

Think Python

How to think like a computer scientist

ALLEN B. DOWNEY

5 de agosto de 2021

Conteúdo

1	The Way of the Program	1
1.1	O que é um Programa?	1
1.2	Executando o Python	1
1.3	Operadores Aritiméticos	2
2	Variables, Expressions and Statements	3
3	Functions	5
4	Case Study: Interface Design	7
5	conditionals and Recursion	9
6	Fruitful Funtions	11
7	Iteration	13
8	Strings	15
9	Case Study: Word Play	17
10	Lists	19
11	Dictionaries	21
12	Tuples	23
13	Case Study: Data Structure Selection	25
14	Files	27
15	Classes and Objects	29
16	Classes and Functions	31
17	Classes and Methods	33

18 Inheritance	35
19 The Goodies	37
20 Debugging	39
21 Analysis of Algorithms	41

1

The Way of the Program

*“The single most important skill for a computer scientist is **problem solving**. That is the ability to formulate problems, think creatively about solutions, and express a solution clearly and accurately”*

– página 1

1.1 O que é um Programa?

Um **programa** é uma sequência de instruções que especifica como fazer um determinado computation. Um **computation**¹ é qualquer tipo de cálculo que inclua passos aritméticos (computação matemática) e não aritméticos (computação simbólica) e que segue uma determinada ordem.

Algumas características comuns em qualquer linguagem de programação:

- input → dados de teclado, arquivo, rede, etc.
- output → mostrar dados na tela, salvar em um arquivo, enviar na rede, etc.
- math → operações matemáticas
- conditional execution → só executar um pedaço do código após certas condições
- repetition → reexecutar um pedaço do código com alguma variação

1.2 Executando o Python

O **interpretador** do Python é um programa que lê e executa códigos escritos em Python. Tem duas maneiras de se executar códigos em Python a

¹<https://en.wikipedia.org/wiki/Computation>

primeira é no modo **interativo** onde podemos enviar as ordens direto pro interpretador e a segunda em modo **script** onde o interpretador analisa o código inteiro antes de começar o processamento.

1.3 Operadores Aritiméticos

```
1  # soma
2  40 + 2
3
4  # subtra o
5  40 - 2
6
7  # multiplica o
8  40 * 2
9
10 # divis o
11 40 / 2
```

Listing 1.1: Operadores Aritiméticos em Python

2

Variables, Expressions and Statements

3

Functions

4

Case Study: Interface Design

5

conditionals and Recursion

6

Fruitful Funtions

7

Iteration

8

Strings

9

Case Study: Word Play

10

Lists

11

Dictionaries

12

Tuples

13

Case Study: Data Structure Selection

14

Files

15

Classes and Objects

16

Classes and Functions

17

Classes and Methods

18

Inheritance

19

The Goodies

20

Debugging

21

Analysis of Algorithms