



UNIVERSIDADE DE COIMBRA  
FACULDADE DE CIÊNCIAS E TECNOLOGIA

*Departamento de Engenharia  
Informática*

## Projeto #4 v3.2

### Algoritmos e Estruturas de Dados

(desenvolvimento ao longo de quatro sessões PL)

2021-2022 – 2º Semestre

**Submissão relatório (InfoEstudante) e Mooshak:**

|            |                  |            |                  |
|------------|------------------|------------|------------------|
| <b>4.1</b> | 1 de maio 23:59  | <b>4.2</b> | 8 de maio 23:59  |
| <b>4.3</b> | 15 de maio 23:59 | <b>4.4</b> | 22 de maio 23:59 |

**Anotações:** esta ficha foi preparada para ser resolvida maioritariamente no espaço das quatro sessões práticas. O código deve ser submetido no Mooshak e o relatório no infoEstudante.

É incentivado que os alunos discutam ideias e questões relativas ao trabalho. É entendido que, quer a reflexão final sobre os resultados obtidos, quer o código desenvolvido, são da autoria de cada estudante. A utilização, mesmo que parcial de código obtido de terceiros desvaloriza o processo de aprendizagem e é fortemente penalizada na avaliação do trabalho. A cópia de código desenvolvido por colegas ou obtido da net, sem que as fontes sejam referenciadas, constitui uma fraude por parte do aluno.

Para além de a fraude denotar uma grave falta de ética e constituir um comportamento não admissível num estudante do ensino superior e futuro profissional licenciado, esta marca definitivamente o processo de aprendizagem do infrator.

#### Objetivos:

Pretende-se que o aluno consolide conhecimentos adquiridos sobre

(1) complexidade temporal e espacial

(2) algoritmos de ordenamento.

O trabalho apresentar indicadores de desempenho dos algoritmos implementados.

#### Tarefas

As várias tarefas incluídas neste Projeto correspondem à resolução de um mesmo problema (descrito abaixo) usando diferentes alternativas de implementação:

##### **S1. solução I - "força bruta"**

solução sem recurso a algoritmos de ordenamento;

##### **S2. solução II - elementar**

solução baseada num algoritmo elementar de ordenamento;

##### **S3. solução III - eficiente**

solução baseada numa implementação otimizada do algoritmo *quicksort*

##### **S4. solução IV - muito eficiente**

solução com complexidade temporal linear

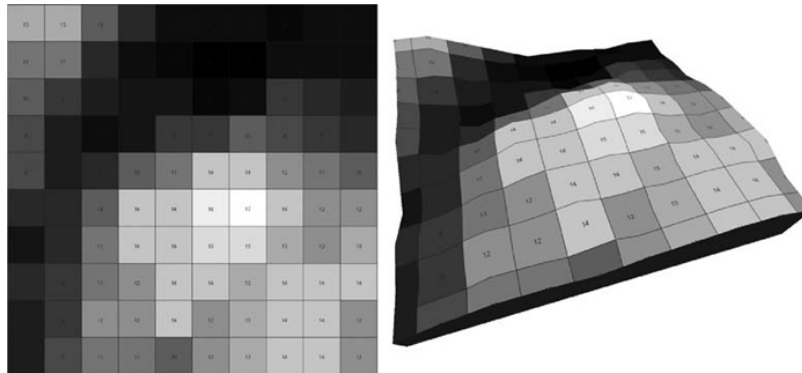
**R1 a R4. relatório semanal** de análise e consolidação de conhecimento (formulários em anexo).

#### Descrição do Problema – Pontuação de Um Região quanto à sua Ciclabilidade

Uma das preocupações de quem planeia a mobilidade em espaços urbanos é a promoção de meios suaves de transporte - a pé, bicicleta, trotinete.

O relevo do espaço urbano é um elemento fundamental que facilita ou dificulta a adoção de meios suaves de transporte. A informação do relevo pode ser guardada e representado em

formato *raster* (simplicadamente, temos uma grelha de pixéis na forma de uma matriz A x B em que cada célula guarda um valor numérico representando a elevação desse ponto em relação ao nível médio do mar (figura 1)).



