

Uma abordagem lúdica para a aprendizagem de programação de computadores

Fabio Vinicius Binder¹, Vidal Martins¹

¹Departamento de Ciência da Computação
Pontifícia Universidade Católica do Paraná (PUCPR) – Curitiba – PR - Brasil

{fabio.binder, vidal.martins}@pucpr.br

Abstract. *This paper describes the experience in teaching computer programming with the aid of games at PUCPR through the last four years. The activity known as “Tutoria de Jogos” is helping students of Computer Science to learn programming. It motivates them to participate in events and it is an important factor to decrease the evasion.*

Resumo. *Este artigo descreve a aprendizagem de programação com o uso de jogos na PUCPR nos últimos quatro anos. A atividade de Tutoria de Jogos vem melhorando a aprendizagem de programação no curso de Ciência da Computação, motivando alunos a participarem de eventos e colaborando com o aumento da procura pelo curso e com a diminuição das evasões.*

1. Introdução

A programação de computadores é um dos principais fundamentos da ciência da computação, sendo base para diversas disciplinas como computação gráfica, inteligência artificial, compiladores, estruturas de dados, complexidade de algoritmos, sistemas operacionais, processamento de imagens, entre outras. Além disso, o cientista da computação precisará desenvolver programas, ou pelo menos propor soluções algorítmicas para os problemas, em suas atividades profissionais e de pesquisa. Assim, pressupõe-se que um bom cientista da computação seja também um bom programador.

Aprender a programar não é uma tarefa simples: exige abstração, raciocínio lógico, rigor no uso das linguagens de programação, habilidade para manipular as ferramentas de desenvolvimento de programas, concentração, memória, capacidade de lidar com várias linhas de raciocínio simultâneas e imaginação para encontrar erros. As dificuldades na aprendizagem de programação podem desmotivar os estudantes e, em último caso, fazê-los desistir do curso de Ciência da Computação.

Levando-se em conta essas dificuldades, o problema abordado neste artigo é descobrir como podemos tornar a aprendizagem de programação de computadores uma atividade prazerosa e motivadora, contribuindo, assim, para minimizar os índices de evasão dos cursos de Ciência da Computação.

Uma das ações propostas pela PUCPR para solucionar o problema foi a criação do projeto de Tutoria de Jogos. O objetivo principal desse projeto é motivar os alunos a aprenderem a programar. Foi implantado como atividade optativa no primeiro ano do

curso e consiste em desenvolver programas atrativos e relacionados aos interesses da maioria dos estudantes: jogos digitais.

No início do processo é ensinado o ciclo de vida do software e aspectos do desenvolvimento de jogos. Após o aprendizado dos aspectos mais básicos de programação os alunos começam a desenvolver partes mais simples de um jogo pré-programado, porém incompleto. A complexidade dos trabalhos vai aumentando, exigindo-se a aplicação de conceitos mais difíceis. Os alunos são estimulados a desenvolverem trabalhos de qualidade, tanto para receberem notas válidas para a disciplina de iniciação à computação quanto para participarem de eventos de jogos, internos e externos.

A tutoria de jogos está em seu quinto ano de existência e já apresenta como resultados práticos: a participação em eventos; a percepção por parte dos alunos de que a atividade é relevante para a sua formação; e o aumento pela procura do curso e diminuição da taxa de evasão.

Até o momento foram realizadas três mostras de jogos e dois trabalhos foram apresentados no SBGames. Para avaliar a motivação dos alunos, foi aplicado um questionário a três turmas que já fizeram a Tutoria de Jogos em anos anteriores. O resultado da avaliação mostrou que a grande maioria dos alunos (92%) considera que a atividade foi relevante na sua formação. Outro resultado positivo que pode ser considerado, em parte, fruto da tutoria de jogos é que nos últimos anos foi observado um aumento contínuo no número de vestibulandos e também uma diminuição da taxa de evasão do curso.

O presente artigo está organizado da seguinte forma. A Seção 2 apresenta o projeto de tutoria de jogos com seus objetivos e fases. Na Seção 3 é mostrado o aumento gradual do nível de dificuldade dos programas cobrados dos alunos durante o ano letivo. A Seção 4 descreve a Mostra de Jogos. Na Seção 5 são discutidos os resultados obtidos pela Tutoria de Jogos. A Seção 6 analisa os trabalhos relacionados ao uso de jogos no ensino de programação. Finalmente, a Seção 7 conclui o artigo e propõe trabalhos futuros.

