

# Relato de um modelo de tutoria para programação baseado em experiências com ingressantes de um curso de Licenciatura em Computação

Pasqueline Scaico, Any Carolyn Duarte, Gabriella Mayara T. Alves, Mariana Maia, Fernando Mateus de Oliveira, Sinval Vieira M. Neto, Marcus Rafael X. Laurentino, Alexandre Scaico, Vanessa Farias Dantas<sup>1</sup>

Centro de Ciências Aplicadas e Educação, Universidade Federal da Paraíba, Rio Tinto, PB

{pasqueline, any.caroliny, gabi.mayara, mariana.maia, fernando.mateus, sinval.mendes, marcus.rafael, alexandre, vanessa}@dce.ufpb.br

**Abstract.** Despite programming being the basis for the development of many required skills in Computer Science courses, learn how to program can be a great challenge for beginners. Considering the various initiatives already proposed to solve this problem, this article reports the results of a mentoring experience that has adopted several strategies to motivate and support these students, producing not only good indicators of success, but also a model that can be used in other courses.

**Resumo.** Apesar de programação ser a base para o desenvolvimento de muitas habilidades exigidas nos cursos de Ciência da Computação, aprender a programar pode ser um grande desafio para os iniciantes. Diante das várias iniciativas já propostas para solucionar esse problema, este artigo relata os resultados de uma experiência de tutoria que adotou diversas estratégias para motivar e apoiar esses alunos, tendo como resultado não apenas bons indicadores de sucesso, mas também um processo que pode servir de modelo para outros cursos.

## 1. Introdução

As disciplinas de programação são a base para o desenvolvimento lógico e algorítmico de um universitário na área de Computação e são essenciais para construir a forte fundamentação necessária para outras disciplinas relacionadas. Na perspectiva dos novatos, aprender a programar é relatado como um processo desanimador, solitário e desestimulante que finda, por vezes, no aumento dos altos índices de evasão e retenção dos cursos superiores em todo o mundo (Kinnunen 2006) e (Jenkins 2001).

Para facilitar a aprendizagem dos iniciantes, diversas pesquisas têm proposto mecanismos para mediar o processo de ensino. Kelleher (2005) menciona a necessidade de se utilizar ambientes de programação que sejam mais simples e intuitivos. Miller (2004) sugere que o primeiro contato com a programação seja por meio de linguagens de sintaxe mais simples. Outro problema relatado é a dificuldade de analisar problemas mal estruturados. Assim, Mendonça (2010) retoma esta questão e propõe um modelo de programação orientado à exploração de problemas. Além dessas abordagens, percebe-se

que os métodos de ensino têm incorporado mais mecanismos motivacionais, com a robótica pedagógica e os objetos de aprendizagem (Rosa et al 2006) e (Rocha 2011).

Todavia, muitos desafios estão estabelecidos em função de questões relacionadas à habilidade de lidar com a complexidade, de abstrair conceitos e de reconhecer padrões (Sudol 2011), (Teague 2009) e (Ala-Mutka2004). Outros problemas de aprendizagem podem ser resultantes da inabilidade do aluno estabelecer uma nova rotina de estudos, que lhe permita, inclusive, fazer uso de uma melhor gerência de tempo. Assim, acompanhar o percurso dos estudantes e assisti-los para que novas práticas sejam estabelecidas também pode agregar ao processo de ensino-aprendizagem.

Este trabalho apresenta uma experiência de tutoria com iniciantes de um curso de Licenciatura em Computação que resultou em um processo de assistência, que pode servir de modelo para outros cursos. Foi realizado um experimento que observou a influência desse acompanhamento para o desempenho dos tutelados na disciplina. Como os tutores foram os próprios alunos do curso (veteranos), pode-se dizer que houve uma contribuição na melhoria das suas habilidades de comunicação e docência. Registra-se neste artigo a aplicação de alternativas pedagógicas em relação à abordagem tradicional de sala de aula, a exemplo do uso de atividades que estimulam o pensamento computacional, o uso de robótica, objetos de aprendizagem, ferramentas visuais de programação e o uso de técnicas para a análise de problemas mal estruturados.

