

FUNÇÕES. SOLUÇÕES AOS EXERCÍCIOS

MC102 - Algoritmos e
Programação de
Computadores

Santiago Valdés Ravelo
[https://ic.unicamp.br/~santiago/
ravelo@unicamp.br](https://ic.unicamp.br/~santiago/ravelo@unicamp.br)

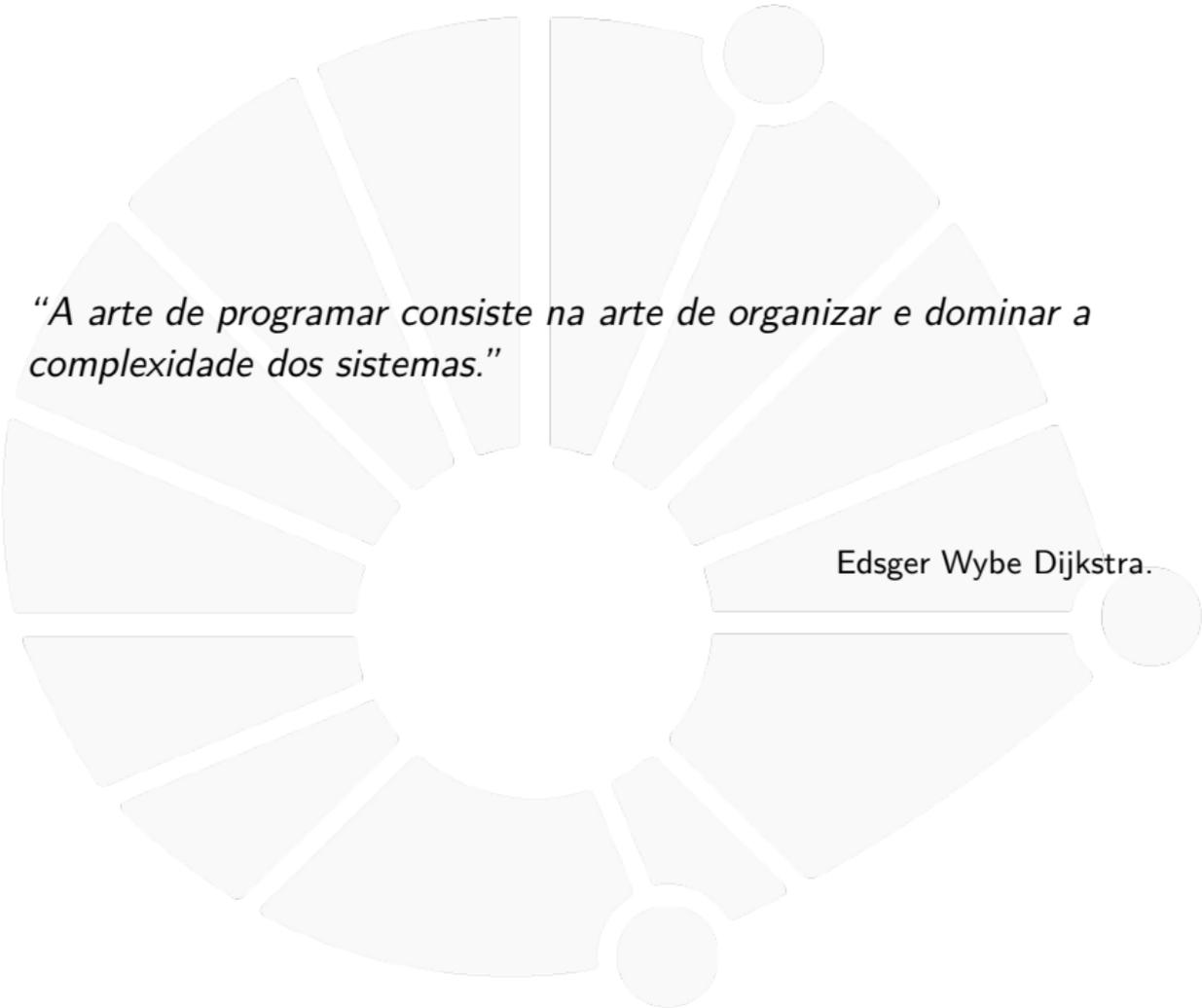
04/25

9



UNICAMP



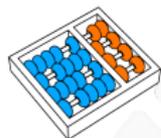


“A arte de programar consiste na arte de organizar e dominar a complexidade dos sistemas.”

