



**UNIVERSIDADE ESTADUAL DA PARAÍBA  
CAMPUS VII  
CENTRO DE CIÊNCIAS EXATAS E SOCIAIS APLICADAS  
CURSO DE GRADUAÇÃO EM LICENCIATURA PLENA EM COMPUTAÇÃO**

**RIVANILSON DA SILVA RODRIGUES**

**ENSINO DE ALGORITMOS E LINGUAGEM DE  
PROGRAMAÇÃO NO NÍVEL MÉDIO: UM RELATO DE  
EXPERIÊNCIA**

**PATOS – PB  
2013**

**RIVANILSON DA SILVA RODRIGUES**

**ENSINO DE ALGORITMOS E LINGUAGEM DE  
PROGRAMAÇÃO NO NÍVEL MÉDIO: UM RELATO DE  
EXPERIÊNCIA**

Trabalho de Conclusão de Curso apresentado ao  
Curso de Graduação em Licenciatura Plena em  
Computação da Universidade Estadual da Paraíba,  
em cumprimento à exigência para obtenção do grau  
de Licenciado em Computação.

Orientadora: Cheyenne Ribeiro Guedes Isidro  
Abilio

PATOS – PB  
2013

UEPB - SIB - Setorial - Campus VII

R696e Rodrigues, Rivanilson da Silva.  
Ensino de algoritmos e linguagem de programação no nível médio: um relato de experiência [manuscrito] / Rivanilson da Silva Rodrigues. – 2013.  
17 f. : il.

Digitado.

Trabalho de Conclusão de Curso (Licenciatura em Computação) – Centro de Ciências Exatas e Sociais Aplicadas, Universidade Estadual da Paraíba, 2013.

“Orientação: Prof<sup>ª</sup>. Msc. Cheynne Ribeiro Guedes Isidro Abilio”.

1. Algoritmos. 2. Linguagem de programação. 3. Ensino de informática. I. Título.

21. ed. CDD: 005.73

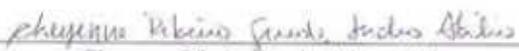
Rivanilson da Silva Rodrigues

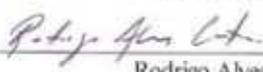
**Ensino de Algoritmos e Linguagem de Programação no Nível Médio: um  
Relato de Experiência**

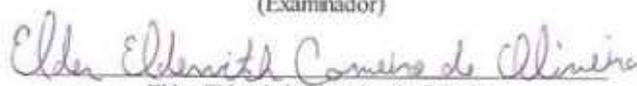
Trabalho de Conclusão de Curso apresentado ao  
Curso de Licenciatura em Computação da  
Universidade Estadual da Paraíba, em cumprimento  
à exigência para obtenção do grau de Licenciado em  
Computação

Aprovado em 18 de fevereiro de 2013

**BANCA EXAMINADORA**

  
\_\_\_\_\_  
Cheyenne Ribeiro Guedes Isidro Abílio  
(Orientadora)

  
\_\_\_\_\_  
Rodrigo Alves Costa  
(Examinador)

  
\_\_\_\_\_  
Elder Eldervitch Carneiro de Oliveira  
(Examinador)

# **ENSINO DE ALGORITMOS E LINGUAGEM DE PROGRAMAÇÃO NO NÍVEL MÉDIO: UM RELATO DE EXPERIÊNCIA**

RODRIGUES, Rivanilson da Silva<sup>1</sup>

## **RESUMO**

Considerando as altas taxas de evasão dos cursos ligados à tecnologia e a crescente demanda por profissionais qualificados nessa área, decidiu-se introduzir noções de algoritmos e programação a alunos do Ensino Médio, através de um projeto de extensão universitária como forma de incentivar desde cedo o ingresso de jovens nessa área. Assim, esse artigo propõe relatar as experiências obtidas durante a aplicação de um mini-curso de introdução a algoritmos e linguagem de programação para alunos do Ensino Médio provenientes de escolas públicas do município de Patos – PB. O curso contou com uma carga horária de 40 horas aula, dividido em duas unidades e os alunos participantes foram submetidos a um processo de avaliação continuada da aprendizagem. A linguagem de programação Python foi utilizada no curso para introduzir os conceitos básicos da programação estruturada de maneira simples. 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.

**PALAVRAS-CHAVE:** algoritmos, linguagem de programação, ensino

---

<sup>1</sup> rivanilson@gmail.com

## INTRODUÇÃO

O avanço da informática e a sua utilização em diversas atividades desenvolvidas pela sociedade, assim como o aumento da demanda por profissionais das áreas de tecnologia, sistemas e computação (AGUIARI, 2010; BARROS, 2011), constituem fatores determinantes para a incorporação da computação no processo educativo formal visando o desenvolvimento intelectual, motivacional e vocacional de alunos para as áreas científicas e tecnológicas.