O artigo está organizado da seguinte maneira. A Seção 2 apresenta um breve histórico do desempenho dos discentes da disciplina Introdução à Programação em uma universidade pública. Depois são apresentados trabalhos relacionados ao tema deste artigo na Seção 3. A Seção 4 apresenta o processo utilizado para a tutoria. A Seção 5 descreve a aplicação de um experimento para validação e refinamentos do processo, assim como, apresenta os resultados observados. Na Seção 6, as considerações finais são apresentadas, assim como, as contribuições deste trabalho.

## 2. Contextualização: Desempenho discente na disciplina Introdução à Programação

Um mapeamento relacionado à disciplina Introdução à Programação foi realizado no curso de Licenciatura em Computação, da Universidade Federal da Paraíba, para que se pudesse dimensionar o desempenho dos estudantes. Os dados referentes de 2007 a 2010 apontaram uma realidade preocupante. Considerando a média de sessenta alunos por turma, observou-se que, no segundo semestre de 2008, por exemplo, 90% dos alunos não obteve aprovação na disciplina (Figura 1a).

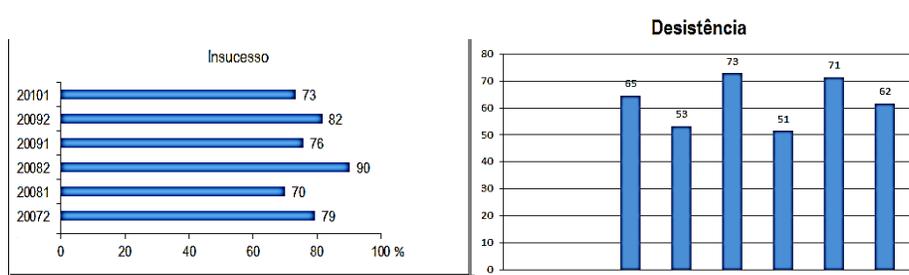


Figura 1: (a)Taxa de insucesso (b) Taxa de desistência

A análise também refletiu o índice de evasão existente no curso. Como se pode verificar pela Figura 1b, o percentual de alunos que abandonou a disciplina atingiu em mais de um semestre o patamar de 70%. Na tentativa de identificar se aqueles alunos que haviam abandonado a disciplina ainda continuavam no curso, foi realizado um cruzamento de dados que mostrou que cerca de 80% dos alunos que desistiram do curso estavam no terceiro semestre letivo. Percebeu-se, então, que os três primeiros semestres eram críticos para a permanência desses estudantes.

Diante dessa situação, foi realizada uma primeira mudança para reestruturar a disciplina Introdução à Programação. A linguagem de programação até então utilizada era Pascal. Com o intuito de utilizar uma linguagem mais simples e motivacional, Python foi escolhida, inclusive por dispor de muitas bibliotecas (para o desenvolvimento de jogos, por exemplo) e por ser uma linguagem que os alunos poderiam continuar explorando mesmo com o fim da disciplina, o que não era o caso de Pascal. Além disso, pela natureza orientada a objetos, Python facilitaria a transição para o paradigma das disciplinas de programação subsequentes.

Um semestre após a mudança da linguagem de programação, uma melhora pouco sensível foi percebida. Assim, iniciou-se em 2011 um trabalho experimental com robótica educativa com alguns ingressantes, para que a disciplina pudesse absorver com o tempo o uso de recursos didáticos de apoio. Durante o acompanhamento dos discentes ficou evidente diversos fatores que influenciavam o seu desempenho, a exemplo da dificuldade em organizar o tempo de estudo em função das outras disciplinas; de não saberem pesquisar bons materiais na Internet; de dificuldades de socialização, entre outros. Desse modo, percebeu-se que a incorporação de novos recursos poderia não gerar resultados satisfatórios até que os alunos entendessem o contexto necessário para que um ambiente adequado de estudo existisse. Com base nisso, um processo de tutoria foi estabelecido, estando a sua descrição apresentada na Seção 4 deste documento.

