

# 11 Reunião 11: 06/OUT/2020

## 11.1 Reuniões passadas

- treino de raciocínio (sempre!) ←
- decomposição de um problema em problemas mais simples (sempre!):  
fatoração de inteiros



- indicadores de passagem: números primos
- repetições encaixadas: while ... dentro de while ... dentro de ...



Figure 1: Fonte: Alex teuscher via Getty Images

## 11.2 Hoje

- mais indicadores de passagem ainda
- repetições encaixadas: `while ... dentro de while ... dentro de ...`

## 11.3 Um exemplo interessante bolado por um aluno

Talvez ajude a entender os operadores `and` and `not`. Infelizmente, não lembro do nome do autor...

```
tenho_tempo = True
assisti_o_filme = False
if tenho_tempo and not assisti_o_filme:
    print("Vou ao cinema!")
else:
    print("Não vou ao cinema.")
```

Legal!

## 11.4 Exercício: menor e maior

Dados um número inteiro  $n$ ,  $n > 0$ , e uma sequência com  $n$  números inteiros, determinar o maior e o menor inteiro da sequência e a posição em que eles aparecem. Por exemplo, para a sequência

$n$	6						
	-2	7	0	-5	8	4	
	0	1	2	3	4	5	

o seu programa deve escrever o número

maior = 8 posição = 4

menor = -5 posição = 3

```
n = int(input("Digite n: "))
```

```
maior = menor = int(input( ))
```

```
imaior = imenor = 0 ←
```

```
i = 1
```

```
while i < n:
```

```
    num = int(input(...))
```

```
    if num > maior:
```

```
        maior = num
```

```
        imaior = i
```

```
→ elif num < menor:
```

```
    menor = num
```

```
    i += 1
```

```
print(f"{maior} {imaior}")
```

```
print(f"{menor} {imenor}")
```

Pensar

De usar

padrão

## 11.4.1 Rascunhos

```
n = int(input("Digite o tamanho da sequência: "))
maior = menor = int(input("Digite um número: "))
imenor = imaior = 0
i = 1
while i < n:
    num = int(input("Digite um número: "))
    if num > maior:
        maior = num
        imaior = i }
    elif num < menor:
        menor = num
        imenor = i
    i = i + 1

print("O maior número da sequência é", maior)
print("O menor número da sequência é", menor)
```

## 11.5 Exercício - tabuada

Escreva um programa que leia dois inteiros  $n$  e  $m$ . O programa deve imprimir as  $n$  tabuadas, dos números variando de 1 a  $n$ . Cada tabuada deve mostrar as multiplicações até  $m$ .

### 11.5.1 Exemplos

Digite o valor de  $n$ : 1

Digite o valor de  $m$ : 5

Tabuada do 1

1 x 1 = 1

1 x 2 = 2

1 x 3 = 3

1 x 4 = 4

1 x 5 = 5

Digite o valor de  $n$ : 2

Digite o valor de  $m$ : 6

Tabuada do 1

1 x 1 = 1

1 x 2 = 2

1 x 3 = 3

1 x 4 = 4

1 x 5 = 5

1 x 6 = 6

Tabuada do 2

2 x 1 = 2

2 x 2 = 4

2 x 3 = 6

2 x 4 = 8

2 x 5 = 10

2 x 6 = 12

```
n = int(input("Digite n: "))
m = int(input("Digite m:"))
```

```
i = 1
while i <= n:
```

```
    j = 1
    while j <= m:
        print(f"{i} x {j} = {i*j}")
        j += 1
    i += 1 # i = i + 1
```

Digite o valor de n: 3

Digite o valor de m: 4

Tabuada do 1

$$1 \times 1 = 1$$

$$1 \times 2 = 2$$

$$1 \times 3 = 3$$

$$1 \times 4 = 4$$

Tabuada do 2

$$2 \times 1 = 2$$

$$2 \times 2 = 4$$

$$2 \times 3 = 6$$

$$2 \times 4 = 8$$



Tabuada do 3

$$3 \times 1 = 3$$

$$3 \times 2 = 6$$

$$3 \times 3 = 9$$

$$3 \times 4 = 12$$

### 11.5.3 Solução

```
n = int(input("Digite o valor de n: "))
m = int(input("Digite o valor de m: "))

i = 1
while i <= n:
    print(f"Tabuada do {i}")
    j = 1
    while j <= m:
        print(f"{i}x {j} = {i*j}")
        j+=1
    print()
    i+=1
```

*possivel = False*

```
a = int(input("valor nota de a: "))
b = int(input("valor nota de b: "))
c = int(input("digite valor a ser trocado: "))
na = 0
while na*a <= c:
    nb = 0
    while nb*b + na*a <= c:
        if 3*a + nb*b == c:
            print(f"{na} notas de {a}, {nb} notas de {b}")
            nb += 1 → possivel = True
        nb += 1
    na += 1
```

*5ª feira*  
*Não perca!*

*if not possivel: # possivel == False*  
*print("Não há a troco possível")*



## 11.6 Exercício: troco

Digamos que um caixa automático tem somente notas de a e b reais para fornecer. Suponha que quando um(a) cliente deseja sacar c reais de sua conta, o caixa deve decidir se é possível fornecer esse valor.

Escreva um programa que resolva o problema do caixa. O seu programa deve ler os inteiros positivos a, b e c e imprimir todas as combinações possíveis de notas de a reais e notas de b reais, totalizando c reais, ou uma mensagem informando que não é possível sacar esse valor.

### 11.6.1 Exemplos

A seguir estão **três exemplos** de execução do programa.

Digite o valor de a: 5

Digite o valor de b: 3

Digite o valor do cheque: 11

1 nota(s) de 5 e 2 nota(s) de 3

Digite o valor de a: 4

Digite o valor de b: 6

Digite o valor do cheque: 24

0 nota(s) de 4 e 4 nota(s) de 6

3 nota(s) de 4 e 2 nota(s) de 6

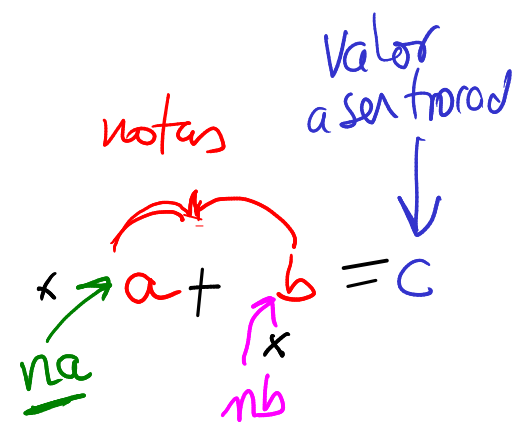
6 nota(s) de 4 e 0 nota(s) de 6

Digite o valor de a: 4

Digite o valor de b: 7

Digite o valor do cheque: 17

Não é possível fornecer esse valor



$$n_a * a + n_b * b = c$$

## 11.6.2 Rascunho

## 11.6.3 Solução

```
a = int(input("Digite o valor de a: "))
b = int(input("Digite o valor de b: "))
c = int(input("Digite o valor do saque: "))

possivel = False
na = 0
while na * a <= c:
    nb = 0
    while nb * b < c - na * a:
        nb += 1
    if nb * b + na * a == c:
        print(f"{na} nota(s) de {a} e {nb} nota(s) de {b}")
        possivel = True
    na += 1

if not possivel:
    print("Não é possível fornecer esse valor")
```