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 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 (Grupo de Trabalho de Licenciatura em Computação-vinculado à SBC, responsável por discutir questões relativas aos cursos de Licenciatura em Computação) (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.

Este trabalho está organizado nas seguintes seções: a seção 1 contextualiza a pesquisa sobre a temática da educação em computação. A seção 2 apresenta alguns trabalhos correlacionados ao tema deste trabalho. A seção 3 mostra as vantagens da linguagem de programação Python em relação a outras linguagens utilizadas em cursos de introdução a programação. A seção 4 apresenta a metodologia adotada para o desenvolvimento deste trabalho. A seção 5 apresenta o planejamento e progressão do mini curso. A seção 6 apresenta os resultados obtidos ao final da pesquisa. A seção 7 apresenta a visão dos instrutores a cerca do mini-curso e por fim a seção 8 apresenta as considerações finais.

### 1. EDUCAÇÃO EM COMPUTAÇÃO

A utilização do computador na educação sempre foi um desafio. Durante a década de 50, quando os computadores com capacidade de programação começaram a ser comercializados, surgiram as primeiras iniciativas para o seu uso na Educação. Vale destacar a sua aplicação como máquina de ensinar no Centro de Pesquisa Watson da IBM e na Universidade de Illinois (RALSTON; MEEK, 1976, p. 272 apud VALENTE, 1999). Contudo, naquela época o computador era utilizado quase sempre como um simples transmissor de informações predeterminadas, já “hoje, a utilização de computadores na educação é muito mais diversificada, interessante e desafiadora, do que simplesmente a de transmitir informação ao aprendiz.” (VALENTE, 1999, p. 11). Atualmente sabe-se que o computador

pode ser utilizado para enriquecer ambientes de aprendizagem e auxiliar o estudante no processo de construção do seu conhecimento.

De acordo com Grandellet al(2006), o principal objetivo do ensino Fundamental e Médio é preparar o aluno com conhecimentos necessários para sua vida na sociedade. Segundo Nascimento (2009), a influência da informática na sociedade moderna e os seus reflexos na realidade escolar são notórios.

Considerando o avanço tecnológico incorporado na sociedade atual, algumas matérias que outrora foram consideradas exclusivas do nível Superior vêm sendo incorporadas nas camadas mais baixas de ensino, como é o caso da Ciência da Computação e da programação de computadores. Estas disciplinas gradativamente estão sendo incorporadas nas matrizes curriculares do ensino Médio de países europeus desde meados de 2004 (GRANDELL et al, 2006).

Para Valente (1999, p. 82), “o computador pode ajudar o processo de conceituação e o desenvolvimento de habilidades importantes para a sobrevivência na sociedade do conhecimento se é usado como um dispositivo para ser programado”. Nesse sentido, cada vez mais pesquisas têm mostrado que conhecimentos sobre programação de computadores não são importantes apenas para quem tem interesse nas áreas tecnológicas ou de Ciência da Computação. Mesmo estudantes que não se interessam por essas áreas podem se beneficiar desenvolvendo novas habilidades como, por exemplo, raciocínio lógico, resolução de problemas e pensamento algorítmico ao estudarem programação (GRANDELL et al, 2006).

Para Gomes et al (2008), o nível mais básico do ensino de programação tem como propósito conseguir que os alunos desenvolvam conhecimentos necessários para conceber programas capazes de resolver problemas reais simples. 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 freqüentemente 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).

Para Pereira Júnior et al(2005), o desenvolvimento de algoritmos e programas é uma das competências mais difíceis de serem desenvolvidas. Cristovão (2008) considera que a maior dificuldade enfrentada em um curso na área de computação é o ensino de algoritmos para os ingressantes. Tal dificuldade pode vir do fato de não existir uma metodologia ótima para o ensino nestes cursos. Contudo, para que se tenha um melhor aproveitamento nestes cursos é necessário que mais estudos que relatem experiências da aplicação de novas metodologias e abordagens de ensino relacionados ao tema em questão sejam realizados.

Segundo Souza (2009), a abordagem tradicional de ensino nesses cursos envolve conceitos básicos como tipos de dados, variáveis e constantes, comandos de entrada e saída de dados, operadores de atribuição, aritméticos, relacionais e lógicos. Nessa abordagem os componentes principais da programação estruturada também são apresentados utilizando geralmente uma forma de pseudocódigo que é intermediária entre o rigor das linguagens de programação reais e a liberdade da linguagem natural.

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).

