



Laboratório de Programação 1

Algoritmos e a Lógica de Programação

Mário Hozano
professor@hozano.com
Ciência da Computação
UFAL - Arapiraca

Relembrando a aula anterior...

- O que é um algoritmo?
- Como é o fluxo de um algoritmo ?
- Estruturas de Controle?
- Como um algoritmo pode ser representado?
- Representação com Diagrama de Blocos?
- Representação Textual?

Roteiro da aula

- Linguagem de Programação
- Linguagens de Baixo Nível
- Linguagens de Alto Nível
- Compiladores e Interpretadores
- Introdução à Linguagem Python

Representação de Algoritmos

- Um algoritmo pode ser descrito de duas formas:
 - 1. Forma Gráfica:** utilizando diagrama de blocos
 - 2. Forma Textual:** utilizando uma linguagem natural ou ainda uma linguagem de programação

Como expressar um algoritmo para utilizando linguagem de programação?

Linguagem de Programação

- **Resolver problemas** é uma das maiores habilidades de um Cientista da Computação
- Para resolver problemas os cientistas devem:
 1. Interpretar o problema
 2. Pensar criativamente nas soluções do problema
 3. Expressar a solução de forma clara e específica (muitas vezes utilizando diagrama de blocos)
 4. Escrever a solução utilizando uma **linguagem de programação**

Linguagem de Programação

O que é uma linguagem de programação?

Linguagem de Programação

O que é uma linguagem de programação?

O que é um programa?

Linguagem de Programação

- **Linguagem de Programação**

... é um método padronizado para expressar instruções para um computador.

... um conjunto de regras sintáticas e semânticas usadas para definir um programa.

- **Programa**

... conjunto de instruções que descrevem uma tarefa a ser realizada por um computador.

... é um software.

Linguagens de Baixo Nível

- Os primeiros programas eram implementados diretamente com circuitos lógicos
- Atualmente, a maioria dos computadores só entendem instruções escritas em linguagens de programação muito complexas conhecidas como **linguagens de baixo nível** de abstração ou **linguagens de máquina**
- Estas linguagens conseguem controlar/utilizar os circuitos lógicos, registradores e unidades de processamento do computador
- Um exemplo de linguagem de baixo nível é a linguagem **Assembly**

Linguagens de Baixo Nível

- Programa em Assembly para mostrar um '*' na tela

```
;exemplo2
.model small
.stack
.code
mov ah,2h ;move o valor 2h para o registrador ah
mov dl,2ah ;move o valor 2ah para o registrador dl
        ;(é o valor ASCII do caractere *)
int 21h  ;interrupção 21h
mov ah,4ch ;função 4ch, sai para o sistema operacional
int 21h  ;interrupção 21h
end      ;finaliza o programa
```

Linguagens de Baixo Nível

- Programas escritos em linguagens de baixo nível
 - Não são multiplataforma
 - Não são fáceis de escrever/entender
 - Não são fáceis de testar
 - São extremamente rápidos

Linguagens de Baixo Nível

- Programas escritos em linguagens de baixo nível
 - Não são multiplataforma
 - Não são fáceis de escrever/entender
 - Não são fáceis de testar
 - São extremamente rápidos

Existem linguagens de “alto” nível ?

Linguagens de Alto Nível

- Os programas escritos com linguagens de alto nível de abstração podem ser:
 - Multiplataforma
 - Fáceis de escrever/entender
 - Fáceis de testar
 - A maioria dos programas atuais são escritos com linguagens de alto-nível
 - Podem ser lentos devido as operações de tradução para linguagens de baixo nível

Linguagens de Alto Nível

- Programa em linguagem de alto nível que mostra um '*' na tela

```
print('*')
```

Linguagens de Alto Nível

- **Algumas linguagens de alto-nível**

Python, Java, C, C++, C#, Groovy, LUA,
Haskell, Basic, Pascal, Fortran,
PHP, Ruby, Eiffel, Flash, Javascript, Whitespaces,
Algol, Perl, Objective-C, Delphi, LISP

- **É possível comparar o nível de abstrações entre as linguagens**

- C é de mais baixo nível comparado à Java
- Java é de mais baixo nível comparado à Groovy
- Python é de mais alto nível comparado à C

Linguagens de Alto Nível

Como o computador entende programas escritos em linguagens de alto nível ?

Linguagens de Alto Nível

- Linguagens de alto nível devem prover mecanismos para traduzir seus programas para uma linguagem de baixo nível
- Estes mecanismos devem entender as instruções escritas na linguagem de alto-nível e preparar o programa para execução na linguagem de máquina
- Estes mecanismos podem ser de dois tipos:
 - Compiladores
 - Interpretadores

Linguagens de Alto Nível - Compiladores

- Os **Compiladores** são entidades responsáveis por traduzirem inteiramente um programa escrito na linguagem de alto-nível (código-fonte)
- Ele traduz o programa **inteiro** para um outro formato de mais baixo nível (código-objeto) que é entendido por um mecanismo de execução próprio
- C, C++, Delphi são exemplos de linguagens que utilizam compiladores