2. O projeto de tutoria de jogos

Uma das atividades que dão mais prazer à maioria dos alunos é o ato de jogar jogos de computador. Este atrativo muitas vezes estende-se ao desenvolvimento de um jogo, já que a satisfação ao ver sua criação pronta e funcionando pode ser maior do que a de jogar até o final o seu jogo preferido.

O projeto de tutoria de jogos foi implantado como atividade optativa no primeiro ano do curso aproveitando-se da capacidade que os jogos têm de divertir e encantar. Nela são desenvolvidos programas atrativos e relacionados aos interesses da maioria dos estudantes: jogos digitais. Os alunos que não gostam da área de jogos ou que não se sentem motivados por ela, podem propor outro aplicativo de complexidade semelhante para que o mesmo grau de motivação seja atingido. A título de exemplo, já foi desenvolvida nesta tutoria um editor gráfico simples, semelhante ao Paint da Microsoft.

Os encontros da tutoria de jogos são semanais, previstos na grade horária do curso. Esclarecimentos de dúvidas são realizados no horário da permanência do professor responsável. O índice médio de adesão é superior a 90%, tanto pela motivação da atividade quanto pelo fato de valer nota para outra disciplina e também contar como

atividade complementar para aqueles que a terminarem. Todo aluno é obrigado a cumprir um número mínimo de horas complementares para se formarem, de acordo com regimento interno da instituição.

É utilizada a linguagem C como padrão para os projetos dos alunos e considera-se que a maior parte dos alunos não sabe programar, fato corroborado pelo questionário descrito na seção 5.2. Ela é dividida em quatro fases que levam em consideração o conjunto de habilidades de programação adquiridas pelos alunos até o momento: projeto de jogos, desenvolvimento de pequenos jogos simples e de parte da lógica de jogos semi-prontos, prototipação de um jogo de autoria própria e desenvolvimento de um jogo completo em 2D.

No início do processo, quando os alunos ainda não sabem programar, é ensinado o ciclo de vida do software (especificação de requisitos, análise, codificação, testes, validação e documentação) e os alunos são desafiados a apresentarem o projeto de um jogo simples 2D. Também são ensinados aspectos específicos do desenvolvimento profissional de jogos como game design, imagens, animação e sons.

O processo de desenvolvimento de software escolhido é simples e direto, compatível com o nível de maturidade dos alunos que são iniciantes na área. Inicialmente eles definem o conceito do jogo que pretendem seguir. Também fazem uma pequena análise de mercado, comparando o seu jogo com possíveis concorrentes. Em seguida apresentam um documento de game design detalhando o jogo. Nele são indicados os requisitos funcionais e não funcionais, a navegação das telas, descrição de personagens, identificação de animações, cenários, efeitos sonoros e músicas.

Após a etapa de projeto, os alunos praticam programação desenvolvendo jogos texto bem pequenos como jogos de adivinhação e *adventures* e também terminam jogos 2D que são apresentados de forma incompleta. Dessa maneira eles não só treinam programação, mas também aprendem como se organiza um programa e têm a satisfação de colaborar no desenvolvimento de um jogo 2D e vê-lo funcionando.

A etapa de prototipação do jogo leva em conta principalmente os aspectos acessórios do projeto, como navegação de telas, apresentação de menus, leitura e gravação de arquivos de recordes e *savegames*. O conteúdo do jogo deve estar representado no protótipo de acordo com o seu tipo. Por exemplo, na categoria plataforma, o jogo deve conter o personagem principal se movimentando em um labirinto sem inimigos e sem itens adicionais como saúde, bônus, etc.

A etapa final consiste na apresentação e entrega do jogo, de um site completo, de um manual de instruções de instalação e de como jogar e um documento de *post mortem* (lições aprendidas). O site deve conter a história do jogo; links para download do jogo, documento de game design, manual de instruções e *post mortem*; e também algumas screenshots do jogo.

