

Curso de introdução à programação de jogos para adolescentes na Vila Residencial

Eixo 4 – Tecnologias da Informação e Comunicação para Transformação Social

Artur A. M. Sarlo¹, Gabriel A. Souza², João V. F. Duarte³, Thársus A. Proux⁴

1 Universidade Federal do Rio de Janeiro - UFRJ, Cidade Universitária, Rio de Janeiro - RJ – artursarlo@poli.ufrj.br

2 Universidade Federal do Rio de Janeiro - UFRJ, Cidade Universitária, Rio de Janeiro - RJ – gabrielandrade@poli.ufrj.br

3 Universidade Federal do Rio de Janeiro - UFRJ, Cidade Universitária, Rio de Janeiro - RJ – joaov.ferreira@poli.ufrj.br

4 Universidade Federal do Rio de Janeiro - UFRJ, Cidade Universitária, Rio de Janeiro - RJ – tharsus_ap@poli.ufrj.br

Resumo

O Laboratório de Informática para Educação do Departamento de Engenharia Eletrônica da Escola Politécnica da UFRJ – LlpE desenvolve práticas educacionais na área de informática para crianças, adolescentes e adultos. Possui como objetivo incentivar o uso dos programas computacionais e reforçar os conteúdos do ensino fundamental, utilizando o computador como ferramenta no processo de ensino-aprendizagem, visando uma metodologia participativa entre alunos, monitores e professores.

No final do semestre passado (2012.2), através do diálogo com participantes de oficinas anteriores, foi levantada a possibilidade de oferecermos um curso de “criação de jogos eletrônicos”. A partir desta proposta, e incentivados pelo interesse dos alunos em um trabalho deste gênero, criamos uma oficina cujo público alvo eram adolescentes com faixa etária entre 12 e 15 anos, na qual poderiam aprender os básicos da lógica de programação, necessários para criar seus próprios jogos, simulações, desenhos ou programas para fazer contas no computador. Esse curso foi ministrado dentro da Associação de Amigos e Moradores da Vila Residencial da UFRJ - AMAVILA, no laboratório Jair Duarte, revitalizado em 2012 pelo LlpE.

No decorrer deste artigo, apresentaremos um relato da experiência vivenciada por nós nesses meses de atividade buscando enfatizar a metodologia utilizada, o conhecimento adquirido pelos participantes desse curso, sejam alunos ou instrutores, bem como as consequências do trabalho desenvolvido.

Palavras-chave: Extensão; Educação; Informática; Programação; Tecnologia Social;

Introdução

Como tornar a programação, um dos grandes temores das áreas técnicas e uma competência pouco explorada pelo ensino convencional, algo natural? O objetivo desse projeto foi demonstrar a lógica do funcionamento do computador através da programação de jogos. Ao idealizarmos esta oficina, uma das maiores

preocupações era fazer com que a lógica de programação não fosse um obstáculo intransponível ou desmotivador, afinal, quando os adolescentes procuravam o curso com a vontade de criar jogos, muitas vezes não faziam a menor ideia de como o processo de criação funcionava, ou seja, poucas chegavam com alguma noção do quebra-cabeça lógico que existe por trás de todo jogo eletrônico.

A ideia então, foi buscar softwares educacionais que simulassem uma linguagem de programação “própria”, de fácil adaptação, e que permitissem bastante interação gráfica (necessária a criação de jogos). Devido a seu bem sucedido histórico dentro do LpE, logo nos remetemos ao LOGO, programa no qual a lógica de programação poderia ser bem construída e explorada. Porém, ao analisarmos outras opções, decidimos que um de seus sucessores, o Scratch, atenderia perfeitamente aos objetivos.

O Scratch é um software de código aberto desenvolvido pelo MIT (Massachusetts Institute of Technology), cuja proposta é ensinar crianças e adolescentes com idade entre 8 e 13 anos a criar jogos e animações. Este software configura uma alternativa a programação clássica em editor de texto, já que apresenta uma abordagem gráfica, de fácil entendimento e apropriação. Assim os adolescentes podem criar jogos de forma simples e intuitiva, sem o requerimento de nenhum conhecimento prévio de programação ou lógica computacional.