Outros pontos destacados sobre as dificuldades no processo de ensino de programação pela perspectiva de Gomes et al(2008) são a inexistência de métodos de ensino focados no aluno e os maus hábitos desenvolvidos pelos alunos em abordagens de estudo voltadas para leitura repetitiva e memorização. O autor ainda reforça que assistir aulas e estudar um livro texto não é suficiente para o aluno aprender programação, segundo o mesmo, a aprendizagem

significativa envolveria um intenso trabalho extra classe.

Assim, é evidente a necessidade de mais pesquisas nessa área do conhecimento tão complexa que é o ensino de programação.

## 2. TRABALHOS CORRELATOS

Nesta seção apresentam-se alguns trabalhos que estão relacionados ao foco desta pesquisa.

O estudo relatado em (PEREIRA JÚNIOR et al, 2005) buscava investigar a viabilidade do ensino de algoritmos e programação no nível Médio. Nesse trabalho foi utilizada a linguagem de programação Pascal em uma oficina de Lógica de Programação com duração total de 36 horas divididas em 6 horas semanais. 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.

Outro trabalho relacionado ao tema é o de Garcia et al (2008), que descreve um curso de Lógica de Programação e Estrutura de Dados empregando a linguagem de programação C e voltado para alunos do Ensino Médio da rede pública e privada de ensino, com carga horária total de 228 horas, durante um período de 10 meses. Vale ressaltar que a seleção dos alunos para esse projeto levou em consideração o desempenho de cada um nas disciplinas de Matemática e Física. No entanto, dos 33 alunos que iniciaram o curso apenas 17 concluíram.

Cristovão (2008) apresenta em seu trabalho a ferramenta Scratch e o relato de uma abordagem com calouros do curso de Ciência da Computação na disciplina Algoritmos I da Unidade de Computação e Sistemas da FAESA em Vitória – ES. Segundo o autor, a turma participante do experimento tinha em média 60 alunos entre ingressantes, repetentes e transferidos. O software Scratch foi aplicado durante 1 mês e meio e segundo o mesmo, os resultados foram satisfatórios.

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. Para isso, utilizou a OBI (Olimpíada Brasileira de Informática) como motivação para os alunos competirem entre si e manteve duas abordagens distintas de acordo com o nível dos estudantes.

Os alunos do ensino Fundamental obtiveram aulas puramente teóricas e voltadas essencialmente para resolução de problemas, focando na primeira fase da OBI. Por outro lado, os alunos do ensino Médio participaram de aulas práticas utilizando a linguagem de programação C e o ambiente Scratch de programação descrito em (CRISTOVÃO, 2008). Como no trabalho de Garcia et al (2008) 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. A oficina teve carga horária de 16 horas sendo aplicada em 4 aulas, ou seja, cada aula tinha 4 horas de duração. Ela foi aplicada no campus-IV da UFPB no município de Rio Tinto-PB e não houve processo de avaliação dos estudantes. Na última aula, os autores aplicaram questionários onde os alunos participantes avaliaram a oficina.

Segundo os autores, os principais pontos negativos da oficina foram a carga horária para o desenvolvimento do conteúdo a ser ministrado, devido a sua complexidade e o alto

índice de evasão, onde se constatou que dos 25 alunos que foram matriculados na oficina, apenas 16 alunos a concluíram.

### **3. 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 Grandellet al (2006), Python é uma linguagem de alto nível interpretada, originalmente projetada por Guido Van Rossum para ser facilmente aprendida. Van Rossum (1999) sugere em seu trabalho que qualquer pessoa pode se tornar mestre em programação usando Python.

De acordo com Zelle (1999), Miller (2004), Grandellet al (2006), Luitz (2010) e Borges (2010), algumas das vantagens de Python em relação a outras linguagens são:

- Portabilidade: a linguagem é interpretada via bytecode pela máquina virtual Python, tornando o código portátil.
- Sintaxe clara e concisa: comparado a linguagens como C e Java, Python têm uma sintaxe mais intuitiva.
- Multiparadigma: a linguagem suporta programação modular e funcional, além da orientação a objetos.
- Tipagem dinâmica: não se faz necessário a declaração de tipos de dados para variáveis.
- Feedback imediato: Python possui um interpretador interativo que auxilia o programador, no qual é possível testar e modificar trechos de código, manipular, extrair e converter dados, entre outras possibilidades.
- Indentação obrigatória: Python força o programador a manter boas práticas de programação e indentar blocos de código dentro de estruturas de repetição e de controle.

Com base nos trabalhos acima descritos, e considerando os bons resultados obtidos nos estudos realizados por Zelle (1999), Miller (2004) e Grandellet al (2006) sobre a utilização da linguagem de programação Python em cursos de Introdução a Programação, pretendeu-se no mini-curso explorar os recursos dessa linguagem visando facilitar o primeiro contato com a programação.

### **4. 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 (APÊNDICE A) para ser utilizado como base para o planejamento das aulas.