### **3. Trabalhos Relacionados**

Uma atividade de tutoria se refere ao aconselhamento ou treinamento de um indivíduo a partir de alguém mais experiente. Dessa forma, um tutor pode ser definido como um guia ou conselheiro de confiança, que serve para amparar<sup>1</sup>. As experiências registradas no trabalho de D'Souza et al (2008) mostram que existem diferenças no modelo de atuação de tutores e monitores. Um tutor oferece atenção personalizada, ao passo que um monitor atua mais no contexto de grupo, onde, através de um modelo planejado de aula, certos objetivos pedagógicos podem ser alcançados. Em contraposição, uma sessão de tutoria deve ser um momento facilitador da aprendizagem e não deve limitar temas, de tal modo que o relato de experiências pessoais do tutor possa ser útil para a reflexão do tutelado em relação aos seus hábitos de estudo. Apesar de diferentes, os processos de tutoria e monitoria são complementares.

Diversos trabalhos na literatura registram experiências positivas no uso de sistemas de tutoria com alunos de cursos de Ciência da Computação no seu primeiro ano de graduação. Miliszewska (2007) apresenta como efeito do seu trabalho, um aumento na satisfação dos estudantes com o curso. Jenkins (2002) aponta diversos fatores comportamentais que influenciam a aprendizagem de programação e foram percebidos em função da tutoria, a exemplo de: questões de aptidão, motivação, fatores

---

<sup>1</sup><http://michaelis.uol.com.br/moderno/portugues/index.php?lingua=portugues-portugues&palavra=tutor>

cognitivos relacionados ao estilo de aprendizagem do aluno, o contato com novos interesses e a manipulação de múltiplas habilidades. Além do mais, segundo Klein (apud Reategui et al 2011), “outros estados afetivos negativos, como a confusão, raiva e ansiedade podem afetar a produtividade, a aprendizagem, as relações sociais e o bem-estar geral dos alunos”. Este é o caso de um comportamento comum em relação aos erros que são cometidos no processo de codificar os algoritmos.

Outros autores têm se voltado para o desenvolvimento de tecnologias. Rocha et al (2011), menciona o uso de sistemas tutores inteligentes. Já (Traynor 2004), menciona tecnologias adaptativas, que se baseiam em padrões de aprendizagem, capturados a partir de técnicas capazes de moldar o perfil dos alunos e, também, de modelos capazes de medir a sua motivação (Biggs, 2001). Todavia, o suporte aos estudantes de programação nas suas primeiras experiências de aprendizagem é um dos fatores mais importantes para os resultados de aprendizagem. Como cita D’Souza et al (2008), enquanto muitos recursos tecnológicos são produzidos na tentativa de amenizar as dificuldades de aprendizagem, as questões relacionadas à influência dos hábitos escolares do estudante nesse processo ainda é tratada com negligência.

O trabalho descrito neste artigo inspirou-se na importância de assistir os novatos e de acompanhá-los no processo de engajamento em um novo sistema, que é a universidade. Este trabalho registra uma experiência que envolveu estudantes veteranos da Licenciatura em Computação no papel de tutores, e que resultou em um processo de tutoria, passível de ser repetido em outros cursos. Além disso, o processo permitiu que para cada dificuldade identificada, um recurso pedagógico de natureza específica fosse utilizado.