Edsger Wybe Dijkstra.



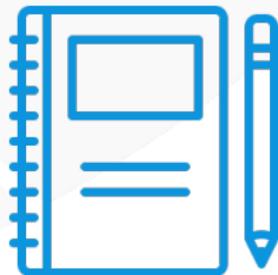
EXERCÍCIOS

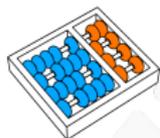


Funções



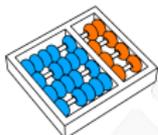
Soluções para os exercícios!





Exercícios

1. Faça uma função que acha o maior entre dois números.
2. Faça uma função que verifica se um número é primo.
3. Faça uma função que recebe uma lista e devolve uma nova lista invertida.
4. Faça uma função que devolve a lista dos divisores de um número.



Soluções

Maior:

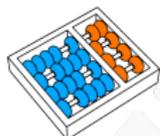
```
1 def maior(a, b):  
2     if a >= b:  
3         return a  
4     return b
```

Verifica primo:

```
1 def primo(p):  
2     k = 2  
3     while k * k <= p :  
4         if p % k == 0:  
5             return False  
6         k = k + 1  
7     return p > 1
```

Inverte lista:

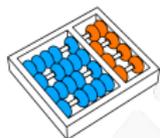
```
1 def inverte(lista):  
2     invertida = []  
3     for i in range(len(lista) - 1, -1, -1):  
4         invertida.append(lista[i])  
5     return invertida
```



Soluções. Continuação

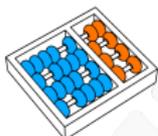
Divisores:

```
1 def divisores(n):
2     resultado = [1] # 1 é divisor de todos
3     k = 2
4     while k * k <= n :
5         if n % k == 0:
6             resultado.append(k)
7             if k * k < n: # Se não for a raiz, adicionar o outro divisor
8                 resultado.append(n // k)
9         k += 1
10    return resultado
```



Exercícios

1. Faça uma função que, dado n , decide se n é perfeito ou não.
2. Faça uma função que, dados dois números m e n , decide se são amigáveis.
3. Faça uma função que, dado um número n , devolve seu valor invertido.
4. Faça uma função que, dado um número inteiro n , desenhe um triângulo invertido na tela de base n , usando caracteres '*'.



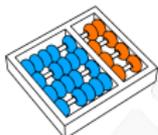
Soluções

Perfeito:

```
1 def soma_divisores(n):
2     if n == 1:
3         return 0
4     soma = 1 # 1 é divisor de todos
5     k = 2
6     while k * k <= n :
7         if n % k == 0:
8             soma += k
9             if k * k < n: # Se não for a raiz, adicionar o outro divisor
10                soma += n // k
11        k += 1
12    return soma
13
14 def perfeito(n):
15    return soma_divisores(n) == n
```

Amigáveis:

```
1 def amigaveis(a, b):
2     return soma_divisores(a) == b or soma_divisores(b) == a
```



Soluções. Continuação

Invertido:

```
1 def inverte(n):
2     resultado = 0
3     while n > 0:
4         resultado = 10 * resultado + n % 10
5         n //= 10
6     return resultado
```

Desenha triângulo:

```
1 def desenha_triângulo(n):
2     linha = n # tamanho da linha a ser desenhada
3     prefixo = 0 # indica quantos espaços devem estar no começo da linha
4     while linha > 0:
5         print((' ' * prefixo) + ('*' * linha))
6         linha -= 2
7         prefixo += 1
```

FUNÇÕES. SOLUÇÕES AOS EXERCÍCIOS

MC102 - Algoritmos e
Programação de
Computadores

Santiago Valdés Ravelo
[https://ic.unicamp.br/~santiago/
ravelo@unicamp.br](https://ic.unicamp.br/~santiago/ravelo@unicamp.br)

04/25

9



UNICAMP

