

# COMBINAÇÃO E USO DE ESTRUTURAS DE DADOS. 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)

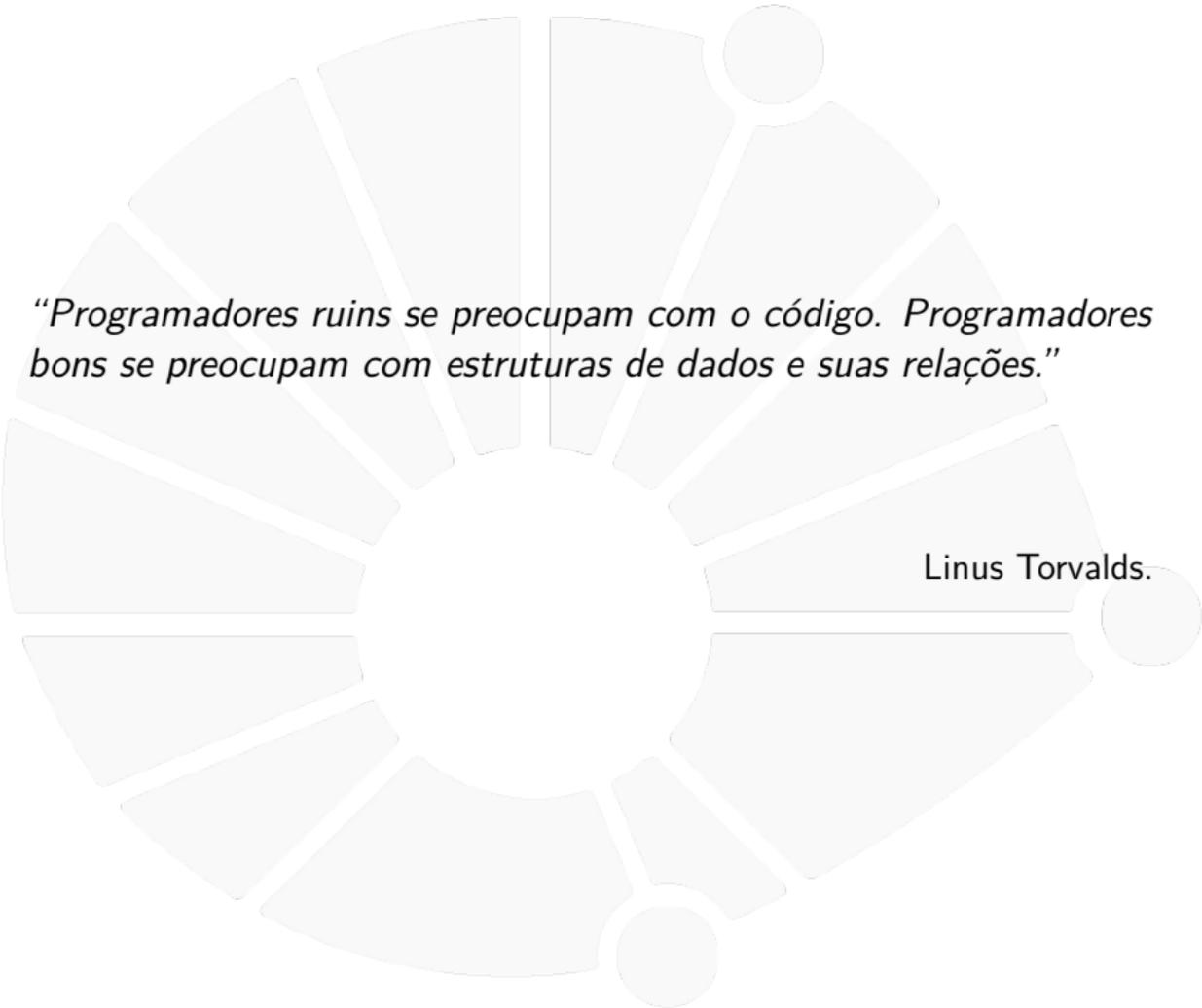
05/25

16



UNICAMP



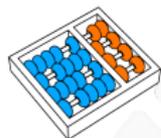


*“Programadores ruins se preocupam com o código. Programadores bons se preocupam com estruturas de dados e suas relações.”*

Linus Torvalds.