#### **4. O processo de tutoria: relato de experiência**

Para desenvolver o trabalho da tutoria utilizou-se a metodologia *pesquisa-ação* que consiste no participante ser conduzido à produção do próprio conhecimento e se tornar o sujeito dessa produção. A metodologia *pesquisa-ação* busca superar, essencialmente, a separação entre o conhecimento teórico e a ação, buscando realizar a prática de conhecer para atuar. Neste caso, considerou-se também como participante os estudantes que exerciam o papel de tutores.

O processo de tutoria se constituiu como um processo iterativo em que, à medida que os tutores reconheciam as dificuldades do tutelado, uma intervenção era planejada. O processo foi organizado em pequenos ciclos semanais, constituído das etapas de planejamento, preparação da sessão de tutoria, sessão de tutoria e avaliação.

A assistência aos ingressantes foi iniciada no final do primeiro mês de aulas da disciplina, para que os primeiros contatos entre tutor e tutelado pudessem identificar alguns problemas que estivessem se instalando e para que fosse possível reconhecer a rotina do tutelado. Feito isto, a primeira etapa da tutoria, o planejamento, passou a ser o momento em que o tutor refletia sobre o contato anterior com o tutelado e identificava a necessidade de uma intervenção para auxiliá-lo. O tutor com frequência se informava sobre os conteúdos que estavam sendo abordados pelo professor da disciplina.

Considerou-se como sendo uma intervenção, uma visita à biblioteca; a demonstração de como o tutelado poderia buscar recursos na Internet que apoiassem seus estudos (listas de discussão de Python, uso de aplicações como o Google Tradutor, que auxiliam a tradução de materiais, etc) ou apenas um diálogo em que o tutor relatava suas experiências quando estava aprendendo a programar. Uma intervenção também

podia ser uma atividade que ajudasse o tutelado a superar dificuldades técnicas de programação. Para este tipo de intervenção, o tutor foi estimulado a pensar em atividades que exercitassem o pensamento computacional; a utilizar objetos de aprendizagem; a apresentar ferramentas para a construção de código com maior suporte à representação visual ou simplesmente resolver as listas de exercício em parceria com o tutelado.

Depois de identificar as dificuldades dos tutelados, os tutores iniciavam a etapa de preparação da sessão de tutoria, já que as intervenções poderiam requerer a investigação de diferentes recursos (jogos, o uso de *kits* e simuladores de robótica, ferramentas de programação); a resolução antecipada de listas de exercício ou a produção de materiais concretos, caso aplicado quando se planejava o uso de *atividades desplugadas*, por exemplo, que seguem as ideias contidas no livro “Ciência da Computação Desplugada” (CC 2005).

Após a preparação, a sessão de tutoria acontecia. Durante uma hora por semana os tutelados eram incentivados, acolhidos e/ou assistidos pelo tutor através da intervenção preparada. Ao final de cada sessão o tutor registrava em um diário de bordo o contato com o seu tutelado, incluindo uma impressão sobre a eficiência da intervenção utilizada e das novas dificuldades detectadas, quando este era o caso. Em alguns momentos também era realizada a autoavaliação dos tutelados e da tutoria.

Pode-se dizer que, em geral, a carga horária semanal de trabalho dos tutores resultou em uma média de quatro horas. Os seguintes *stakeholders* colaboraram com a tutoria:

- i. Tutores: estudantes mais experientes e que se comprometeram durante um semestre com um ingressante do curso;
- ii. Tutelados: estudantes ingressantes com nenhuma experiência em programação;
- iii. Investigadores: professor responsável por analisar os diários de bordo e relacionar o desempenho dos tutelados nas avaliações realizadas pelo professor da disciplina;
- iv. Professor da disciplina, elemento-chave para o estímulo às sessões de tutoria e a manutenção da comunicação sobre os ingressantes.

Para efeito de refinamentos e verificação do processo de aprendizagem, um experimento foi realizado, o qual se encontra descrito na Seção 5.