Já na Preparação e divulgação do mini curso iniciou-se o planejamento das aulas e a escrita dos respectivos planos para cada aula do mini-curso, juntamente com a divulgação do mini-curso. O processo de divulgação ocorreu em duas ocasiões, em um primeiro momento foi feita em duas escolas públicas da cidade de Patos – PB que oferecem o ensino Médio, a Escola Estadual de Ensino Fundamental e Médio Monsenhor Manoel Vieira e o Colégio Estadual Dr. Dionísio da Costa.

A divulgação ocorreu brevemente nas salas de aula e através de exposições com

banner nos intervalos entre as aulas. Neste momento procurou-se apresentar a proposta do mini-curso de maneira clara e objetiva, esclarecendo que no curso de licenciatura em computação se estuda disciplinas como a que estava sendo oferecida e que nesta se aprende como os programas de computador são desenvolvidos e funcionam.

Após a divulgação foram realizadas pré-inscrições de alunos que se mostraram interessados pelo mini-curso e, estes foram convidados a participar de um seminário no intuito de esclarecer possíveis dúvidas que ficaram durante a divulgação, e ao final destehomologar as inscrições para eventual participação no mini-curso.

O segundo momento da divulgação aconteceu no laboratório de informática da universidade onde foi apresentado um seminário sobre o curso de licenciatura em computação, dando ênfase à sua estrutura curricular e a sua abrangência em reação a outros cursos da área de informática, 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, que aconteceria através da aplicação de mini-testes, exercícios e mini projetos de acordo com o aprofundamento dos seus conhecimentos sobre o assunto abordado no mini-curso. Como critério motivacional, foi esclarecido que os estudantes que obtivessem desempenho final superior ou igual a 7 ganhariam um certificado de participação. Esclarecido todas as dúvidas dos pré-inscritos sobre o mini-curso as inscrições dos interessados presentes foram homologadas.

Os critérios para participação foram o interesse por parte do estudante em aprender sobre algoritmos e linguagem de programação e a sua disponibilidade de tempo no horário em que as aulas fossem realizadas. Assim, 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. Isso foi feito para tentar evitar altos índices de evasão.

O mini-curso foi realizado entre os meses de setembro e novembro de 2011, as aulas tiveram carga horária de 2 horas aula contando com dois encontros semanais totalizando, foram homologadas 78 inscrições de alunos das escolas já citadas. Estes foram organizados em duas turmas (Turma A e Turma B) levando em consideração o melhor horário definido por cada aluno para se deslocar até a universidade. O curso foi ministrado no laboratório de informática do campus VII da UEPB no município de Patos e deu aos estudantes a oportunidade de conhecer o ambiente da instituição, bem como os cursos oferecidos no campus, particularmente o de licenciatura em computação. Por fim, na avaliação e resultados, os resultados analisados nesse trabalho são provenientes de dados obtidos ao longo da vigência do mini-curso.

## **5. PLANEJAMENTO E PROGRESSÃO DO MINI-CURSO**

O mini-curso foi dividido 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. Aqui, os alunos puderam aprender conceitos fundamentais sobre os algoritmos e refletir sobre as práticas cotidianas que se assemelham com a sua

definição, e sobre o histórico das linguagens de programação e sua relação com os algoritmos. No decorrer das aulas, o conceito de algoritmo foi formalizado, a partir daí foram apresentadas às principais formas de representação dos algoritmos, com foco principal nos pseudocódigos.

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. Um fragmento dessa variação pode ser visto na Figura 1. Durante esta unidade, a maior parte dos problemas propostos em sala de aula, além de avaliações, foram resolvidos pelos estudantes utilizando pseudocódigos.

Algoritmo em pseudocódigo	Script em Linguagem Python
1: x = leia('inf. um número:')	1: x = input('inf. um número:')
2: y = leia('inf. um número:')	2: y = input('inf. um número:')
3: se x == y:	3: if x == y:
4:     escreva x, 'e', y, 'são iguais'	4:     print x, 'e', y, 'são iguais'
5: senão:	5: else:
6:     se x < y:	6:     if x < y:
7:         escreva x, 'é menor que', y	7:         print x, 'é menor que', y
8:     senão:	8:     else:
9:         escreva x, 'é maior que', y	9:         print x, 'é maior que', y

Figura 1- Equivalência entre pseudocódigo e script em linguagem Python

A abordagem empregada no decorrer da unidade prática foi do tipo top-down, na qual os alunos puderam aplicar os conceitos aprendidos na unidade anterior e converter os pseudocódigos desenvolvidos em scripts na linguagem de programação Python. Além disso, novos conhecimentos e aplicações foram apresentados.

