



Algoritmos e Linguagens de Programação

Luiz Carlos Melo Muniz



Visão geral da disciplina



Apresentação

- Professor Luiz Carlos Melo Muniz
- Doutorando em Ciência da Computação pela UFMA.
- Contato: lcmuniz@gmail.com



Disciplina

- Algoritmo e Linguagens de Programação
- 80 horas
- 2 avaliações
- 1 trabalho
- Exercícios
- Aulas expositivas e práticas de laboratório
- Site: lcmuniz.github.io/algoritmos



Conteúdo

- Visão geral da disciplina
- Análise e desenvolvimento de sistemas
- Algoritmos
- Linguagens de programação
- Instalação do Python
- Introdução ao Python
- Variáveis e primeiro programa



Bibliografia

- **Use a cabeça! Python**
Paul Barry
Alto Books
- **Introdução à computação e programação usando Python**
John V. Guttag
Infopress



Bibliografia

- **Entendendo algoritmos: um guia ilustrado para programadores e outros curiosos**
Aditya Y. Bhargava
Novatec
- **Estrutura de dados e técnicas de programação**
Francisco Bianchi
Elsevier




Análise e desenvolvimento de sistemas



Ciência da Computação

- Surgiu em meados do século XX.
- Engloba aspectos de várias áreas do conhecimento:
 - Matemática
 - Engenharia
 - Ciências Naturais
 - Arte



Análise e desenvolvimento de sistemas

- A análise e desenvolvimento de sistemas é uma subárea da Ciência da Computação.
- O tecnólogo com esta formação desenvolve, analisa, projeta, implementa e atualiza sistemas de informação.
- Tem noções de gerenciamento.
- Sua especialidade é a criação de sistemas informatizados: programação de computadores e desenvolvimento de softwares.



Análise e desenvolvimento de sistemas

- Implanta e desenvolve banco de dados.
- Conhece a estrutura física dos equipamentos e seus periféricos.
- Precisa se manter muito atualizado sobre aplicativos, ambientes operacionais e linguagens de programação.
- Precisa ter boa noção dos negócios da companhia para a qual trabalha.



Habilidade mais importante

- Capacidade de resolver problemas do mundo real de forma computacional.



Habilidades mais básicas

- Formular um problema do mundo real de forma computacional.
- Elaborar uma solução (um algoritmo) para este problema em termos computacionais.
- Escrever um programa em uma linguagem de programação que implemente este algoritmo.
- Testar o programa para verificar se ele resolve o problema corretamente.



Habilidades mais avançadas

- Gerenciar software de grande porte
- Construir software para lidar com grande quantidade de dados (big data, mineração de dados, aprendizado de máquina)
- Gerenciar equipes de desenvolvimento de software
- Comunicar-se com clientes e usuários.



Habilidades desenvolvidas nesta disciplina

- Habilidade de solucionar problemas
 - Formular questões
 - Pensar criativamente
 - Expressar a solução de forma clara e precisa
- Aprender a programar em uma linguagem de programação (Python)



Algoritmo

Algoritmo

- Uma lista de instruções passo-a-passo para resolver determinado problema
- Simplificadamente, o que fazemos é:
 - Analisar o problema
 - Criar um algoritmo que resolva o problema
 - Escrever um programa que implemente o algoritmo
 - Testar o programa para verificar se ele funciona

Problema: Torre de Hanói

- Três hastes e três discos de tamanhos diferentes que podem ser colocados em qualquer haste.
- Inicialmente, os discos estão empilhados em ordem crescente na mesma haste com o menor no topo
- Deseja-se mover todos os discos para outra haste mas só se pode movimentar um disco de cada vez e um disco maior não pode ser colocado sobre um disco menor

Pré-condições

- Nomeia-se as hastes como A, B e C
- Nomeia-se os discos como 1, 2 e 3
- Considere que os discos estão na haste A
- Os discos estão na haste A empilhados em ordem crescente com o disco 1 no topo
- Deseja-se mover os discos para a haste C

Algoritmo: Torre de Hanói

- 1) Mover o disco 1 para a haste C
- 2) Mover o disco 2 para a haste B
- 3) Mover o disco 1 para a haste B
- 4) Mover o disco 3 para a haste C
- 5) Mover o disco 1 para a haste A
- 6) Mover o disco 2 para a haste C
- 7) Mover o disco 1 para a haste C



Média de dois números

- 1) Obter o primeiro número
- 2) Obter o segundo número
- 3) Somar o primeiro número com o segundo número
- 4) Dividir o resultado do terceiro passo por 2
- 5) Mostrar o resultado do quarto passo

Programa em Python


```
primeiro = input('Digite o primeiro número: ')
```

```
segundo = input('Digite o segundo número: ')
```

```
soma = float(primeiro) + float(segundo)
```

```
media = soma / 2
```

```
• print('A média é', media)
```



Linguagens de programação

Linguagens de programação

- Linguagens formais, precisas e cujas instruções podem ser executadas por um computador
- Linguagens de baixo nível:
 - Linguagem de máquina
 - Linguagem Assembly
- Linguagens de alto nível:
 - Python, C, Java, Ruby, JavaScript, . . .

Tipos de linguagens

- Linguagens interpretadas



- Linguagens compiladas





Linguagens Interpretadas e Compiladas

- Exemplos: Java e Python
- Antes de iniciar a execução do programa, um compilador traduz o código-fonte para bytecode
- Ao iniciar a execução do programa, o interpretador lê os bytecodes um-a-um executando as instruções correspondentes



Por onde começar?

- Aprender a escrever código simples em uma linguagem de alto nível
- Executar pequenos programas
- Aprender novos conceitos incrementalmente
- Aumentar a sofisticação e o tamanho dos programas progressivamente



Instalação do Python

Windows

- Visite www.python.org/downloads/windows
- Clique em Latest Python 3 Release
- No fim da página, faça o download do arquivo Windows x86-64 executable installer
- Após o download, execute o arquivo para fazer a instalação do Python e do IDLE



Linux

- O Python já vem instalado na maioria das distribuições Linux
- Para instalar o IDLE, pode-se utilizar o gerenciador de pacotes da distribuição
- Pela linha de comando:
 - Ubuntu/Debian/Mint: `sudo apt-get install idle3`
 - Fedora/openSUSE/CentOS: `sudo yum install python3-tools python3-idle`