

# Ensino de Algoritmos e Linguagem de Programação no Nível Médio: Um Relato de Experiência

Rivanilson da Silva Rodrigues<sup>1</sup>, Luiz Augusto de Macêdo Moraes<sup>1</sup>, Sormany Silva Dantas<sup>1</sup>, José Gomes Lopes Filho<sup>1</sup>, Cheyenne Ribeiro Guedes Isidro Abílio<sup>1</sup>, Pablo Ribeiro Suárez<sup>1</sup>

<sup>1</sup>Departamento de Computação – Universidade Estadual da Paraíba (UEPB) Campus VII - Rua Alfredo Lustosa Cabral, s/n, CEP: 58.706-560, Patos-PB

{rivanilson, luizaugustomm, sormany, charibeiro, prsuarez}@gmail.com, zefilho@msn.com

**Abstract.** *Considering the high dropout in courses related to technology and demand for professionals in this area, it was decided to introduce notions of algorithms and programming to high school students from public schools in the town of Patos - PB. Thus, this paper proposes to report the experiences during the implementation of a course introduction to algorithms and programming language. The students used the Python programming language while introducing basic concepts of structured programming. The conclusion, the course was satisfactory and getting a dropout rate of 46%, a rate lower than similar courses.*

**Resumo.** *Diante das altas taxas de evasão dos cursos ligados à tecnologia e demanda por profissionais nessa área, decidiu-se introduzir noções de algoritmos e programação a alunos do Ensino Médio provenientes de escolas públicas do município de Patos – PB. Assim, esse artigo propõe relatar as experiências obtidas durante a aplicação de um mini-curso de introdução a algoritmos e linguagem de programação. Os alunos participantes utilizaram a linguagem de programação Python durante a introdução de conceitos básicos da programação estruturada. Ao final, o mini-curso foi considerado satisfatório atendendo a uma demanda de 78 alunos do Ensino Médio e obtendo uma taxa de evasão de 46%, índice inferior a cursos similares.*

## 1. Introdução

Segundo Barros (2011), em 2020 o Brasil contará com um déficit de 750 mil profissionais de tecnologia da informação e comunicação. As causas para a carência desses profissionais vão desde o baixo interesse dos estudantes brasileiros por ciências exatas até a alta evasão dos alunos nos cursos ligados à tecnologia [BARROS, 2011].

Considerando este cenário, o projeto de extensão universitária – intitulado “Hermes: Prática Pedagógica para a Formação Contínua do Futuro Licenciado em Computação e para a Divulgação do Curso de Licenciatura Plena em Computação” – objetivou divulgar e explicitar as bases da formação do Licenciado em Computação da Universidade Estadual da Paraíba - Campus VII nas escolas públicas do município de Patos – PB a fim de atrair potenciais aprendizes oriundos do Ensino Médio para o curso em questão e diminuir o seu alto índice de evasão.

Para o cumprimento dos objetivos do projeto, foi considerada a idéia de inserir conteúdos provenientes da área da computação no ensino Médio proposta pelo GT-3

[PEREIRA JÚNIOR *et al*, 2005]. Partindo dessa idéia, decidiu-se oferecer um mini-curso de introdução a algoritmos e programação a alunos do Ensino Médio utilizando a linguagem de programação Python, que atualmente tem atraído programadores principalmente pela sua simplicidade [LUTZ, 2010].

Assim, o presente artigo propõe relatar as experiências obtidas durante a vigência de um mini-curso de introdução a algoritmos e linguagem de programação para alunos do ensino Médio de escolas públicas do município de Patos – PB. O intuito é o de contribuir para identificação de alguns pontos de dificuldade a serem superados no processo de ensino-aprendizagem de algoritmos e linguagem de programação, bem como motivar o desenvolvimento de iniciativas similares em outras localidades.

## **2. Educação em Computação**

Considerando o avanço tecnológico inerente a sociedade, disciplinas como Ciência da Computação e programação de computadores gradativamente estão sendo incorporadas nas matrizes curriculares do ensino Médio de países europeus desde meados de 2004. Tais conhecimentos são importantes mesmo para estudantes que não se interessam por estas áreas, pois eles podem se beneficiar desenvolvendo novas habilidades como, por exemplo, raciocínio lógico, resolução de problemas e pensamento algorítmico [GRANDELL *et al*, 2006].

Apesar disso, a programação propriamente dita é uma disciplina que nem sempre é vista com bons olhos por alunos que ingressam em cursos superiores na área de informática [DANTAS *et al*, 2010]. Isso ocorre frequentemente porque a aprendizagem de conceitos e métodos para o desenvolvimento de programas de computador não é trivial e muitas vezes requer altos níveis de abstração [CASTRO *et al*, 2009].