A segunda unidade (Unidade Prática) foi desenvolvida de forma mais dinâmica, pois os alunos já tinham conhecimento teórico acerca de algoritmos. Os conteúdos pertinentes a esta unidade foram estruturas de controle: simples, encadeadas e aninhadas; estruturas de repetição; funções da linguagem Python para manipulação de strings; modularização: funções; estruturas de dados e suas funções de manipulação: listas e dicionários.

Para o acompanhamento das atividades dos estudantes e disponibilização de materiais relacionados ao mini-curso, foram utilizados e-mail e um grupo de discussão, onde era possível também postar dúvidas referente a algum dos assuntos visto em sala de aula.

Em ambas as unidades as aulas foram ministradas no laboratório de informática do Campus VII da Universidade Estadual da Paraíba no município de Patos – PB. Cada aula contou com um professor e monitores para auxiliar os alunos durante a realização de atividades. Todos os estudantes foram avaliados individualmente de forma continuada em ambas as unidades, exercícios realizados em sala, ou extraclasse, além de mini testes e pequenos projetos na segunda unidade foram as ferramentas para avaliação.

As aulas expositivas foram auxiliadas por materiais didáticos como Data Show, quadro branco, apostilas e livros em meio digital. 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 concretos para os alunos e representassem situações do dia-a-dia. Todas as aulas contaram com exemplos e exercícios práticos realizados gradativamente com os alunos.

## 6. 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, como mostra a Figura 2.

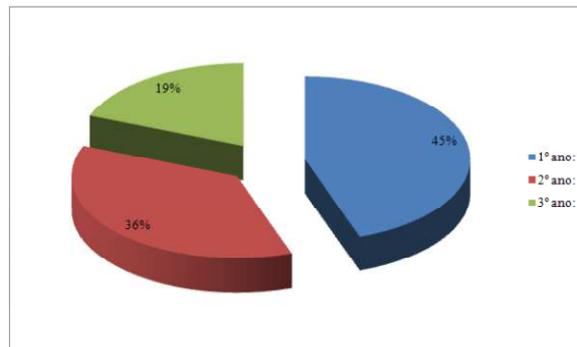


Figura 2 - Interesse dos alunos pré-inscritos

O desempenho individual de cada aluno foi analisado com base em sua média em cada unidade. As Figuras 3 e 4 representam respectivamente o desempenho dos alunos das turmas A e B. Através destas figuras é possível verificar claramente que houve um melhor desempenho pela maior parte dos estudantes das turmas A e B durante a segunda unidade do curso, sendo que apenas 14,28% destes obtiveram resultados abaixo da primeira unidade.

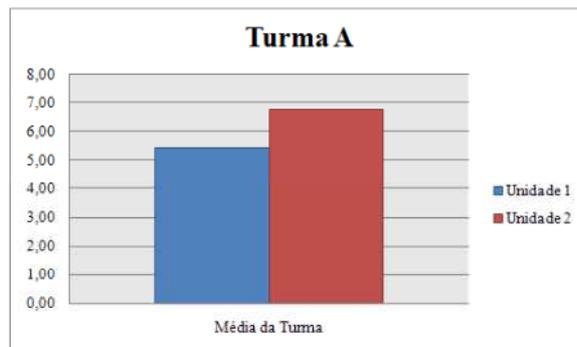


Figura 3 - Médias dos alunos da turma A em relação à unidade

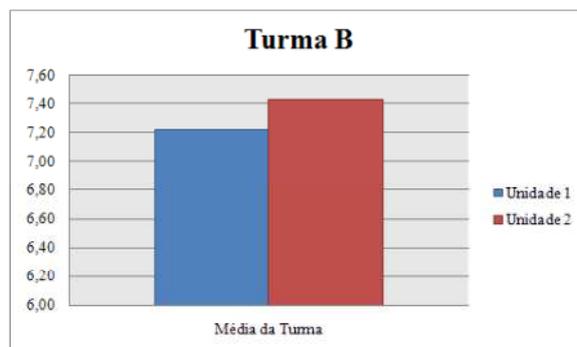


Figura 4 - Médias dos alunos da turma B em relação à unidade

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.

A avaliação final do aluno foi feita a partir da média aritmética das unidades (vide APÊNDICE A). Diante desses dados foi possível analisar que apenas 32% dos alunos da turma A concluíram o curso com desempenho igual superior a 7,0. Já na turma B, esse índice foi de 57%, como mostram as Figuras 5 e 6. Vale ressaltar que as turmas avaliadas diferem no número de alunos matriculados.

