – cooperação para a construção coletiva de uma solução. Após a fase das soluções individuais, o objetivo agora é buscar uma solução que seja de consenso da turma, com a mediação do professor. Após muitas interações entre os alunos, é registrada no quadro a solução para o jogo. Nesse momento, o grupo tende a perceber que a linguagem utilizada deve ser tal que o todo o grupo entenda e que a seqüência de passos deve ser clara, sem ambigüidades. Essa fase permite chegar a uma representação de pseudo-código criada pelo grupo.

5ª estratégia – Como conclusão, o desafio final é construir uma representação na forma de algoritmo. Nessa etapa o aluno precisa fazer com que a sua solução registrada em papel seja entendida e executada pelo computador. A turma é alertada para a necessidade de um outro nível de abstração, que é a tradução do algoritmo para uma linguagem.

As estratégias descritas acima podem ser utilizadas nos jogos aqui apresentados. Vamos utilizar o jogo dos Vasos (figura 5) como exemplo para mostrar como as estratégias podem ser utilizadas. Por exemplo:

- 1ª Estratégia - O jogo possui uma história e suas regras são explícitas. Quando o aluno não consegue resolver o problema surge uma tela com uma frase de estímulo para que o mesmo não desista.
- 2ª Estratégia – Nesta etapa o aluno pode descrever a seqüência do jogo da seguinte forma: Encher vaso 3L; transferir conteúdo de vaso 3L

para vaso 5L; encher vaso 3L com o conteúdo do vaso 8L e assim, sucessivamente;

- 3ª Estratégia – O aluno representa a seqüência de forma mais simbólica: Passo 1- $V8=5$, $V5=0$ e $V3=3$; Passo 2 - $V8=5$, $V5=3$ e $V3=0$; e assim, sucessivamente;

- 4ª Estratégia – registra-se uma solução para o jogo e busca-se uma descrição da solução de modo que todos entendam. Como por exemplo:

Nome: Jogo Vasos

Variáveis: V8, V5, V3 do tipo inteiro;

Início

1. $V8 <= 8$;
2. $V5 <= 0$;
3. $V3 <= 0$;

etc...

Fim.

- 5ª Estratégia – representação de um algoritmo. O professor pode apresentar algumas linguagens e paradigmas da programação. Nos jogos foram desenvolvidos no paradigma OO, na linguagem *Action Script*. Abaixo segue um exemplo de como declarar um vetor em *Action Script*.

```
var vqtde=new Array(3);
vqtde[0]='8';
vqtde[1]='0';
vqtde[2]='0';
```

Nota: vetor em linguagem *Action Script*.

No Quadro1 podemos ver como alguns conceitos de programação podem ser trabalhados nos jogos.

Quadro 1: Conceitos de programação inseridos nos jogos

Jogos	Variáveis e tipos de variáveis	Atribuição de valores	Decremento e incremento.	Estrutura de seleção, condicional, operadores lógicos
Sobrevivência	1 = variável P1: inteiro;	P1 recebe 3, ou P1=3;	Variável Tempo recebe -3 ou $T=T-3$;	Se P1 atravessar então $T=T-3$;
Código secreto	quadrado l= variável Q1:inteiro;	Linha 1 recebe Q1,Q2e Q3 ou $L1= Q1+Q2 +Q3$;	Somar todas as linhas, colunas e diagonais.	Se $L5 < 5$ então jogador PERDE. Nota:L5 como quadrado do meio.
A Grande Fuga	piloto = variável P: inteiro;	P recebe 70, ou $P=70$;	Peso Total recebe 70 ou $PT=PT +70$;	Se P fica sozinho com o dinheiro então jogador PERDE.
Vasos	vaso 5 litros = variável V5:inteiro;	V5 recebe 5, ou $V5=5$;	Transporto V8 para V5 então $V8 \% V5=3$; $\% =$ resto da divisão	Se $V8=4$ e $V5=4$ então jogador VENCE.

5. O uso com alunos de ensino médio

Um dos aspectos na utilização dos jogos é o fator motivador. Convém lembrar que os jogos propostos

têm como objetivo desenvolver o raciocínio lógico indispensável à construção de algoritmos e programação. A verificação da funcionalidade da aplicação dos jogos integrados as estratégias descritas acima foi feita com um grupo de 19 alunos egressos do ensino médio em um programa de inclusão digital na UENF. A idade dos alunos varia dos 16 aos 22 anos, sendo que a maior parte dos alunos têm 17 anos. Destes alunos, apenas metade tem algum conhecimento mais próximo de informática, mas ainda assim restrito ao uso de microcomputadores. Os jogos foram utilizados em paralelo com aulas de reforço de matemática, física, química e biologia. Após o uso dos jogos, o grupo aprendeu a ler código JavaScript, fazendo pequenas adaptações que implicassem em entender a lógica do código analisado. A evasão do curso foi grande: ainda que todos alunos tenham passado pela etapa de uso dos jogos, apenas 04 chegaram a etapa de análise de código, o que não permite maiores conclusões. Por outro lado, a constatação de que dois concluintes com muita dificuldade inicial para usar o teclado, conseguiram analisar código e fazer alterações simples mostra que o caminho utilizado pode dar bons resultados. Os outros alunos, já mais familiarizados com o uso de computadores, manifestaram interesse em desenvolver habilidades de programação, até perguntando aos instrutores onde poderiam aprender mais sobre isso. Ainda, a informação que os jogos utilizados foram feitos por estudantes e não por profissionais contribuiu difundir a idéia de que programação de computadores pode ser um alvo a ser alcançado até mesmo por integrantes de comunidades de excluídos digitalmente.

Referências

- FAGUNDES, L., 2004. *Como podemos vencer a exclusão digital*. Nova Escola, Brasília: v. 19, n. 172. Disponível em: <http://eebfmazzola.blogspot.com/2007/06/como-podemos-vencer-excluso-digital-lea.html> [15/08/2007].
- INSTITUTO BRASILEIRO DE GEOGRAFIA E ESTATÍSTICA, IBGE, 2005. *IBGE contou 32,1 milhões de usuários da internet no país*, Disponível em: http://www.ibge.gov.br/home/presidencia/noticias/noticia_visualiza.php?id_noticia=846&id_pagina=1, [04/08/07].
- JOHNSON, S., 2005. *Surpreendente!: a televisão e o videogame nos tornam mais inteligentes*. RJ: Elsevier.
- PEREIRA JR, J. C. R. E RAPKIEWICZ, C. E., 2004. O Processo de Ensino e Aprendizagem de Algoritmos e Programação: Uma Visão Crítica da Literatura. *WEIMIG*.
- PIAGET, J., 1983. *A epistemologia genética: sabedoria e ilusões da filosofia, problemas de psicologia genética*. São Paulo, Abril Cultural.
- RODRIGUES JUNIOR, M. C., 2002. Como Ensinar Programação? Informática – *Boletim Informativo* Ano I nº 01, ULBRA, Canoas, RS.
- TAROUCO, L., 2005. *Jogos educativos via WWW*, em: [http://penta3.ufrgs.br/animacoes/JogosEducaionais/\[18/07/07\]](http://penta3.ufrgs.br/animacoes/JogosEducaionais/[18/07/07]).