Os alunos são estimulados de duas formas a desenvolverem trabalhos de qualidade: cada atividade desenvolvida vale nota para a disciplina de iniciação à computação; e os melhores trabalhos são escolhidos para participarem da Mostra de Jogos, evento no qual os jogos dos alunos são apresentados à comunidade, podendo receber o prêmio de melhor jogo. Os alunos fazem uma apresentação do jogo e da página web correspondente perante uma banca de professores, que avalia cada jogo de acordo com critérios de

qualidade previamente estabelecidos. Apenas os jogos que obtiverem nota igual ou superior a 7,0, em uma escala de 0 a 10, são convidados a participar da mostra de jogos e são estimulados a se aprimorarem para participar do festival de jogos independentes do SBGames. Os critérios de qualidade utilizados são: qualidade gráfica e análise segundo o framework de avaliação de jogos MDA [Hunicke 2004].

Paralelamente ao projeto da tutoria, os alunos desenvolvem na disciplina de iniciação à computação o site de apresentação do jogo. Nele são utilizadas diversas tecnologias de apoio como Flash, JavaScript, HTML e CSS. Ao final do ano letivo os alunos entregam, junto com o jogo, um site contendo informações gerais sobre a equipe e sobre o jogo, formulário de contato, *screenshots*, manual e download do jogo.

Os jogos e links para os sites dos alunos selecionados para a Mostra podem ser encontrados na página de jogos do curso [Mostra de Jogos 2010].

3. O aumento gradativo de dificuldade dos programas

Após o aprendizado dos aspectos mais básicos de programação (variáveis, operadores, condições e repetições), os alunos já estão em condições de desenvolver jogos bem simples ou partes de um jogo de relativa complexidade.

Existem muitos exemplos de jogos em modo texto que podem ser desenvolvidos por alunos iniciantes em programação. Jogos de perguntas e respostas, jogos simples de tabuleiro que não exigem animação, aventuras e rpgs textuais, jogos de adivinhação, entre outros. Muitos deles podem ser criados rapidamente durante o horário de aula e por isso são ótimos candidatos para as atividades iniciais de programação na tutoria. Como esses jogos são pequenos aplicativos práticos e auto-contidos, oferecem uma boa visão inicial da estruturação de aplicativos. Também auxiliam no entendimento de como os elementos básicos de programação (condições e repetições) são utilizados para formarem programas maiores.

A motivação pode aumentar quando um aluno contribui em um projeto maior mesmo que não o tenha feito sozinho. Para tornar isso possível, são fornecidos jogos 2D incompletos, nos quais as partes mais complexas estão prontas, cabendo ao aluno completar o jogo, normalmente escrevendo funções que tratam da lógica do jogo. Essa abordagem mantém os alunos motivados, pois eles conseguem ver resultados satisfatórios desde o início de seu aprendizado de programação. Outra consequência positiva deste processo é o conhecimento da organização de um programa mais complexo que o aluno adquire, já que ele necessariamente tem que estudar o programa para poder completá-lo.

Um dos exemplos de jogos incompletos propostos para os alunos é um jogo de naves, semelhante a *space invaders*, no qual tanto a nave quanto os inimigos não se movimentam. As funções de manipulação de teclado e mouse, de movimentação dos inimigos e de atualização do placar existem dentro do código, mas estão vazias. Cabe aos alunos elaborarem as soluções e as implementarem nos locais adequados. Cada conceito importante ensinado na tutoria é acompanhado de exemplos como o descrito acima, fazendo com que os alunos tenham que entender algumas partes do código para que possam resolver o problema.

4. Mostra de Jogos

A Mostra tem como objetivo valorizar o trabalho dos alunos por meio de divulgação e eleição dos melhores trabalhos e discutir a realidade do mercado de jogos digitais por meio de palestras e apresentação de experiências.

O evento é realizado em dois espaços físicos: saguão do Centro de Ciências Exatas e Tecnológicas (CCET), onde os alunos expõem seus jogos para quem quiser conhecê-los; as palestras são ministradas no auditório do mesmo bloco. Dura um dia inteiro, manhã tarde e noite, sendo amplamente divulgado no âmbito da universidade e também para a comunidade local. Normalmente, há cobertura da imprensa local.