Castro e Furks (2011) argumentam que, em geral, os cursos de introdução a programação têm turmas grandes e diante disso o professor muitas vezes não consegue acompanhar os aprendizes durante o processo de ensino-aprendizagem. Tais cursos têm em média 60% de evasão [ROCHA, 1993]. Assim, é evidente a necessidade de mais pesquisas nessa área do conhecimento tão complexa que é o ensino de programação.

## **3. Trabalhos Correlatos**

O estudo relatado em [PEREIRA JÚNIOR *et al*, 2005] buscava investigar a viabilidade do ensino de algoritmos e programação no nível Médio. Ao final do trabalho, o autor afirma que a avaliação dos concluintes da oficina permite projetar as expectativas de inclusão do tema algoritmo e programação na matriz curricular do ensino Médio.

Machado *et al* (2010) apresenta experiências obtidas em um projeto de extensão que tem por base desenvolver em alunos do ensino Fundamental e Médio o raciocínio lógico-matemático e algorítmico, de maneira que estes alunos possam compreender a utilização desses modelos de raciocínio para resolução de problemas. O critério de seleção dos alunos para participação no projeto foi o desempenho dos mesmos nas disciplinas de Matemática e Física. Todavia obtiveram níveis de evasão de até 69% [MACHADO *et al*, 2010].

Marques *et al* (2011) promoveram uma oficina para alunos do Ensino Médio utilizando jogos como recurso motivacional. Os jogos utilizados como atrativos foram

desenvolvidos na linguagem de programação Python, usando a biblioteca para jogos *PyGame*. Segundo os autores foram disponibilizadas 25 vagas para os alunos da rede pública de ensino. O processo de seleção destes alunos levou em consideração apenas o interesse deles em participar do curso em questão.

#### **4. Utilização da Linguagem Python no Aprendizado de Programação**

A escolha pela linguagem Python foi motivada principalmente por algumas vantagens em relação a outras linguagens, que serão descritas nesta seção. Segundo Grandell *et al* (2006), Python é uma linguagem de alto nível interpretada, originalmente projetada por Guido Van Rossum para ser facilmente aprendida. Van Rossum (1999) sugere que qualquer pessoa pode se tornar mestre em programação usando Python. Portanto, pretendeu-se no mini-curso explorar os recursos dessa linguagem visando facilitar o primeiro contato com a programação.

#### **5. Metodologia**

A metodologia empregada para realização deste trabalho foi baseada em Dantas (2010), e consistiu nas seguintes etapas: Planejamento e Projeto; Preparação e divulgação do mini curso; Aplicação; Avaliação e resultados.

Durante a etapa de Planejamento e Projeto foram realizadas reuniões entre os envolvidos no projeto (alunos do curso de Licenciatura em Computação da UEPB) para traçar objetivos e metas a serem alcançadas ao longo da vigência do mini-curso. Ao final desta etapa, foi elaborado um plano de curso com carga horária de 40 horas aula.

Já na Preparação e divulgação do mini curso iniciou-se o planejamento das aulas, juntamente com a divulgação do mini-curso. A divulgação foi feita primeiramente em duas escolas públicas da cidade de Patos-PB.

O segundo momento da divulgação aconteceu no laboratório de informática da universidade onde foi apresentado um seminário sobre o curso de Licenciatura em Computação, em consequente foi apresentado o plano de curso elaborado pelos membros do projeto e a sua carga horária. Nesse momento, também foi esclarecido que os alunos participantes do mini-curso seriam submetidos a uma avaliação continuada e ao final do curso aqueles que obtivessem desempenho final superior ou igual a 7,0 ganhariam um certificado de participação.

O curso foi oferecido apenas para alunos que demonstraram interesse após a divulgação e participação do seminário. Além disso, no ato da inscrição os alunos puderam escolher entre dois horários para as aulas, o que ocasionou a abertura de duas turmas para atender a demanda. Por fim, na avaliação e resultados, os resultados analisados nesse trabalho são provenientes de dados obtidos ao longo da vigência do mini-curso.

#### **6. Planejamento e Progressão do Mini-Curso**

O mini-curso foi realizado entre os meses de Setembro e Novembro de 2011, as aulas tiveram carga horária de 2 horas aula e foram homologadas 78 inscrições. O curso foi organizado em duas unidades, a primeira destinada ao aprendizado de algoritmos (Unidade Teórica) e a segunda destinada ao aprendizado de programação (Unidade Prática). Foi pressuposto que todos os participantes inscritos não tivessem conhecimentos sobre os assuntos relacionados ao mini-curso.