<https://sk.sagepub.com/reference/geoinfoscience/n47.xml> (consultado em 30 Março 2022)

Neste projeto pretende-se desenvolver um programa que recebendo uma matriz de inteiros AxB, representando a elevação em metros, devolve uma série de indicadores sobre o relevo desse lugar. O objetivo é perceber a facilidade/dificuldade de implementar uma rede de transportes suaves. O valor da elevação é representado por um inteiro na gama 0..10000(m). Os parâmetros que se pretende calcular são:

1. amplitude (elevação máxima - mínima)
2. mediana (valor central da elevação)<sup>1</sup>
3. percentil a que pertence um conjunto N de valores de elevação<sup>2</sup>

### Tarefas A1 a A3

As tarefas A1 a A3 são semelhantes, no sentido em que têm os mesmos objetivos, apenas diferindo nos algoritmos que se utilizam e nas estruturas necessárias.

Em todas as tarefas o programa a desenvolver deve implementar as seguintes ações:

- a) “**RASTER N M**” – este comando RASTER precede dois inteiros com as dimensões A e B do raster seguido de A linhas cada uma compreendendo B inteiros. Todas as linhas têm um ‘\n’ a indicar o fim da linha. No fim de ler todas as linhas o programa deve devolver a mensagem “RASTER GUARDADO” seguido de mudança de linha.
- b) “**AMPLITUDE**” – este comando devolve um valor inteiro que representa a parte inteira da amplitude de elevação no RASTER seguido de mudança de linha.
- c) “**PERCENTIL N**” – este comando precede um valor inteiro que representa o número de valores de elevação de que se quer calcular o percentil. Devolve uma linha de inteiros que representam o

<sup>1</sup> Se o número de valores é ímpar, a mediana é o número localizado no meio da lista. Se o número é par, a mediana é a média dos dois valores do meio.

<sup>2</sup> Para calcular a que percentil pertence um dado valor fazer  
Percentil do Valor X = (num. valores inf. a X / num. total de valores) \* 100  
(para efeitos de validação mooshak arredondar ao inteiro inferior mais próximo)

percentil a que pertencem os valores de elevação pela ordem que foram dados como entrada. Se o valor de elevação for abaixo de todos os valores no RASTER deve devolver o valor zero

d) “**MEDIANA**” – este comando devolve o valor da mediana das elevações no RASTER<sup>3</sup> seguido de mudança de linha.

e) “**TCHAU**” – termina a sequência de comandos.

### Inputs / Outputs

*Exemplo de input 1:*

```
RASTER 3 2
10 20
20 50
10 10
AMPLITUDE
PERCENTIL 1
35
MEDIANA
TCHAU
```

*Exemplo output 1 (apresenta tudo em maiúsculas):*

```
RASTER GUARDADO
40
83
15
```

*Exemplo de input 2:*

```
RASTER 1 1
10
AMPLITUDE
PERCENTIL 2
35 5
MEDIANA
TCHAU
```

*Exemplo output 2 (apresenta tudo em maiúsculas):*

```
RASTER GUARDADO
0
100 0
10
```

---

<sup>3</sup> Para efeito de validação mooshak, se o valor da mediana for decimal, devolve o inteiro inferior mais próximo.

*Exemplo de input 3:*

```
RASTER 4 1
10
5
35
80
AMPLITUDE
PERCENTIL 4
13 35 70 90
MEDIANA
TCHAU
```

*Exemplo output 3 (apresenta tudo em maiúsculas):*

```
RASTER GUARDADO
75
50 50 75 100
22
```

*Exemplo de input 4:*

```
RASTER 5 5
10 60 40 30 5
10 15 60 40 10
10 10 60 20 5
10 10 60 10 0
15 15 20 60 0
PERCENTIL 7
5 10 55 35 15 100 77
MEDIANA
AMPLITUDE
TCHAU
```

*Exemplo output 4 (apresenta tudo em maiúsculas):*

```
RASTER GUARDADO
8 16 80 72 48 100 100
15
60
```