Os melhores trabalhos, votados por uma comissão de professores, são estimulados a participar do SBGames e outros concursos de jogos e são convidados para serem expostos na Feira de Cursos da PUCPR, evento destinado a apresentar à comunidade os cursos de graduação oferecidos pela instituição com o intuito de auxiliar os vestibulandos na decisão sobre o curso a ser seguido.

Até o presente artigo, 3 mostras de jogos já haviam sido realizadas nos anos de 2006, 2007 e 2009. Nos dois primeiros anos as mostras de jogos foram conduzidas ao final do ano. No ano de 2008, a data da Mostra foi alterada para o segundo trimestre do ano seguinte com o objetivo de dar mais tempo para que os alunos interessados pudessem melhorar o jogo e para motivar os calouros a desenvolverem seus jogos e a participarem da mostra no ano seguinte. Os jogos apresentados nas Mostras anteriores estão disponíveis em [Mostra de Jogos 2010].

5. Resultados obtidos

Dentre os resultados positivos do projeto de Tutoria de Jogos, destacamos os seguintes: a apresentação de dois trabalhos no festival de jogos independentes do SBGames 2008, na PUC Minas – Snakexpress e Goldberry Land [SBGames 2008]; a motivação dos alunos para a aprendizagem de programação; a percepção do projeto como uma atividade relevante por parte dos alunos, conforme discutido na Subseção 5.1; o aumento da procura pelo curso e a diminuição da evasão, conforme demonstrado na Subseção 5.2.

5.1. Motivação e Evasão do Projeto

Com o objetivo de verificar se a Tutoria de Jogos está alcançando seus objetivos, realizamos uma pesquisa junto aos alunos do segundo, terceiro e quarto anos do curso, que participaram do projeto nos últimos três anos, e obtivemos resultados interessantes. De um total de 91 alunos que passaram pela tutoria nesse período, 50 responderam nossa pesquisa.

O questionário utilizado na pesquisa está estruturado em duas partes: na primeira, o aluno responde uma pergunta aberta para identificar pontos positivos e negativos da atividade, enquanto, na segunda, as perguntas são mais específicas e versam sobre aspectos relevantes da atividade. A Tabela 1 apresenta a parte dois do questionário. Essas partes foram respondidas em ordem e sem que os alunos lessem a segunda parte antes da primeira. Também não foi permitido alterar a primeira parte após sua finalização. O motivo dessa divisão foi confrontar *a posteriori* as respostas abertas com as respostas induzidas. Em todos os casos as respostas das duas partes foram coerentes, indicando

que a percepção dos alunos foi a mesma tanto na pesquisa induzida, quanto na parte aberta.

Tabela 1. Questionário de avaliação da Tutoria de Jogos – 2ª parte

Questão	Respostas possíveis
1) Você já sabia programar antes de fazer a Tutoria de Jogos?	(a) sim, já conhecia o básico (b) sim, já tinha um conhecimento intermediário ou avançado sobre o tema (c) não
2) O objetivo principal da Tutoria de Jogos é ensinar a programar de forma lúdica e agradável. O objetivo foi cumprido, ou seja, você ficou motivado a programar por causa das atividades da Tutoria de Jogos?	(a) sim, totalmente (b) sim, parcialmente (c) não
3) Explique sua resposta ao item anterior se você marcou a letra b ou c.	aberta
4) Você conseguiu desenvolver o jogo completo? Se não conseguiu, explique qual foi o motivo principal que o impediu de completar o trabalho.	Sim ou não. Motivo de não conseguir completar o trabalho
5) Algum tópico visto na Tutoria de Jogos o ajudou a entender melhor tópicos vistos em algoritmos / programação?	(a) não (b) sim, poucas vezes (c) sim, algumas vezes (d) sim, muitas vezes
6) A atividade de Tutoria de Jogos foi relevante para a sua formação como programador? Por que?	Sim ou não. Justificativa da resposta

O resultado geral das questões que estavam diretamente associadas com os objetivos da Tutoria de Jogos (motivar os alunos, facilitar o aprendizado de programação e ser relevante na sua formação) foram muito positivos. Os resultados das questões abertas ou semi-abertas foram categorizados para identificar os diferentes grupos de alunos e quantificar com mais facilidade as respostas recebidas, conforme será descrito a seguir.