Figura 1 – Tela principal do Scratch

Fonte: imagem retirada da tela inicial do software Scratch, durante o curso

Sua interface completamente em português, e a organização de cada função por blocos de encaixe com cores respectivas a natureza de seus comandos, faz com que o usuário compreenda rapidamente o grupo de cada comando e deduza os outros a partir dos mais básicos, fazendo com que a única e verdadeira dificuldade na elaboração do programa seja a estruturação lógica do mesmo, e não o conhecimento específico da linguagem.

Por ter uma resposta gráfica a tudo que é planejado na criação de um programa, o

scratch permite a rápida identificação do erro por parte do aluno, possibilitando um aprendizado construído em sala de aula por ele mesmo, em cima da análise e correção de seus erros.

Interações entre Alunos e Mediadores

As aulas, que aconteciam uma vez por semana, tinham como objetivo apresentar através da criação de jogos, o conceito da lógica de programação, logo, a cada encontro, tentávamos propor um jogo que, independente da maneira de construção, necessitasse de uma função lógica nova, fazendo com que os alunos, aos poucos, fossem se apropriando de todas as ferramentas do Strach podendo assim, tornar seus programas mais rebuscados e criativos.

Assim, ao começarmos uma aula nós buscávamos despertar o interesse nos alunos sugerindo um jogo novo. Perguntávamos suas opiniões, se alguma alteração poderia ser feita pra ele ficar mais interessante, e assim o desmembrávamos em diversos pequenos problemas mais intuitivos.

A resolução de cada um desses pequenos problemas começava com perguntas que nós fazíamos, que instigavam os alunos a pôr em prática as ferramentas que eles já dominavam. Tentávamos exercitar a ideia de “dividir para conquistar” onde um problema grande é dividido em diversos pequenos problemas fáceis de se resolver. Fomentávamos o raciocínio de questionamento do problema, ou seja, de se perguntar qual consequência uma nova função adicionada ao código do programa poderia acarretar. CHAVES e SOFNNER (2005, p.80) destacam a importância da reflexão no processo de aprendizagem, o que faz um paralelo interessante à abordagem do curso em incentivar o aluno a construir o conhecimento, e não apenas reproduzi-lo:

[...]Aprender, no contexto da informação disponível hoje no mundo, é algo bem diferente do que era ao final da Segunda Guerra Mundial. Em 1945 a informação disponível era escassa e o acesso a ela difícil. Por isso a educação formal das pessoas – realizada nas escolas – se limitava a um processo linear de transmissão de informações (organizadas em matrizes de disciplinas e séries), de professores (que possuíam essas informações) para alunos (que careciam delas), através do ensino (apresentação organizada e simplificada das informações). Aprender era visto como algo passivo, equivalente a absorver e assimilar as informações apresentadas – em geral sem maior entendimento e compreensão e mesmo que não tivéssemos a menor idéia de sua utilidade e aplicação no processo de tomada de decisão e ação no mundo em que nos cabe viver[...]

Nosso papel então pode ser descrito como o de mediadores da atividade (VALENTE, 2005), uma vez que instigávamos nos alunos questões novas a serem solucionadas, mas nunca dando maneiras prontas de resolver o problema (FREIRE, 1997), sempre dando a liberdade para que cada aluno resolvesse essas questões da maneira que se sentisse mais confortável. O fato de não intervirmos nos códigos dos alunos mesmo quando errados, fazia que os mesmos refletissem sobre seus

erros, e quando não os identificavam, buscávamos fazer perguntas que os fizessem compreender a não conformidade com o esperado.

Figura 2 – Aula de programação de jogos na Vila Residencial

Fonte: foto retirada no laboratório Jair Duarte durante o curso de programação

Metodologia

Vale ressaltar o caráter participativo da metodologia (THIOLLENT, 1998) empregada no curso, de forma que o planejamento das aulas era modificado de acordo com as sugestões dos alunos, ao apontarem alterações ou formas de melhorar os programas que eram propostos por nós.