Na unidade teórica, optou-se por uma abordagem de ensino *bottom-up* [SOUZA, 2009], começando pela importância dos algoritmos para Ciência da Computação e programação de computadores abordando primeiramente os seguintes assuntos: tipos de dados, constantes, variáveis, expressões aritméticas e lógicas, e posteriormente estruturas de controle de fluxo e de repetição.

Para as aulas da primeira unidade foi criada uma variação do português estruturado descrito por Souza (2009) equivalente a linguagem de programação Python traduzida. O objetivo era facilitar a migração do pseudocódigo estudado na primeira unidade do curso para a aplicação na linguagem de programação na segunda unidade. Nesta unidade foi estimulado que os alunos praticassem o “teste de mesa” onde o estudante percorre o programa fazendo o papel do computador [CRISTOVÃO, 2008; DANTAS, 2010].

A segunda unidade (Unidade Prática) foi desenvolvida de forma mais dinâmica, pois os alunos puderam aplicar conhecimentos desenvolvidos na primeira unidade. Os conteúdos pertinentes a esta unidade foram funções da linguagem Python para manipulação de *strings*; modularização: funções; estruturas de dados e suas funções de manipulação: listas e dicionários.

Em ambas as unidades as aulas foram ministradas no laboratório de informática da Universidade. Cada aula contou com um professor e monitores para auxiliar os alunos durante a realização de atividades. Além disso, os exemplos, exercícios, mini testes e demais avaliações utilizadas em sala de aula foram criadas com cuidado para que os conteúdos abordados fossem sempre contextualizados para realidade dos alunos. Todas as aulas contaram com exemplos e exercícios práticos realizados gradativamente com os alunos.

## **7. Resultados e Discussão**

Ao aplicar as práticas apresentadas nas seções anteriores, observou-se que durante a divulgação do mini-curso houve boa aceitação por parte dos alunos. Constatou-se que 120 alunos demonstraram interesse pelo mini-curso e foram previamente inscritos. Destes, 45% eram alunos matriculados no 1º ano do ensino Médio, 36% foram alunos do 2º ano e 19%, alunos do 3º ano.

O desempenho individual de cada aluno foi analisado com base em sua média em cada unidade. Por meio dessa análise, foi possível verificar claramente que houve um melhor desempenho pela maior parte dos estudantes durante a segunda unidade do curso, sendo que apenas 14,28% destes obtiveram resultados abaixo da primeira unidade. Estes resultados, de certa forma, eram esperados, tendo em vista o nível de abstração necessário para o entendimento de algoritmos em uma abordagem tradicional de ensino, ou seja, em uma abordagem puramente teórica.

Diante desses dados foi possível analisar que 40,47% dos alunos concluíram o curso com desempenho igual superior a 7,0. Com relação à frequência o índice de evasão do curso foi de 46%. Considerou-se evasão como sendo a desistência do curso, isto incluiu os alunos que se matricularam, mas nunca se apresentaram no curso, ou que faltaram a mais de 30% do curso.

## **8. Visão dos Instrutores**

Alguns pontos positivos e negativos foram evidenciados pelos instrutores durante a

vigência do mini-curso e são importantes para este estudo. Entre os pontos positivos:

*Avaliação continuada:* a aplicação de um modelo de avaliação continuada da aprendizagem permitiu aos instrutores analisarem o desempenho dos estudantes a cada conteúdo e verificar os principais obstáculos intrínsecos ao processo de ensino-aprendizagem de algoritmos e programação. *Linguagem de Programação Python:* a linguagem de programação Python foi bem aceita pelos alunos, estes não tiveram dificuldade em aprender a sua sintaxe.

*Abordagem Prática:* A abordagem prática da segunda unidade do curso foi fundamental, pois os alunos puderam exercitar os conceitos aprendidos na primeira unidade e principalmente resolver problemas usando a linguagem de programação Python. Durante as aulas práticas os alunos mostraram-se mais interessados e tiveram melhor desempenho que nas aulas teóricas. Contudo, alguns alunos ainda tinham dificuldades em resolver parte dos problemas com auxílio do computador. Como esperado.

*Certificados e Evasão:* Altos índices de evasão eram esperados durante a vigência do mini-curso. No entanto, o índice de 46% obtido ao final do mini-curso pode ser considerado satisfatório em um primeiro momento. A oferta de certificados aos alunos foi um bom atrativo e incentivo para os alunos, e possivelmente contribuiu para a diminuição dos índices de evasão.

Entre os pontos negativos foi evidenciado principalmente:

*Algoritmos em pseudocódigo:* durante as aulas teóricas utilizando pseudocódigo os alunos tiveram muita dificuldade em compreender e resolver a maioria dos problemas. *Dificuldade de abstração dos alunos:* Mesmo passando pela primeira unidade alguns estudantes ainda tinham dificuldade em desenvolver soluções em programas para alguns problemas, possivelmente pelo fato de os conteúdos relacionados à resolução de problemas de forma algorítmica exigir um nível de abstração ainda pouco estimulado no ensino Médio.