A primeira questão, relacionada com um pré-conhecimento de programação, indicou que a maioria dos estudantes (60%) não sabia programar antes de frequentar as aulas da Tutoria de Jogos. Essa informação é de extrema importância já que permite validar os objetivos da tutoria com maior precisão.

A questão dois pergunta claramente se o estudante se sentiu motivado a programar por causa das atividades da Tutoria de Jogos. Apenas 10% das respostas foram negativas, enquanto 90% dos alunos afirmaram ter encontrado na Tutoria motivação total ou parcial para programar. Destaca-se o fato de que todos os alunos que consideraram o objetivo da atividade parcialmente atendido responderam positivamente à questão seis, indicando que a atividade tem relevância para sua formação.

A questão quatro nos permitiu identificar que 68% dos estudantes completaram o trabalho proposto. Entre os que não conseguiram alcançar seus objetivos, o motivo mais citado para o insucesso foi o fator tempo, com 31% dos casos

A questão cinco perguntava se a atividade facilitou o aprendizado de algum tópico de programação. Para 20% dos estudantes, a Tutoria não foi relevante nesse sentido (10% responderam *não* e mais 10%, *poucas vezes*). Porém, o desenvolvimento de jogos facilitou a aprendizagem de programação para 80% dos estudantes (48% responderam *algumas vezes* e 32%, *muitas vezes*). Novamente, o resultado foi positivo, demonstrando que, na maioria dos casos, os alunos aprenderam mais facilmente tópicos de programação graças a tópicos vistos na Tutoria de Jogos.

A Figura 1 resume as respostas fornecidas para a questão seis, que trata da relevância da Tutoria de Jogos na formação dos alunos. Para 92% dos pesquisados, a atividade foi considerada relevante, pelos seguintes motivos: 48% indicaram que a Tutoria

de Jogos facilitou o aprendizado de algum conceito de programação (funções, ponteiros, organização dos programas, arquivos foram os mais citados); 18% indicaram que adquiriram fluência na linguagem; 12% gostaram de trabalhar com gráficos e igual percentual teve satisfação em ver o resultado final funcionando. Dentre os alunos que não consideraram a atividade relevante, 75% indicaram que não gostam de jogos e, portanto, não tiveram motivação para participar da atividade ou terminar o jogo.

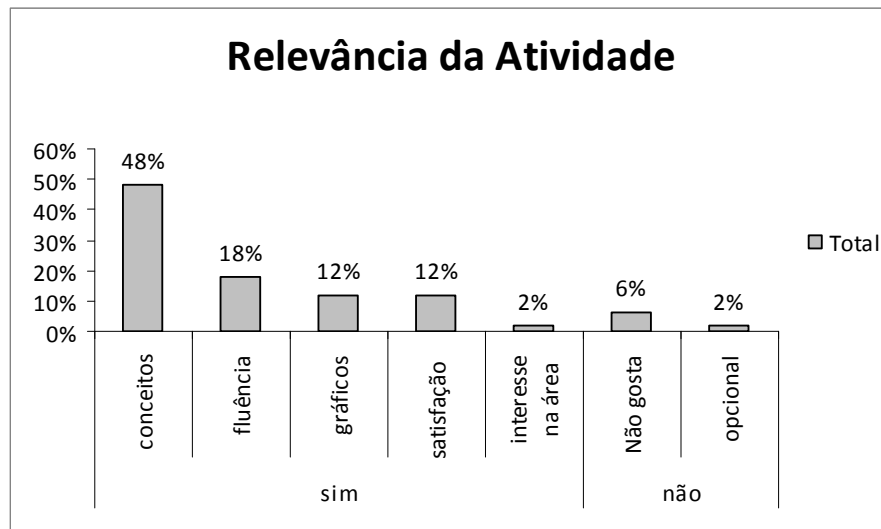


Figura 1. Gráfico das respostas para a questão seis do questionário

5.2. Redução da Evasão no Curso

A Tutoria de Jogos iniciou no ano de 2006 e se consolidou em 2007. Os trabalhos produzidos no contexto desse projeto passaram a ser utilizados na divulgação do curso de Ciência da Computação da PUCPR, em eventos como a Feira de Cursos da PUCPR, nas Feiras de Profissões promovidas por escolas do Ensino Médio, em palestras sobre o curso em escolas do Ensino Médio, entre outros. A partir de 2008 verificou-se um aumento contínuo tanto no número de vestibulandos quanto no percentual de vagas preenchidas do curso, conforme mostra a Tabela 2.

