

1 Programas Python

Este texto é uma pequena ajuda a quem estiver curioso para tentar traduzir os exemplos também para Python. O exemplo do outro texto (conceitos básicos) será desenvolvido, um pedaço de cada vez.

VOCÊ NÃO PRECISA SABER PYTHON PARA TERMINAR ESTA DISCIPLINA!

Podemos ignorar este texto se quiser!

(Mas se estiver curioso/curiosa para conhecer uma linguagem que é mais próxima de pseudocódigo, este texto pode ajudar)

1.1 Python no Netbeans

Há um plugin para Python que funciona com o Netbeans 6.7. Veja aqui as instruções de instalação:

- <http://wiki.netbeans.org/Python>
- <http://www.netbeans.org/features/python/index.html>

Você pode instalar o Netbeans 6.7 em sua casa com o plugin de Python.

1.2 No início de cada programa

Os programas que você fizer em Python devem ter a seguinte linha no início:

```
# coding: utf-8
```

Isso indica ao ambiente Python que o arquivo texto pode conter caracteres acentuados (e que eles estão codificados de uma maneira que chamamos “UTF-8”).

1.3 Variáveis

Não se declara tipo de variável em Python. O ambiente Python tentará adivinhar os tipos.

1.4 Entrada de dados

O comando `leia` que usamos em pseudocódigo é traduzido, em Python, da seguinte forma:

```
a = int(raw_input())
b = int(raw_input())
```

- `raw_input()` lê uma linha do teclado (o resultado é uma string);
- `int(..)` tenta interpretar a string como inteiro.

Isso significa “leia dois valores do teclado e armazene-os nas variáveis `a` e `b`”.

1.5 Decisão

Agora usamos a estrutura de decisão (“se” em pseudocódigo) em Java:

```
if (a == 0):
    print b
    exit()
if (b == 0):
    print a
    exit()
```

Se a for igual a zero, mostre b e termine o programa. Se b for zero, mostre a e termine o programa.
Note:

- O operador de igualdade é `==`, enquanto `=` significa atribuição de valor! Assim, `a == b` significa “a é igual a b?”, enquanto `a = b` significa “pegue o valor de b e armazene em a”;
- `print` mostra texto na tela (neste caso mostrará as variáveis b e a);
- `exit()` termina imediatamente o programa.

1.6 Repetição

```
while ((a % b) != 0):  
    c = a % b  
    a = b  
    b = c  
print b
```

Enquanto o resto da divisão de a por b (`a % b`) for diferente de zero, faça:

```
c <- resto de a por b  
a <- b  
b <- c
```

2 O exemplo completo

Aqui está o exemplo completo.

```
# coding: utf-8  
  
a = int(raw_input())  
b = int(raw_input())  
if (a == 0):  
    print b  
    exit()  
if (b == 0):  
    print a  
    exit()  
while ((a % b) != 0):  
    c = a % b  
    a = b  
    b = c  
print b
```