

# Relatório 1º projecto ASA 2023/2024

Grupo: AL003

Aluno(s): Mafalda Szolnoky Ramos Pinto Dias (106494) e Francisco Lourenço Heleno (106970)

## Descrição do Problema e da Solução

O problema baseia-se em, dadas as dimensões de uma chapa de mármore e as dimensões de várias peças com os respetivos preços, calcular o valor máximo que pode ser obtido a partir da chapa inicial cortando-a em peças correspondentes às dimensões solicitadas pelos clientes.

Para resolver o problema, criamos uma matriz de dimensões  $(X + 1) \cdot (Y + 1)$ , sendo  $X \cdot Y$  a dimensão da chapa, onde inserimos o preço de cada peça (na respetiva dimensão) ou 0. Esta matriz é utilizada, de seguida, num algoritmo que implementámos, aplicando programação dinâmica de forma a resolver o problema ao dividi-lo em subproblemas mais pequenos.

## Análise Teórica

$$\text{calc}(x,y) = \begin{cases} 0 & , x = 0 \text{ ou } y = 0 \\ \max(\text{values}[x][y], \max(\max(\text{val1}, \text{calc}(l,j) + \text{calc}(l, c - j)), \max(\text{val2}, \text{calc}(i, c) + \text{calc}(l - i, c))) & , c. c. \\ \forall j, i \in \mathbb{Z} \text{ tal que } j \in [1, c) \text{ e } i \in [1, l) \end{cases}$$

- **Pseudo código:**

```
calc()
  for l = 1 to X
    for c = 1 to Y
      val1 = 0
      val2 = 0

      // Cortar horizontalmente
      for j = 1 to c
        val1 = max(val1, values[l][j] + values[l][c - j])

      // Cortar verticalmente
      for i = 1 to l
        val2 = max(val2, values[i][c] + values[l - i][c])

      // Associa o valor máximo entre os calculados e o atual
      values[x][y] = max(values[x][y], max(val1, val2))

  return values[X][Y]
```

Observações: X e Y correspondem às linha e coluna máxima da chapa, l e c simbolizam as linha e coluna atual, val1 e val2 são variáveis auxiliares, values é uma matriz de dimensões  $(X + 1) \cdot (Y + 1)$ . Realçamos ainda a alteração do nome da função e das variáveis aqui apresentados relativamente ao nosso código apenas por motivos estéticos e de legibilidade do relatório.

- Leitura dos dados de entrada: simples leitura do input com ciclo a depender linearmente do número de peças (n). Logo,  $\Theta(n)$ ;
- Processamento da instância para aplicação do algoritmo: criação de uma matriz e inserção em tempo constante do preço de cada uma das peças na mesma. Logo,  $O(1)$ ;
- Aplicação do algoritmo para cálculo do valor máximo que é possível obter a partir da chapa (de dimensão  $X \cdot Y$ ) e das peças fornecidas. Logo,  $O(X \cdot Y \cdot (X + Y))$ ;
- Apresentação dos dados.  $O(1)$ ;

Complexidade global da solução:  $O(X \cdot Y \cdot (X + Y)) \approx O(N^3)$ .

# Relatório 1º projecto ASA 2023/2024

Grupo: AL003

Aluno(s): Mafalda Szolnoky Ramos Pinto Dias (106494) e Francisco Lourenço Heleno (106970)