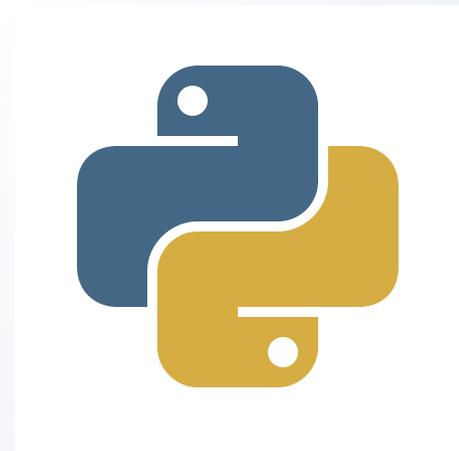
Linguagens de Alto Nível - Interpretadores

- Os **Interpretadores** são entidades responsáveis por traduzirem um programa escrito na linguagem de alto-nível (código-fonte) **linha por linha**
- Ele processa o programa um pouco de cada vez, alternadamente: ora lendo algumas linhas, ora realizando computações
- Python, JavaScript e Ruby são exemplos de linguagens que utilizam interpretadores



A linguagem Python

- Criada em 1991 por Guido Van Rossum
- Multiplataforma (Windows, Linux, Mac OS, Android...)
- Linguagem interpretada
- Vasta documentação
- Comunidade atuante
- Usada por empresas como Google, Yahoo, NASA, IBM, globo.com
- Site oficial: www.python.org
- Site oficial no Brasil: www.python.org.br



A linguagem Python - Interpretador

- Como usar o interpretador Python?
 1. Modo Interativo (*shell*): o interpretador executa instruções indicadas em linha de comando
 2. Modo de *scripts*: o interpretador executa todas as instruções contidas em um arquivo com extensão *.py* (código-fonte)

A linguagem Python – Interpretador Modo Interativo

- Utilizando o Interpretador Modo Interativo:
 1. Acesse o terminal do seu Sist. Operacional
 - No Ubuntu: *Aplicações* → *Acessórios* → *Terminal*)
 2. Digite o comando *python*. Irá aparecer algo como abaixo:

```
hozano@ufal:~$ python
Python 3.5.2 (r265:79063, Nov 23 2017, 13:09:56)
[GCC 5.4.0] on linux2
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

A linguagem Python – Interpretador Modo Interativo

- Uma vez iniciado, o interpretador em modo interativo, indica que está esperando instruções com o ***prompt de comando*** (>>>)
- Outras informações relevantes são exibidas

```
hozano@ufal:~$ python
Python 3.5.2 (r265:79063, Nov 23 2017, 13:09:56)
[GCC 5.4.0] on linux2
Type "help", "copyright", "credits" or "license" for more information.
```

```
>>>
```

Prompt de
comando

A linguagem Python – Interpretador Modo Interativo

- Uma vez iniciado, o interpretador em modo interativo, indica que está esperando instruções com o **prompt de comando** (`>>>`)
- Outras informações relevantes são exibidas

Versão do Python

```
hozano@ufal:~$ python
Python 3.5.2 (r265:79063, Nov 23 2017, 13:09:56)
[GCC 5.4.0] on linux2
Type "help", "copyright", "credits" or "license" for more information.
```

```
>>>
```

Prompt de comando

A linguagem Python – Interpretador Modo Interativo

- Uma vez iniciado, o interpretador em modo interativo, indica que está esperando instruções com o **prompt de comando** (`>>>`)
- Outras informações relevantes são exibidas

Versão do Python

```
hozano@ufal:~$ python
Python 3.5.2 (r265:79063, Nov 23 2017, 13:09:56)
[GCC 5.4.0] on linux2
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

Prompt de comando

Comandos para maiores informações

A linguagem Python – Interpretador Modo Interativo

- No prompt de comando, digite comandos Python:

```
>>> 1 + 1
2
>>> 10 - 3
7
>>> print("ola mundo")
ola mundo
>>> exit()
```

- Para realizar cada instrução, pressione <ENTER>. O interpretador irá realizar a instrução e depois mostrará novamente o prompt de comando
- Para sair do interpretador em modo interativo digite *exit()* ou pressione CTRL-D

A linguagem Python – Interpretador Modo *Script*

- Utilizando o Interpretador Modo *Script*:
 1. Abra um editor de textos (Ex.: gedit ou vi)
 2. Escreva comandos python e salve o arquivo com uma extensão .py

teste.py

```
print(3 + 7)
print(10 - 3)
print("Universidade Federal de Alagoas")
```

Em python, como em outras linguagens, as instruções são executadas de cima para baixo (sequência)

A linguagem Python – Interpretador Modo *Script*

- Utilizando o Interpretador Modo *Script* (*cont.*):
 3. Acesse o terminal do seu Sist. Operacional
 - No Ubuntu: *Aplicações* → *Acessórios* → *Terminal*)
 3. Acesse o diretório onde você salvou o arquivo e digite *python nome_do_arquivo.py*

```
hozano@ufal:~$ python teste.py
10
7
Universidade Federal de Alagoas
hozano@ufal:~$
```

A linguagem Python – Interpretador Modo *Script*

Parabéns, você fez seu primeiro programa em python :)

Exercício

- Acesse www.python.org e www.python.org.br e estude mais sobre a linguagem.
- Faça outros programas :)