## 5. Descrição do experimento

O experimento realizado aconteceu com a turma de ingressantes da disciplina Introdução à Programação, que tem carga horária de 60 horas e utiliza Python como linguagem<sup>2</sup>. Quarenta e cinco alunos estavam matriculados na turma. A grande maioria havia cursado o ensino médio em escolas públicas. A turma era predominantemente do sexo masculino (80%) e a maioria demonstrava muito contato com tecnologias Web, como é o caso das redes sociais. Contudo, ao serem questionados sobre as atribuições de um profissional da área de Computação não sabiam com certeza reconhecer os mais diferentes perfis de formação superior.

---

<sup>2</sup> A ementa da disciplina pode ser consultada na página do curso (link omitido por questões de confidencialidade).

O processo utilizado na tutoria foi realizado com um grupo de controle (tutelados), que foi estabelecido da seguinte maneira: toda a turma foi convidada a participar de um teste de raciocínio lógico (em que se fez uso de quatro objetos de aprendizagem), teste no qual foram avaliados critérios relacionados à habilidade, rapidez e compreensão lógica dos participantes. O nível de desempenho dos participantes durante a avaliação foi classificado como alto, mediano e baixo. Para a seleção da amostra considerou-se um grupo de dez alunos (8 homens e 2 mulheres), que obtiveram o seguinte desempenho: alto (3 alunos), mediano (3 alunos) e baixo (4 alunos). Ao fazer a seleção, pode-se dizer que a distribuição da amostra, que representou 25% do universo, refletia as características da turma.

O grupo de tutores foi composto por dez alunos do curso, bolsistas de um projeto de incentivo à docência, onde cada um deles ficou responsável pelo acompanhamento de um participante do grupo de controle. O acompanhamento e supervisão dos tutores aconteceram através de um professor do curso.

O projeto foi apresentado e divulgado ao público-alvo a partir da segunda semana de aulas e as sessões de tutoria iniciaram na quarta semana. A tutoria durou cerca de três meses (do final de agosto a dezembro de 2011). Foram realizados em média quinze encontros por tutor. A Tabela 1 apresenta um resumo das principais dificuldades de aprendizagem identificadas, assim como, um breve relato do tipo de intervenção aplicado pelo tutor, quando a dificuldade identificada era técnica.

**Tabela 1. Amostra de atividades executadas em sessões da tutoria**

<b>Dificuldade identificada</b>	<b>Intervenção</b>	<b>Natureza</b>
Identificar elementos de um algoritmo	Atividade do livro Computação desplugada - <u>Seguindo instruções</u> (atividade ajustada) – além do conceito de algoritmo foi introduzido o conceito de variável.	Pensamento computacional
Entender o conceito de variável e alocação de memória em Python – capacidade de abstração de conceitos	Atividade desplugada criada pela equipe - demonstração do fluxo de execução de um código e a necessidade do acesso à memória.  Uso da ferramenta Jelliot, utilizada para demonstrar testes de mesa.	Pensamento computacional e uso de Objetos de Aprendizagem
Analisar um problema e extrair dados para montar a sua solução	Uso do método Qualifica (Mattos 2007) e produção de um formulário, em que questões guiam a extração de informações para a resolução do problema, a partir do enunciado da questão.	Apresentação de uma técnica para extração de problemas
Entender a execução dos fluxos de controle da estrutura IF	Uso do kit de robótica educacional PNCA e do jogo LightBot.	Uso de robótica e objetos de aprendizagem
Recursividade	Atividade desplugada criada pela equipe - foi utilizada uma escada para abstração sobre o empilhamento das chamadas das funções recursivas.	Pensamento computacional

Vale mencionar que algumas sessões foram realizadas em grupo. A Figura 2 registra duas atividades. Em uma delas foi explorado o uso de robótica educativa para estimular a comunicação entre os tutelados. A outra atividade realizou um pequeno

campeonato de programação, fazendo uso do jogo LightBot<sup>3</sup>, para praticar habilidades de programação e indiretamente, a capacidade de analisar problemas. Os tutores também organizaram uma visita técnica a um laboratório de robótica existente na cidade.