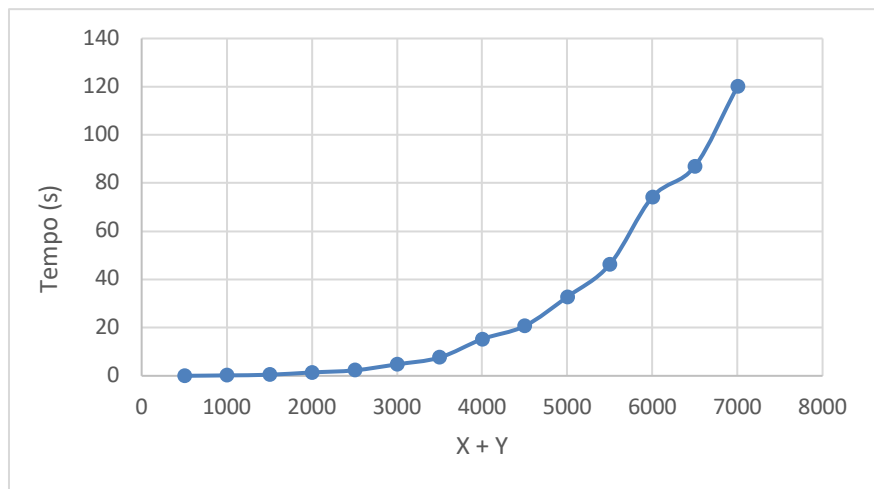
## Avaliação Experimental dos Resultados

Para a realização de experiências, utilizámos o gerador de instâncias, de modo a levar o nosso programa “ao limite” e perceber se estava de acordo com a análise teórica prevista.

Testámos assim o nosso código com 14 instâncias de tamanho incremental, mas mantendo constante o número de peças (550).

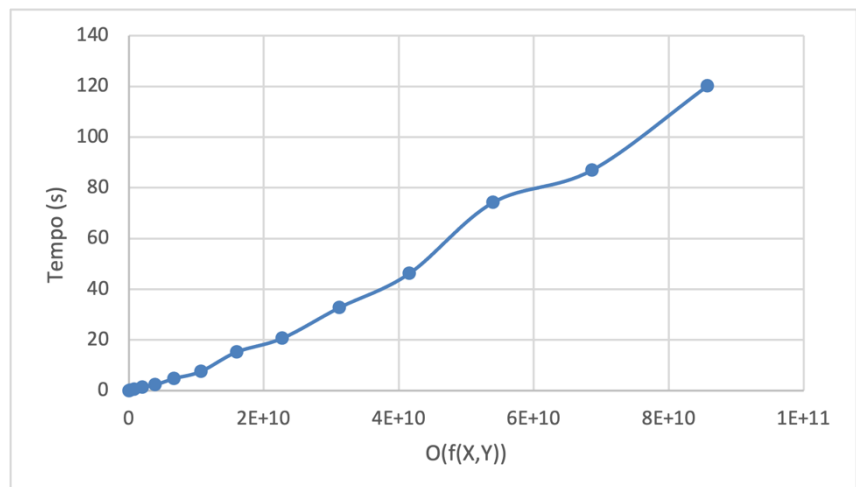
O gráfico seguinte representa o tempo de execução do nosso programa em função do tamanho das instâncias utilizadas.

X + Y	Tempo (s)
500	0,008
1000	0,184
1500	0,453
2000	1,412
2500	2,275
3000	4,78
3500	7,586
4000	15,238
4500	20,669
5000	32,769
5500	46,215
6000	74,216
6500	86,958
7000	120,116



Analisando o gráfico, é evidente que o tempo de execução não é linear nas dimensões da chapa. Desta forma, alterámos o eixo das abcissas para a variar com a quantidade prevista pela análise teórica, ou seja, colocámos o tempo em função de  $O(f(X,Y)) = O(X \cdot Y \cdot (X + Y))$ .

$X \cdot Y \cdot (X + Y)$	Tempo (s)
31250000	0,008
250000000	0,184
843750000	0,453
2000000000	1,412
3906250000	2,275
6750000000	4,78
10718750000	7,586
16000000000	15,238
22781250000	20,669
31250000000	32,769
41593750000	46,215
54000000000	74,216
68656250000	86,958
85750000000	120,116



Com esta mudança no eixo das abcissas, é notória a relação linear com os tempos no eixo das ordenadas, concluindo assim que a análise teórica de  $O(X \cdot Y \cdot (X + Y))$  estava correta relativamente à nossa implementação.