*Teste de mesa:* o teste de mesa, muito comum em cursos introdutórios de programação nas universidades foi incentivado na primeira unidade, no entanto, se observou que essa prática não é viável se quisermos obter um bom desempenho com alunos que não tem conhecimento prático com programação.

Diante do exposto, é importante destacar que durante a experiência de instruir os jovens durante este mini-curso se observou que há um indicativo que utilizar métodos de ensino de programação *top-down* descrito em Souza (2009), onde desde o início se prioriza o uso do computador, tais métodos provavelmente sejam mais efetivos que adotar as práticas comuns de ensino.

## **9. Considerações Finais**

Através desse trabalho foi possível observar indícios de que ensinar algoritmos e programação para alunos de nível Médio é algo interessante e viável, se adotadas as práticas de ensino corretas. Conclui-se ainda que as práticas de ensino adotadas ao longo do mini-curso produziram conhecimento satisfatório acerca do tema em questão, podendo contribuir para trabalhos similares. Desse modo, pretende-se aplicar novamente esse mini-curso e obter resultados ainda melhores e mais expressivos, tendo em vista as experiências obtidas nesta pesquisa.

## Referencias

- Barros, F. (2010) “Déficit de profissionais de TI chegará a 750 mil vagas em 2020”. Disponível em: <http://convergenciadigital.uol.com.br/cgi/cgilua.exe/sys/start.htm?infoid=26160&sid=57>
- Castro, T. and Furks, H. (2011) “Sistematização da Aprendizagem de Programação em Grupo”. In: XVII Workshop de Educação em Informática (WEI), 2011, Aracaju. p. 1763-1773.
- Castro, T.; Furks, H. and Spósito. M. A. F.; Castro JR. A. N. (2009) “Análise de um Estudo de Caso para Aprendizagem de Programação em Grupo”. IEEE-RITA, v. 4, n 2, p. 155-160.
- Cristovão, H. M. “Aprendizagem de Algoritmos num Contexto Significativo e Motivador: Um Relato de Experiência” (2008). In: XXVIII Congresso da SBC, Belém do Pará, p. 30-40.
- Dantas, A. ; Lopes, D. ; Costa, L. F. S. ; Silva, M. A. A. (2010) “Aprendendo a Ensinar Programação Combinando Jogos e Python”. In: XXI Simpósio Brasileiro de Informática na Educação, João Pessoa.
- Gomes, A.; Henriques, J.; Mendes, A. J. (2008) “Uma proposta para ajudar alunos com dificuldades na aprendizagem inicial de programação de computadores”. Educação, Formação & Tecnologias, v. 1, Maio.
- Grandell, L.; Peltomaki, M.; Back, R. B.; Salakoski, T. (2006) “Why Complicate Things? Introducing Programming in High School Using Python”. In: VIII Australasian Computing Education Conference, 2006, Hobart. Proceedings... Hobart.
- Lutz, M. (2010) “Programming Python”. 4. ed.[S.l]: O'Reilly Media.
- Machado, E. Z. A. ; Vasconcelos, I. R. S. ; Amorim, K. M. ; Adrade, A. M. S. ; Barreto, L. P. and Abdalla, D. (2010) “Uma Experiência em Escolas de Ensino Médio e Fundamental para a Descoberta de Jovens Talentos em Computação”. In: Workshop de Educação em Informática, Belo Horizonte.
- Marques, L. M.; Costa, L. F. S.; Silva, M. A. A. and Rebouças, A. D. D. S. (2011) “Atraindo Alunos do Ensino Médio para a Computação: Uma Experiência Prática de Introdução a Programação utilizando Jogos e Python”. In: XVII Workshop de Educação em Informática.
- Pereira Júnior, J.; Rapkiewicz, C. E.; Delgado, C. and Xexeo, J. A. M. (2005) “Ensino de Algoritmos e Programação: Uma Experiência no Nível Médio”. In: XXV Congresso da Sociedade Brasileira de Computação, São Leopoldo.
- Rocha, H. V. (1993) “Representações Computacionais Auxiliares de Conceitos de Programação”. Disponível em: <Acesso em: 5 jul. 2012.
- Souza, C. M. (1999) “VisuAlg – Ferramenta de Apoio ao Ensino de Programação”. Revista TECCEN, [S.l], v. 2, n. 2, p. 10 , Setembro.
- Van Rossum, G. (1999) “Computer Programming for Everybody”. Disponível em: <<http://www.python.org/doc/essays/cp4e.html>> Acesso em: 5 jul. 2012.