**Figura 2: Atividade em grupo com kits de robótica e com o jogo LightBot**

Todos os artefatos produzidos na tutoria (registro das sessões, planos de aula, diários de bordo, roteiro para preparação das atividades de computação desplugada, dados das avaliações, bibliografia utilizada, objetos de aprendizagem utilizados e fotos) estão disponíveis para download no endereço web [http://www2.ccae.ufpb.br/pibid/?page\\_id=156](http://www2.ccae.ufpb.br/pibid/?page_id=156).

### **5.1. Resultados do experimento**

Dos quarenta e cinco alunos matriculados na disciplina, treze alunos obtiveram aprovação por média. O resultado final da disciplina mostrou que no semestre em que a tutoria ocorreu, a disciplina alcançou a taxa de aprovação de 38%, tendo obtido êxito dezessete alunos (dos quais, quatro foram aprovados através de exame final). Historicamente, nos últimos três anos, a média de aprovação na disciplina é de 25%. Assim, percebe-se uma melhora, considerada satisfatória, que pode ter sido resultante do canal de comunicação aberto pelos tutelados com o professor da disciplina. Em função dos tutores relatarem com mais frequência o que estava acontecendo com o grupo de controle, o docente passou a perceber as dificuldades mais rapidamente e assim, conseguiu ajustar algumas práticas em sala de aula.

Em relação ao abandono, a partir do número de reprovados tem-se que onze alunos abandonaram a disciplina no decorrer do período. Isto representa também uma taxa de permanência muito superior aos semestres anteriores. Sendo assim, pode-se supor que a melhora registrada no contexto geral da disciplina pode ser decorrente do processo de assistência que alguns alunos receberam.

Em relação aos alunos que conseguiram aprovação na disciplina e fizeram parte do grupo de controle tem-se o seguinte panorama. Dos quatro alunos que foram classificados com baixo desempenho no teste lógico, dois foram aprovados na disciplina, o que representa 50%. Dos três alunos que obtiveram classificação mediada,

<sup>3</sup> <http://armorgames.com/play/2205/light-bot>

dois foram aprovados. Ao se considerar os alunos que obtiveram aprovação por média (13 alunos), dentre eles, quatro participaram da tutoria. Isto significa que a taxa de sucesso para o grupo de controle foi de 40%.

Fazendo uma análise das notas da turma, percebeu-se que a média dos alunos que não participaram do experimento foi 4.0, enquanto a média dos alunos participantes da tutoria foi de 5.0. Estes dados apontam que os resultados de aprendizagem do grupo de controle foi 25% melhor.

Todos os alunos do grupo de controle participaram com assiduidade das sessões de tutoria. Os alunos do grupo de controle avaliaram como muito importante o acompanhamento que tiveram e apontaram como melhores intervenções (técnicas) aquelas em que houve a aplicação de alguma atividade desplugada.

## **6. Considerações finais**

Melhorar o ensino de programação ainda será um desafio de pesquisa para os próximos anos. Neste processo é preciso dispor de ferramentas melhores para a construção de código, tecnologias de mediação mais eficientes para lidar com as emoções humanas, mas, sobretudo, é preciso melhorar a assistência aos ingressantes que chegam à universidade.

A experiência relatada com este modelo de tutoria foi resultado de várias tentativas para contribuir com o ensino de programação. Um aspecto interessante do processo foi a percepção de que, o aluno que ingressa em cursos superiores, em especial nas universidades públicas, é carente de conhecimentos basilares mas, principalmente, de maturidade para superar os desafios que lhe serão impostos ao longo da sua formação. Embora, novos métodos de ensino e tecnologias educacionais provenham muito suporte, elas serão ineficazes sempre que os estudantes não receberem o reforço correto para que usufruam e persigam as oportunidades de aprendizagem que elas propõem.