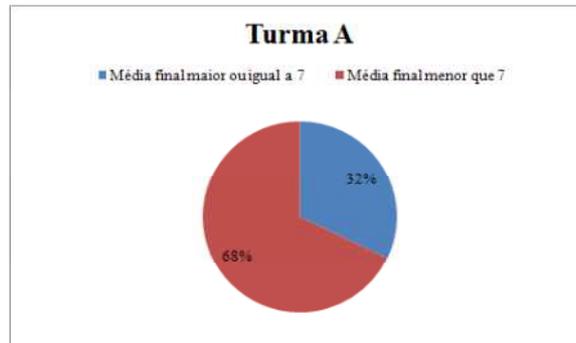


Figura 5- Avaliação final da turma A

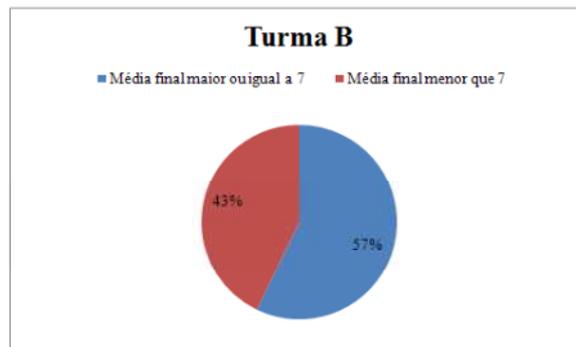


Figura 6- Avaliação final da turma B

Ao analisar os dados referentes à frequência dos alunos no mini-curso de uma forma geral foi possível verificar que seu índice de evasão foi de 46% como pode ser observado no gráfico da Figura 7.

Considerou-se evasão como sendo a desistência do curso, isto incluiu os alunos que se matricularam, mas nunca se apresentaram no curso (FAVERO; FRANCO, 2006) ou que faltaram a mais de 30% do curso.

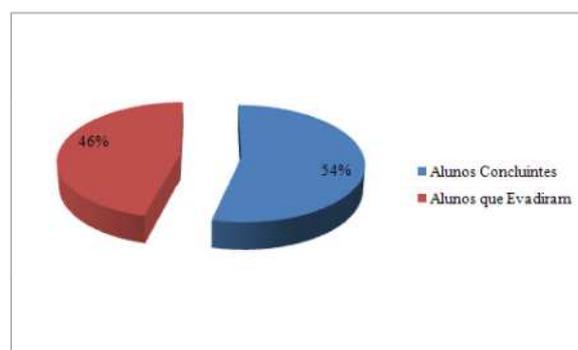


Figura 7- Taxa de evasão dos alunos

## 7. 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 em ambas as turmas.
- Linguagem de Programação Python: Outro ponto positivo foi a utilização da linguagem de programação Python, que foi bem aceita pelos alunos que não tiveram dificuldade em aprender a sua sintaxe.
- Abordagem Teórica inicial: Consistiu do início da primeira unidade (unidade teórica) e nesta etapa os alunos compreenderam a definição de algoritmo e praticaram desenvolvendo algoritmos em linguagem natural, isso foi importante para os alunos compreenderem os conceitos iniciais sobre programação de computadores.
- 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 algoritmos implementados na linguagem de programação Python. Durante as aulas práticas os alunos mostraram-se interessados logo nas primeiras aulas 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.
- 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.
- Certificados: 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: A primeira unidade do curso que tratou de algoritmos utilizando pseudocódigo foi vista pelos alunos como a mais difícil do curso. Durante as aulas teóricas utilizando pseudocódigo e o teste de mesa os alunos tiveram muita dificuldade em compreender e resolver a maioria dos problemas. Poucos alunos conseguiram realizar o teste com aptidão, e a grande maioria dos alunos estavam claramente desmotivados.
- Dificuldade de abstração dos alunos: Cabe ressaltar que 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 nas aulas iniciais, tais métodos provavelmente sejam mais efetivos que

adotar as práticas comuns de ensino.

## **8. 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 no futuro publicar este trabalho no Workshop sobre Educação em Computação, bem como aplicar novamente esse mini-curso e obter resultados ainda melhores e mais expressivos, tendo em vista as experiências obtidas nesta pesquisa.

## ABSTRACT

Considering the high dropout rates of courses related to technology and the growing demand for qualified professionals in this area it was decided to introduce notions of algorithms and programming language to the High school students through a university extension project as a way to encourage early entry of young as promising in this area. Thus, this paper proposes to report the experiences gained during the implementation of a short-course introduction to algorithms and programming language for High school students from public schools in the town of Patos - PB. The course included 40 hours lecture, divided into two units and participating students were subjected to a process of continuous evaluation of learning. The Python programming language was used in the course to introduce the basic concepts of structured programming in a simple way. At the end of the short-course was satisfactory meeting a demand of 78 High school students and getting a dropout rate of 46%, a rate lower than similar courses.