Acredita-se que os jogos criaram certa identidade para o curso, ou seja, constituem um meio compreensível e motivador de explicar aos vestibulandos o que um cientista da computação é capaz de fazer após terminar o curso. Na própria Feira de Cursos da PUCPR é oferecida uma Oficina de Jogos na qual o aluno é guiado no processo de construir um pequeno *sprite* e inseri-lo dentro de um jogo pré-programado. A clareza alcançada com essas iniciativas ajuda a direcionar para o curso um número maior de estudantes com vocação para o assunto, fato que reduz as frustrações e, consequentemente, a evasão.

Embora a Tutoria de Jogos não seja a única iniciativa que visa motivar e reter o aluno no curso, ela certamente contribuiu para o alcance desse objetivo, principalmente, quando se considera o resultado da pesquisa apresentado na Seção 5.1.

Tabela 2. Vagas preenchidas no curso de Ciência da Computação da PUCPR

Ano	Vagas Preenchidas Início do Ano	Vagas Preenchidas Final do Ano	Vagas Preenchidas / Vagas Oferecidas
2008	115	104	43,3%
2009	124	111	46,3%
2010	142	128*	53,3%*

* previsão baseada na evasão dos dois anos anteriores

6. Trabalhos relacionados

A utilização de jogos e *puzzles* no ensino de programação pode ser encontrada de modo genérico em diversos livros de algoritmos na forma de exemplos ou exercícios propostos como torre de Hanói, jogos de adivinhação, jogo da velha, entre outros.

Em [Feijó *et all* 2010] é proposta uma forma de apresentar o assunto de programação utilizando jogos e a linguagem Java. O assunto de jogos é utilizado em alguns exemplos, nos exercícios propostos e nos trabalhos sugeridos. As principais diferenças com relação ao trabalho da Tutoria de Jogos são: a simplificação do assunto de programação, a utilização de gráficos e animações apenas ao final da disciplina. Outro diferencial da tutoria é a motivação adicionada pela Mostra de Jogos e pelo SBGames.

A questão da linguagem utilizada para ensinar programação com jogos é tratada de maneira objetiva em [Leutenegger 2004]. Os alunos iniciam o aprendizado de programação com Flash e ActionScript, passando em seguida para C++, culminando com o ensino de OpenGL e jogos 3D. A construção de jogos é utilizada na própria disciplina que ensina programação.

Uma abordagem baseada em jogos mais simples é utilizada em [Rajaravivarma 2005] para minimizar os erros mais comuns cometidos por alunos iniciantes nas quatro etapas de desenvolvimento: identificação do problema, elaboração da solução, implementação e testes. Os jogos propostos foram classificados em duas categorias: palavras e números, sendo todos feitos apenas em modo texto, tendo como objetivo principal o desenvolvimento das habilidades de pensamento crítico e solução de problemas.

Uma experiência feita no sentido de identificar o real interesse pelo desenvolvimento de jogos em um curso de Ciência da Computação é descrita em [Bayliss 2006]. Com o objetivo de responder a uma reclamação recorrente dos alunos de que o ensino de programação estaria dissociado da realidade de aplicações práticas, foi procurado e identificado um assunto de interesse e conhecimento da maioria dos alunos: o desenvolvimento de jogos digitais. Foi oferecido um curso opcional à distância no qual os alunos interessados foram selecionados de acordo com seus desempenhos na seleção e no seu conhecimento de programação, sendo dada prioridade para os programadores iniciantes. O estudo demonstrou que o interesse existe e que a maior parte dos alunos teve sucesso na finalização dos projetos.

As principais contribuições do nosso trabalho em relação a essas experiências que abordam o mesmo problema são: a programação de jogos sendo oferecida como uma atividade distinta; a possibilidade de programar jogos 2D na linguagem principal

do curso e nos primeiros meses da disciplina; a resolução de problemas complexos no final da Tutoria; a descoberta natural de uma visão simplificada do ciclo de vida do software; o estímulo ao estudante à superação dos seus limites por meio da Mostra de Jogos e do apoio à participação no SBGames; a validação do processo considerando a percepção dos alunos.