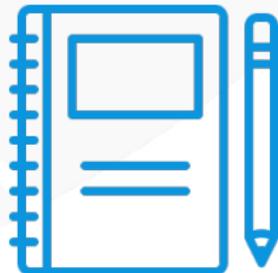
# EXERCÍCIOS

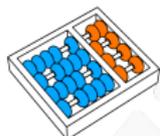


## Combinação e uso de estruturas de dados



**Soluções para os exercícios!**

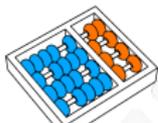




## Exercícios de compreensão

Use compreensão de listas e dicionários para:

1. Converter uma lista de temperaturas dadas em Fahrenheit para Celsius. Lembre-se que  $C = ((F - 32) * 5) / 9$ .
2. Fazer uma lista com os divisores de um **n** dado.
3. Pegar uma lista de strings que inclui palavras e números, e obter uma lista de inteiros apenas das strings que representam números não negativos.
  - ▶ Dica: use o método **isdecimal** de **str**.
  - ▶ Repita o exercício anterior para números inteiros quaisquer.
4. Fazer um dicionário que representa um histograma de uma lista (Dica: use o método **count** de **list**):
5. Criar uma matriz **n** por **m** com valor zero em cada entrada.
6. Criar uma matriz **n** por **m** com valores sequencias começando em **1** da esquerda para direita, de cima para baixo.



## Soluções

Converter uma lista de temperaturas dadas em Fahrenheit para Celsius:

```
1 celsius = [(f - 32) * 5 / 9 for f in fahrenheit]
```

Fazer uma lista com os divisores de um  $n$  dado:

```
1 divisores = [d for d in range(1, n + 1) if n % d == 0]
```

Obter uma lista de inteiros apenas das strings que representam números não negativos:

```
1 inteiros_positivos = [int(s) for s in lista if s.isdecimal()]
```

Para quaisquer inteiros:

```
1 inteiros = [int(s) for s in lista if s.isdecimal() or (len(s) > 1 and s[0] == '-' and
↳ s[1:].isdecimal())]
```

Histograma de uma lista:

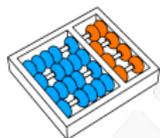
```
1 histograma = {e:lista.count(e) for e in lista}
```

Matriz  $n$  por  $m$  zerada:

```
1 matriz = [[0] * m for i in range(n)]
```

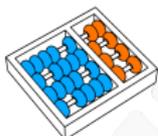
Matriz  $n$  por  $m$  sequencial:

```
1 matriz = [[1 + i * m + j for j in range(m)] for i in range(n)]
```



## Exercícios de estruturas com valores

1. Faça uma função que, dados um dicionário onde as chaves são pontos bidimensionais e os valores são números e um ponto  $(x, y)$ , encontra o valor do ponto mais próximo (distância Euclidiana) de  $(x, y)$  que é chave do dicionário.
2. Faça uma função que, dado um dicionário onde as chaves são nomes e os valores são listas de notas, calcula a média aritmética de cada nome, devolvendo um dicionário com os resultados.



## Soluções

Valor da chave mais próxima

```
1 def valor_proximo(pontos_valor, ponto):
2     proximo = None
3     distancia = 0
4     for p in pontos_valor:
5         dx = p[0] - ponto[0]
6         dy = p[1] - ponto[1]
7         d = dx * dx + dy * dy
8         if proximo == None or d < distancia:
9             proximo = p
10            distancia = d
11     return pontos_valor[proximo]
```

Média das notas:

```
1 def medias(nomes_notas):
2     return {nome: (sum(notas) / len(notas)) for nome, notas in nomes_notas.items()}
```

# COMBINAÇÃO E USO DE ESTRUTURAS DE DADOS. 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)

05/25

16



UNICAMP