Ao professor precisam ser oferecidas possibilidades para lidar com diferentes estilos de aprendizagem em uma turma de natureza heterogênea e de permitir um sistema de avaliação que aponte as barreiras que vão se estabelecendo no processo de aprendizagem, para que estas não se transformem em elementos que limitem o interesse do aluno em continuar estudando. As experiências que vêm sendo desenvolvidas mostram que o acompanhamento individualizado dos iniciantes em programação é um dos fatores decisivos para que possam aprender a superar desafios, reconhecer a sua capacidade de aprender conceitos novos e de melhorar a sua autoestima, já que em algum momento eles se questionam sobre a sua capacidade cognitiva.

Como contribuição deste trabalho, pode-se destacar além do suporte fornecido aos tutelados, o fortalecimento da formação técnica e pedagógica dos tutores, já que tiveram a oportunidade de experimentar a docência em várias dimensões, até mesmo, desenvolvendo uma perspectiva mais humanizada do processo. Como resultados diretos, os tutores tiveram a chance de melhorar a sua capacidade de comunicação em público, de articulação, de expressão (ao terem que desenvolver uma linguagem mais apropriada para que se fizessem compreender pelos novatos) e, também, de aprimoramento dos conhecimentos técnicos sobre a programação. Neste caso específico, registra-se que alguns tutores assumiram uma postura extremamente reflexiva e inventiva para resolver as dificuldades técnicas dos seus tutelados. Foi o que aconteceu quando produziram diversas novas atividades desplugadas.

Uma grande contribuição deste trabalho é que o processo que conduziu a tutoria foi relatado de uma forma que pode servir como um modelo para que outros cursos reproduzam as atividades realizadas, a partir, é óbvio, de ajustes necessários provenientes das peculiaridades de cada curso. O fato de todo o material está disponível é algo que também auxiliará os interessados. Para os cursos de Licenciatura em Computação, em especial, este serviço de tutoria abre um espaço extremamente importante para as práticas de ensino de seus alunos, assim como, para os estágios supervisionados obrigatórios, que podem fomentar, através dos estagiários, o atendimento a uma grande parcela dos ingressantes. Este também pode ser um excelente espaço para a observação, diagnóstico-ação e regência de aulas de ensino de Computação, ao se considerar que os ingressantes são alunos que saíram há pouco tempo do ensino médio, e, portanto, ainda podem ser vistos como tal. Aos demais cursos de Computação, muitas alternativas podem compor o corpo de tutores, sendo um deles um redirecionamento do trabalho dos monitores ou da composição de um grupo de alunos voluntários para o projeto.

Vale salientar que atualmente o serviço de tutoria foi expandido para todos os ingressantes da disciplina Introdução à Programação através de um ajuste no modelo, que se tornou semipresencial, e que utiliza atualmente uma plataforma de Educação à Distância para estabelecer o canal de acompanhamento com os tutelados. Os relatos provenientes da tutoria têm alimentado a coordenação do curso, que tem formado grupos de trabalho para estabelecer estratégias de contenção para as dificuldades detectadas. Uma delas, inclusive, se refere ao ajuste nas ementas das disciplinas Cálculo I e Cálculo II, que têm incorporado diversos conteúdos da Matemática do ensino médio, que nunca tinham sido ensinados para grande parte dos ingressantes.

## **Agradecimentos**

As atividades deste projeto foram realizadas com o apoio da equipe do Programa de Incentivo à Iniciação à Docência (PIBID – Computação). Portanto, nossos agradecimentos à Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – CAPES.