**KEYWORDS:** algorithms, programming language, teaching.

## REFERÊNCIAS

- AGUIARI, Vinicius. **Demanda por profissionais de TI cresceu 110%**. Disponível em: <<http://info.abril.com.br/noticias/carreira/demanda-por-profissionais-de-ti-cresceu-110-27122010-3.shl>> Acesso em: 29 jul. 2012.
- BAGGI, C. A. S. **Evasão e Avaliação Institucional: Uma Discussão Bibliográfica**. 2010. 81 f. Dissertação (Mestrado em Educação) – Faculdade de Educação, Pontifícia Universidade Católica de Campinas, Campinas, 2010.
- BARROS, Fábio. **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>> Acesso em: 29 jul. 2012.
- BORGES, Luiz Eduardo. **Python para Desenvolvedores**. 2010. Disponível em: <[http://ark4n.files.wordpress.com/2010/01/python\\_para\\_desenvolvedores\\_2ed.pdf](http://ark4n.files.wordpress.com/2010/01/python_para_desenvolvedores_2ed.pdf)> Acesso em: 5 jul. 2012.
- CASTRO, T.; FURKS, H. Sistematização da Aprendizagem de Programação em Grupo. In: SIMPÓSIO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO WORKSHOP DE EDUCAÇÃO EM INFORMÁTICA (WEI), 2011, Aracaju. **Anais...** Aracaju: [S.l.:s.n.], 2011. p. 1763-1773.
- CASTRO, T.; FURKS, H.; SPÓSITO. M. A. F.; CASTRO JR. A. N. Análise de um Estudo de Caso para Aprendizagem de Programação em Grupo. **IEEE-RITA**, v. 4, n 2, p. 155-160, mai. 2009.
- CRISTOVÃO, H. M. Aprendizagem de Algoritmos num Contexto Significativo e Motivador: Um Relato de Experiência. In: CONGRESSO DA SBC, 28., 2008, Belém do Pará. **Anais...** Belém do Pará: [S.l.:s.n.], 2008. p. 30-40.
- DANTAS, A.; LOPES, D.; COSTA, L. F. S.; SILVA, M. A. A. Aprendendo a Ensinar Programação Combinando Jogos e Python. In: SIMPÓSIO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO, 21., 2010, João Pessoa. **Anais...** João Pessoa: [S.l.:s.n.], 2010. p. 1-10.
- FAVERO, R. V. M.; FRANCO, S. R. K. Um estudo sobre a permanência e a evasão na Educação a Distância. **Revista Novas Tecnologias na Educação**, [S.l.:s.n.], v. 4, n. 2, p. 10, 2006.
- GARCIA, R. E.; CORREIA, R. C. M.; SHIMABUKURO, M. H. Ensino de Lógica de Programação e Estruturas de Dados para Alunos do Ensino Médio. In: CONGRESSO DA SBC, 28., 2008 **Anais...** Belém do Pará: [S.l.:s.n.], 2008. p. 245-249.
- GOMES, A.; HENRIQUES, J.; MENDES, A. J. Uma proposta para ajudar alunos com dificuldades na aprendizagem inicial de programação de computadores. **Educação, Formação & Tecnologias**, v. 1, Maio 2008.
- GRANDELL, L.; PELTOMAKI, M.; BACK, R. B.; SALAKOSKI, T. Why Complicate

Things? Introducing Programming in High School Using Python. In: AUSTRALASIAN COMPUTING EDUCATION CONFERENCE, 8., 2006, Hobart. **Proceedings...** Hobart: [S.l.:s.n.], 2006.p.71-80.

LUTZ, Mark. **Programming Python**. 4. ed.[S.l]: O'Reilly Media, 2010. 1632 p.

MACHADO, E. Z. A. ; VASCONCELOS, I. R. S. ; AMORIM, K. M. ; ANDRADE, A. M. S. ; BARRETO, L. P. ; ABDALLA, D. Uma Experiência em Escolas de Ensino Médio e Fundamental para a Descoberta de Jovens Talentos em Computação. In: WORKSHOP DE EDUCAÇÃO EM INFORMÁTICA (WEI), 2010, Belo Horizonte. **Anais...** Belo Horizonte: [S.l.:s.n.], 2010. p. 798-807.

MARQUES, L. M.; COSTA, L. F. S.; SILVA, M. A. A.; REBOUÇAS, A. D. D. S. Atraindo Alunos do Ensino Médio para a Computação: Uma Experiência Prática de Introdução a Programação utilizando Jogos e Python. In: SIMPÓSIO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO WORKSHOP DE EDUCAÇÃO EM INFORMÁTICA (WEI), 2011, Aracaju. **Anais...** Aracaju: [S.l.:s.n.], 2011. p. 1138-1147.

MILLER, J. A. **Promoting Computer Literacy through Programming Python**. 2004. Disponível em: <<http://www.python.org/files/miller-dissertation.pdf>> Acesso em: 5 jul. 2012.