*Exemplo de input 5:*

```
RASTER 6 6
10 60 40 30 5 20
10 15 60 40 10 25
10 10 60 25 5 25
10 10 60 10 0 25
15 15 20 60 0 25
15 25 25 25 25 25
PERCENTIL 9
5 10 55 35 15 100 77 20 -30
MEDIANA
AMPLITUDE
TCHAU
```

*Exemplo output 5 (apresenta tudo em maiúsculas):*

```
RASTER GUARDADO
5 11 86 80 33 100 100 44 0
22
60
```

## Relatórios

Em anexo os formulários para os relatórios. **\*\*também num ficheiro WORD em separado\*\***

## Relatório Projeto 4.1 AED 2021/2022

Nome:

Nº Estudante:

PL (inscrição):

Login no Mooshak:

**Tabela (S1)**

**Gráfico (S1)**

Complexidade temporal da função PERCENTIL em ordem ao número  $M$  de valores para os quais é calculado o percentil e ao número  $N$  de valores de elevação na matriz raster. Considerar na complexidade temporal o tempo de ordenamento quando se aplicar.  
SUGESTÃO: usar  $M = N$ , ex.  $M = N = 100K \ 200K \dots 1000K$

A expressão  $O(f(n))$  para a complexidade temporal está de acordo com o esperado para as soluções S1 e S2? Justifique.

---

---

---

---

---

Qual a expressão  $O(f(n))$  para a complexidade espacial nas soluções S1 e S2? Justifique.

---

---

---

---

---

## Relatório Projeto 4.2 AED 2021/2022

Nome:

Nº Estudante:

PL (inscrição):

Login no Mooshak:

**Tabela (S3)**

**Gráfico (S3)**

Complexidade temporal da função PERCENTIL em ordem ao número  $M$  de valores para os quais é calculado o percentil e ao número  $N$  de valores de elevação na matriz raster. Considerar na complexidade temporal o tempo de ordenamento quando se aplicar.

SUGESTÃO: usar  $M = N$ , ex.  $M = N = 100K \ 200K \dots 1000K$

(1) Descreva sucintamente as otimizações feitas ao QuickSort. A expressão  $O(f(n))$  está de acordo com o esperado? Justifique.

---

---

---

---

---

---

Qual a expressão  $O(f(n))$  para a complexidade espacial na solução S3? Justifique.

---

---

---

---

---

## Relatório Projeto 4.3 AED 2021/2022

Nome:

Nº Estudante:

PL (inscrição):

Login no *Mooshak*:

**Tabela (S4)**

**Gráfico (S4)**

Complexidade temporal da função PERCENTIL em ordem ao número  $M$   
de valores para os quais é calculado o percentil e ao número  $N$  de valores de elevação na matriz raster.  
Considerar na complexidade temporal o tempo de ordenamento quando se aplicar.  
SUGESTÃO: usar  $M = N$ , ex.  $M = N = 100K \ 200K \ .. \ 1000K$

A expressão  $O(f(n))$  está de acordo com o esperado? Justifique.

---

---

---

---

---

Qual a expressão  $O(f(n))$  para a complexidade espacial na solução S4? Justifique.

---

---

---

---

---

## Relatório Projeto 4.4 AED 2021/2022

Nome:

Nº Estudante:

PL (inscrição):

*Login no Mooshak:*

**S1 - Tabela (complexidade temporal)**

**S2 - Tabela (complexidade temporal)**

**S3 - Tabela (complexidade temporal)**

**S4 - Tabela (complexidade temporal)**



**Gráfico de Complexidade Temporal S1 .. S4 (escala logarítmica)**

Explique sucintamente a implementação "força bruta" implementada em S1. E a solução implementada em S4.

Desenvolva os comentários que considere relevantes sobre a complexidade temporal vs espacial das várias implementações da solução.

This image shows a blank sheet of white paper with horizontal ruling lines. The lines are evenly spaced and run across the width of the page. There are no margins, text, or other markings on the paper.