## **Referências**

- Ala-Mutka K. (2004) “Problems in Learning and Teaching Programming-A literature Study for Developing Visualizations in the Codewitz-Minerva Project,” Codewitz Needs Analysis, pp. 1-13.
- Biggs, J., Kember, D. & Leung D.Y.P. (2001), The revised two-factor Study Process Questionnaire: RSPQ-2F , British Journal of Educational Psychology.Vol 71(1), pp. 133–149.
- Cunha V.V., Klemann M., Vicari R. (2006) “Estratégias pedagógicas no ensino de algoritmos e programação associadas ao uso de jogos educacionais,” Novas Tecnologias na Educação, vol. 4, 2006, p. 1–10.
- Computer Science Unplugged. (2005) Livro digital disponível em <http://csunplugged.org/>
- D'Souza D., Hamilton M., Harland J., Muir P., Thevathayan C., Walker C. (2008) Transforming learning of programming: a mentoring Project. Proceeding ACE '08 Proceedings of the tenth conference on Australasian computing education - Volume 78.

- Jenkins, T. (2001) The motivation of students of programming. In Proceedings of ITiCSE 2001: The 6th annual conference on innovation and technology in computer science education (pp. 53–56).
- Jenkins, T. (2002), On the Difficulty of Learning to Program Proceedings of the 3rd Annual Conference of the LTSN Centre for Information and Computer Sciences pp. 53.
- Kelleher C., Pausch R. (2005) “Lowering the barriers to programming: A taxonomy of programming environments and languages for novice programmers,” ACM Computing Surveys (CSUR), vol. 37, p. 83–137.
- Kinnunen, P. and Malmi, L. (2006) “Why students drop out CS1 course?” In: Proceedings of the Second international Workshop on Computing Education Research (Canterbury, United Kingdom). ICER '06.ACM, New York, NY, 97-108.
- Miller, J. A. (2004) “Promoting Computer Literacy through Programming Python”. Tese de Doutorado - Universidade de Michigan, EUA.
- Mattos M. M., Fuchs J. F. (2007) Qualifica: Uma Ferramenta para Apoio a Construção de Algoritmos Estruturados. Anais do XVI Seminário de Computação (SEMINCO). Blumenau – SC.
- Mendonça, A. P. (2010) Programação Orientada ao Problema: uma Metodologia para Entendimento de Problemas e Especificação no Contexto de Ensino de Programação para Iniciantes. 2010. Tese (doutorado) – Universidade Federal de Campina Grande, Campina Grande.
- Miliszewska, I., Tan G. (2007), Befriending Computer Programming: A Proposed Approach to Teaching Introductory Programming, Issues in Informing Science and Information Technology Vol 4, pp. 278--289, 2007.
- Sudol L.A. (2011) “Deepening Students’ Understanding of Algorithms: Effects of Problem Context and Feedback Regarding Algorithmic Abstraction,” Carnegie Mellon University.
- Reategui E., Bercht M., Iepsen E. F. (2011) Detecção e Tratamento do Estado Afetivo Frustração do Aluno na Disciplina de Algoritmos. XXII Simpósio Brasileiro de Informática na Educação. XVII Workshop de Informática na Educação. Aracaju-SE. ISSN:2176-4301.
- Rocha P.S., Ferreira B., Monteiro D., Nunes D.S.C., Góes H.C.N. (2011) “Ensino e Aprendizagem de Programação: Análise da Aplicação de Proposta Metodológica Baseada no Sistema Personalizado de Ensino,” RENOTE, vol. 8, 2011.
- Rosa N. S, Rapkiewicz, C. E.; Manhães, J.; Pereira Júnior, J. C. R. (2006) “Jogos Educacionais como ferramentas para o ensino de programação.” In: XIV Workshop sobre Educação em Computação - WEI, 2006, Campo Grande, MS, 2006.
- Teague D. (2009) A people-first approach to programming. Proceeding ACE '09 Proceedings of the Eleventh Australasian Conference on Computing Education - Volume 95.
- Traynor, D. & Gibson, J.P. (2004), Implementing Cognitive Modelling in CS Education: Aligning theory and practice of learning to Program , IADIS International Conference CELDA 04, 15th - 17th December, 2004.