NASCIMENTO, João KerginaldoFirmino.**Informática aplicada à educação**. Brasília, DF:[s.n.], 2009. 84 p.

PEREIRA JÚNIOR, J.; RAPKIEWICZ, C.E.; DELGADO, C.; XEXEO, J.A.M. Ensino de Algoritmos e Programação: Uma Experiência no Nível Médio. In: CONGRESSO DA SOCIEDADE BRASILEIRA DE COMPUTAÇÃO, 25., 2005, São Leopoldo. **Anais...** São Leopoldo: [S.l.:s.n.], 2005. p. 2351-2362.

ROCHA, H. V.**Representações Computacionais Auxiliares de Conceitos de Programação**. In: “Computadores e Conhecimento: Repensando a Educação”. Livro organizado por Valente, J. A. Editora Unicamp, 1993.

SOUZA, C. M. VisuAlg – Ferramenta de Apoio ao Ensino de Programação. **Revista TECCEN**, [S.l.], v. 2, n. 2, p. 10,setembro de 2009.

VALENTE, José Armando et al . **O computador na sociedade do conhecimento**. [S.L: s.n.], 1999.114 p.

VAN ROSSUM, Guido. **Computer Programming for Everybody**.1999. Disponível em: <<http://www.python.org/doc/essays/cp4e.html>>Acesso em: 5 jul. 2012.

**APÊNDICE A – Plano de curso**



Universidade Estadual da Paraíba  
 Centro de Ciências Exatas e Sociais Aplicadas  
 Campus VII – Governador Antônio Mariz  
 Curso: Licenciatura em Computação

## PLANO DE CURSO

Curso ofertado: Introdução a Algoritmos e Linguagem de Programação
Carga horária total: 40 horas
Professor: Rivanilson da Silva Rodrigues
Contato: <a href="mailto:rivanilson@gmail.com">rivanilson@gmail.com</a>

1. **Conteúdo:** Algoritmos, programação estruturada, tipos primitivos de dados, estruturas de controle e repetição, estruturas de dados, Subprogramas: funções, strings, arquivos.

### 2. Objetivos do Curso

**Geral:** promover o aprendizado sobre algoritmos e programação estruturada de computadores usando como ferramenta a linguagem de programação Python.

#### Específicos:

- Entender a importância dos algoritmos e da programação de computadores no contexto da informática e da Ciência da Computação;
- Aprender a elaborar soluções de problemas e mapeá-las para algoritmos e programas de computador;
- Aprender as construções básicas da linguagem Python;
- Projetar, implementar e testar soluções de problemas computacionais usando programas na linguagem Python;
- Aprender a usar subprogramação para aumentar a modularidade dos programas em Python.

### 3. Conteúdo Programático:

- Introdução a programação de computadores: programa, algoritmo, linguagem de programação, paradigmas de programação.
- Algoritmos: definição e exemplos no mundo real, tipos de algoritmos, algoritmos na programação de computadores, tipos de dados, expressões lógicas e aritméticas, entrada e saída, estruturas de seleção e repetição.
- Introdução à linguagem Python: histórico, estrutura de um programa, familiarização com o ambiente de desenvolvimento Python.
- Programação com a linguagem Python, estruturas de seleção.
- Strings.
- Subprogramação: funções.
- Estruturas de dados: listas e dicionários.

4. **Metodologia:** o conteúdo da disciplina será dado através de aulas expositivas e práticas no laboratório. As aulas serão ministradas no laboratório de informática da UEPB com auxílio de datashow e/ou quadro e pincel de acordo com o tema abordado.

**5. Avaliação:** a avaliação será constituída por 2 unidades, onde:

$$\text{Média\_Final} = \frac{\text{Und1} + \text{Und2}}{2}$$

As atividades de cada unidade compreendem: mini-testes desenvolvidos durante as aulas, exercícios extraclasse, bem como projeto desenvolvido no decorrer das atividades inerentes ao ensino de programação.

**6. Bibliografia:**

BORGES, Luiz Eduardo. **Python para Desenvolvedores**. 2010. Disponível em: <[http://ark4n.files.wordpress.com/2010/01/python\\_para\\_desenvolvedores\\_2ed.pdf](http://ark4n.files.wordpress.com/2010/01/python_para_desenvolvedores_2ed.pdf)> Acesso em: 5 jul. 2012.

DOWNEY, Allen. **Think Python: How to Think Like a Computer Scientist**. Needham:GreenTea Press, 2008.

FARRER, H. et. Al. **Programação Estruturada de Computadores – Algoritmos Estruturados**. Rio de Janeiro: LTC, 1999.

FERRARI, Fabrício; CECHINEL, Cristian. **Introdução a Algoritmos e Programação**. [S. l. : s. n.]