



universidade de aveiro
escola superior de tecnologia e
gestão de águeda

Introdução à Programação

Introdução ao Python

O que é um programa?

Um programa é uma sequência de instruções que especifica como realizar uma computação. A computação pode ser algo matemático, como resolver um sistema de equações ou encontrar as raízes de um polinômio, mas também pode ser uma computação simbólica, como pesquisar e substituir texto num documento ou algo gráfico, como processar uma imagem ou reproduzir um vídeo.

- ▶ **Input:** obtém dados do teclado, de um arquivo, da rede ou de algum outro dispositivo.
- ▶ **Output:** Exibe dados na tela, salva em um arquivo, envia pela rede, etc.
- ▶ **Math:** Executa operações matemáticas básicas como adição e multiplicação.
- ▶ **Conditional Execution:** Verifique determinadas condições e executa o código apropriado.
- ▶ **Repetition:** Executa alguma ação repetidamente, geralmente com alguma variação

Curiosidades sobre o Python

O Python é uma linguagem de programação de alto nível, criada por Guido van Rossum e lançada pela primeira vez em 1991

O nome “Python” foi inspirado no grupo de comédia britânico "Monty Python".

O Python foi desenvolvido com o objetivo de ser uma linguagem fácil de ler e escrever, com uma sintaxe clara e concisa.

A filosofia de design do Python caracteriza-se pela legibilidade do código e pela simplicidade, tornando-o acessível tanto a iniciantes como a programadores experientes.

Curiosidades sobre o Python

Um exemplo da simplicidade:

Problema: Determinação os números positivos ímpares menores que 100

C:

```
1 int main(void) {
2     int impares[50];
3     short i;
4
5     for (i = 0; i < 50; i++)
6         impares[i] = 2*i + 1;
7
8     return 0;
9 }
```

Python:

```
1 impares = [num for num in range(100) if num % 2 != 0]
```

Curiosidades sobre o Python

A comunidade Python é conhecida por ser ativa e colaborativa, contribuindo constantemente para o desenvolvimento e aprimoramento da linguagem.

Python possui uma vasta biblioteca padrão e muitos Módulos adicionais que facilitam o desenvolvimento em diversas áreas:

NumPy: Para computação numérica e manipulação de arrays.

Pandas: Para análise e manipulação de dados.

Matplotlib: Para visualização de dados.

Scikit-learn: Para machine learning e data science.

Django e Flask: Para desenvolvimento web

Curiosidades sobre o Python

Python é uma das linguagens mais populares do mundo, amplamente utilizada em diversas áreas como desenvolvimento web, ciência de dados, inteligência artificial e automação.

Alguns dos sites mais populares que usam Python:

- ▶ Instagram
- ▶ Google
- ▶ Spotify
- ▶ Netflix
- ▶ Uber
- ▶ Dropbox box
- ▶

Running Python

O interpretador Python é um programa que lê e executa código Python.

Dependendo do ambiente, pode-se iniciar o interpretador clicando num ícone ou escrevendo Python numa linha de comandos. Quando iniciar, deverá ver-se uma saída como esta:

```
Python 3.11.4 (tags/v3.11.4:d2340ef, Jun 7 2023, 05:45:37) [MSC v.1934 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
```

```
>>>
```

As três primeiras linhas contêm informações sobre o interpretador e o sistema operativo em que está a ser executado.

Verificando o número da versão, que neste exemplo é 3.11.4, começa com 3, o que indica que está a ser executado o Python 3. Se começar com 2 significa que está a ser executado o Python 2

Running Python

A última linha é uma prompt que indica que o interpretador está pronto para a inserção de código.

Se, se digitar uma linha de código e pressionar Enter, o interpretador exibirá o resultado:

```
>>> 1 + 1
```

```
2
```

Agora tudo está pronto para começar.

Ambiente de desenvolvimento

- ▶ Fazer download do Python

<https://www.python.org/downloads/>

- ▶ IDE (Integrated Development Environment)

 - ▶ Fazer download do Visual Studio Code

<https://code.visualstudio.com/download>

 - ▶ Configurar Python no Visual Studio Code

<https://code.visualstudio.com/docs/python/python-tutorial>

OU

 - ▶ Fazer download do PyCharm

<https://www.jetbrains.com/edu-products/download/#section=pycharm-edu>

Primeiro programa

Hello, World!

Tradicionalmente, o primeiro programa feito em todas as linguagens de programação.

Em Python tem o seguinte aspeto:

```
>>> print("Hello, World")
```

```
⚡ hello.py > ...
1   print("Hello World")
2
```

Os parêntesis indicam que o print é uma função.

Tipo de dados

Tipo de texto: **str**

Tipos numéricos: **int, float, complex**

Tipos de sequência: **list, tuple, range**

Tipo de mapeamento: **dict**

Tipo booleano: **bool**

Tipos binários: **bytes, bytearray, memoryview**

Nenhum Tipo: **NoneType**

Operadores aritméticos

► + Adição

```
>>> 40 + 2  
  
42
```

► / Divisão

```
>>> 84 / 2  
  
42.0
```

► // Divisão arredondada

```
>>> 101 // 2  
  
50
```

► - Subtração

```
>>> 43 - 1  
  
42
```

► % Resto da divisão

```
>>> 84 % 2  
  
0
```

► * Multiplicação

```
>>> 6 * 7  
  
42
```

► ** Potência

```
>>> 6**2 + 6  
  
42
```

Operadores de comparação

► > Maior

$i > j$

► \leq Menor ou igual

$i \leq j$

► \geq Maior ou Igual

$i \geq j$

► == Igualdade

$i == j$

► < Menor

$i < j$

► != Diferente

$i != j$

Debugging

Os erros de programação são chamados de **bugs** e o processo de os averiguar é chamado de **debugging**.

- ▶ **Syntax errors** - Indicam que há algo errado com a estrutura do programa.

Exemplo: Omitir os dois pontos no final de uma instrução `def` gera a mensagem ***SyntaxError: invalid syntax***.

- ▶ **Runtime errors** - Indicam que algo correu mal durante a execução do programa.

A maioria das mensagens incluem informações sobre onde ocorreu o erro e quais funções estavam a ser executadas.

Exemplo: Uma função recursiva infinita pode eventualmente causar o erro ***Maximum recursion depth exceeded***.

- ▶ **Semantic errors** - São problemas com um programa que é executado sem produzir mensagens de erro, mas sem produzir o resultado esperado.