No curso de Ciência da Computação da PUCPR a programação é ensinada em duas disciplinas: programação imperativa; e algoritmos e estruturas de dados. A Tutoria é uma atividade à parte, ou seja, o aluno aprende os conceitos fundamentais de algoritmos e programação e depois os utiliza para o desenvolvimento de jogos digitais. Gráficos 2D e animações em geral são programados em C desde o início na Tutoria de Jogos por meio da utilização de uma biblioteca própria. Conforme explicado na seção 3 do presente artigo, isso é possível devido ao aumento gradual da dificuldade dos programas propostos.

7. Conclusão

O problema analisado neste artigo foi a dificuldade na aprendizagem de programação de computadores que pode levar muitos alunos a desistirem do curso de Ciência da Computação.

A solução proposta foi a implantação de uma atividade adicional: a Tutoria de Jogos na qual os alunos são convidados a desenvolverem um jogo 2D enquanto aprendem a programar. Nela, os alunos aprendem noções sobre o ciclo de vida de software, detalhes sobre o processo de desenvolvimento de um jogo e programam um jogo 2D completo, utilizando diversos assuntos abordados nas disciplinas de algoritmos e programação. A solução tem o mérito de aumentar a motivação dos alunos e manter alto o nível de exigência nas disciplinas de programação e de algoritmos.

Os principais resultados alcançados com a Tutoria de Jogos foram o aumento da procura pelo curso, diminuição da evasão, a participação dos alunos na Mostra de Jogos da PUCPR e a apresentação de trabalhos no festival de jogos independentes do SBGames. De acordo com uma pesquisa realizada entre os estudantes que participaram do projeto nos últimos 3 anos, 92% dos alunos consideraram que a Tutoria de Jogos foi relevante para a sua formação como programadores e 90% concordam que o objetivo de aprender a programar de forma lúdica foi atingido.

A tutoria de jogos continua e está sendo melhorada a cada ano para atender a novas expectativas e englobar outras iniciativas da PUCPR relacionadas a jogos. Para aumentar a qualidade visual dos jogos, será feita uma integração com o curso de graduação em Design Digital. Os alunos de Design participarão dos jogos tanto na direção de arte quanto na execução das telas, ilustrações e animações. Outro desejo dos alunos que será atendido é a possibilidade do desenvolvimento de jogos para plataformas específicas, como consoles e smartphones, mais especificamente o iPhone. A partir de 2009 outros cursos relacionados a jogos foram lançados na instituição. A Mostra de Jogos será ampliada para abranger os novos cursos de tecnologia em jogos digitais e especialização em jogos digitais. Ainda com relação à Mostra, serão trazidos palestrantes que possuam trabalhos relevantes na área de jogos, a fim de tornar o evento mais atrativo e aumentar a motivação e participação dos estudantes.

7. Referências

- Feijó, B., Clua, E. e da Silva, F. S. C. (2010) Introdução à Ciência da Computação com Jogos, Editora Campus, 1ª. edição.
- Rajaravivarma, R. (2005) A Games-Based Approach for Teaching the Introductory Programming Course *in* The SIGSCE Bulletin, Vol 37, Number 4, pages 98-102.
- Leutenegger, S. e Edgington, J. (2004) A Games First Approach to Teaching Introductory Programming *in* Proceedings of the 38th SIGCSE technical symposium on Computer science education, pages 115-118
- Bayliss, J. D. e Strout S. (2006) Games as a “Flavor” of CS1 *in* Proceedings of the 37th SIGCSE technical symposium on Computer science education, pages 500-504
- Mostra de Jogos (2010) <http://www.pucpr.br/graduacao/cienciacomputacao/jogos.php>
- Hunicke, R et all. (2004) “MDA: A Formal Approach to Game Design and Game Research”. Available at <http://www.cs.northwestern.edu/~hunicke/pubs/MDA.pdf>
- SBGames (2008) <http://www.inf.pucminas.br/sbgames08/>