Como ressaltado anteriormente sobre as interações aluno x mediador, acreditamos que o conhecimento deve ser construído com o aluno/aprendiz, sendo o artefato-computador um facilitador do processo de construção, e o supervisor um mediador desse processo. Ao falar sobre o LOGO, e o processo de construção do conhecimento, VALENTE (1998a, pp. 41-43) mostra a importância do mediador com um guia da atividade:

[...]A interação aluno-computador precisa ser mediada por um profissional que conhece Logo, tanto do ponto de vista computacional, quanto do pedagógico e do psicológico. Esse é o papel do mediador no ambiente. Logo. Além disso, o aluno como um ser social, está inserido em um ambiente social que é constituído, localmente, pelos seus colegas, e globalmente, pelos pais, amigos e mesmo a sua comunidade. O aluno pode usar todos esses elementos sociais como fonte de idéias, de conhecimento ou de problemas a serem resolvidos através do uso do computador[...]

É importante ressaltar que também visamos a pesquisa-ação (THIOLLENT, 1988), de forma que esses alunos inspirados por suas novas capacidades, ensinem ou motivem outros adolescentes a buscar os mesmos conhecimentos, se tornando agentes de transformação de seus meios sociais.

Figura 3 – Construção do jogo Pac Man

Fonte: foto retirada no laboratório Jair Duarte durante o curso de programação

Considerações Finais

A lógica computacional não é muito visada em colégios de ensino fundamental e médio, mesmo sendo uma competência extremamente agregadora. Ela nos permite separar as tarefas, organizá-las em prioridades e realizá-las da melhor maneira possível. Fora isso, o conhecimento de programação é um diferencial importante pra quem busca empregos em áreas técnicas ou administrativas, configurando um conhecimento útil até mesmo para aqueles que venham exercer funções dissociadas deste contexto.

O objetivo do curso que havíamos idealizado no início foi alcançado com êxito. Uma disciplina que envolve um alto grau de dificuldade pôde ser encarada com uma prática divertida e simples. Apesar de uma divulgação pouco intensa, que gerou uma procura pequena, aqueles que ficaram até o final se mostraram muito interessados e conseguiram solucionar todos os problemas sugeridos, absorvendo a ideia da lógica computacional.

Enquanto estudantes de engenharia eletrônica e por sua vez, também de programação, tivemos a oportunidade de não só enraizar os conceitos que já detínhamos, mas também de refletir sobre sua importância na vida fora do ambiente técnico de engenharia. Estar em contato com um grupo, por ele ser responsável e se preocupar em estimulá-lo é uma realidade intrínseca do trabalho de um engenheiro. Lidar com seres humanos, com suas individualidades e limitações e assim saber como agir com cada um é uma competência exigida, também, de um profissional de qualidade. Mas infelizmente o exercício de lidar com essas situações, por muitas vezes se mostra deficiente no ambiente acadêmico. Participando como agentes responsáveis durante o período da oficina, pudemos exercitar esse lado de nossa formação, tornando a atividade ainda mais interessante para nós, pois além do retorno pessoal de ensinarmos algo que gostamos, tivemos um retorno profissional muito valioso.

Referências Bibliográficas

CHAVES, E. O. C. e SOFNNER, R. K. Educação Temática Digital. **Tecnologia e a Educação como Desenvolvimento Humano**. Campinas, v.6, n.2, p.77-84, jun. 2005.

FREIRE, P. **Pedagogia da Autonomia: saberes necessários à prática educativa**. 4ª Edição. Rio de Janeiro: Paz e Terra, 1997.

THIOLLENT, M. J. M. **Extensão Universitária e Metodologia Participativa**. 1ª edição. Rio de Janeiro: COPPE/UFRJ, 1998.

VALENTE, J.A. **A Espiral da Espiral de Aprendizagem: o processo de compreensão do papel das tecnologias de informação e comunicação na educação**. 2005. 232f. Tese para a obtenção do título de Professor Livre Docente – Instituto de Artes, Universidade Estadual de Campinas, Campinas. 2005.

VALENTE, J.A. Análise dos diferentes tipos de software usados na educação. Em: J. A. Valente (Org) **Computadores na sociedade do conhecimento**. São Paulo: UNICAMP/NIED, 1999. p. 89-110.