

# Countingsort

## Estrutura de Dados

prof. Frederico Santos de Oliveira

Universidade Federal de Mato Grosso  
Instituto de Engenharia

# Agenda

1 Exercício 1

2 Exercício 2

# Exercício 1

Considere o algoritmo de ordenação Countingsort apresentado em sala de aula. Utilizando alocação dinâmica, ordene um vetor composto de 1.000.000, 2.000.000 e 3.000.000 elementos. Para cada um, execute os seguintes testes:

- vetor composto por números aleatórios;
- vetor composto por números em ordem crescente;
- vetor composto por números em ordem decrescente.

Verifique quantas iterações são necessárias em cada um dos testes. Analise os resultados.

# Countingsort

## Pseudo-código

---

### Algoritmo 1: Countingsort

---

**Entrada:** Vetor  $V[0..n]$ , tamanho do vetor  $n$

**Saída:** Vetor  $V$  ordenado

1 **início**

```
2   // Considere o vetor auxiliar  $C[0..k]$ 
3   // onde  $k$  é o maior elemento presente no vetor.
4   // Inicializa o vetor auxiliar com zeros.
5   para ( $i \leftarrow 0$  até  $n - 1$ ) faça
6      $C[i] \leftarrow 0$ 
7   // Conta quantas vezes cada elemento aparece no vetor.
8   para ( $i \leftarrow 0$  até  $n - 1$ ) faça
9      $C[V[i]] \leftarrow C[V[i]] + 1$ 
10  // Insere no vetor original.
11   $j \leftarrow 0$ 
12  para ( $i \leftarrow 0$  até  $n - 1$ ) faça
13    enquanto ( $C[i] > 0$ ) faça
14       $V[j] \leftarrow i$ 
15       $C[i] \leftarrow C[i] - 1$ 
16       $j \leftarrow j + 1$ 
```

## Exercício 2

Adapte o Countingsort de forma que ele ordene uma sequência de letras do alfabeto.

# FIM

- FIM