



UNIVERSIDADE DE COIMBRA
Faculdade de Ciências e Tecnologia
Departamento de Engenharia Informática

Estratégias Algorítmicas
Exame Recurso – 28 de junho de 2024

Nome: _____ Nº de estudante: _____

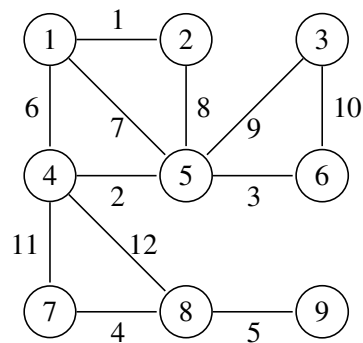
13 pontos no total, 2 horas, sem consulta.

1. Escreva o pseudo-código de um algoritmo recursivo que permita imprimir o Triângulo de Pascal (alinhado à esquerda) até um determinado valor de n , explorando a seguinte recorrência para $n > 0$ e $0 < k \leq n$: $P(n, k) = P(n-1, k-1) + P(n-1, k)$. Nota: Tem de considerar explicitamente os casos-base no seu algoritmo (2 pontos).

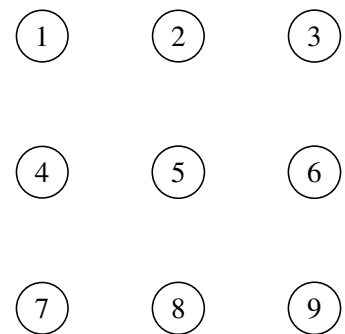
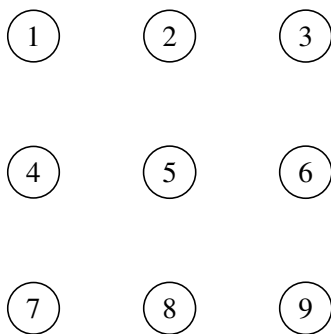
Exemplo para $n = 5$:

```
1
1 1
1 2 1
1 3 3 1
1 4 6 4 1
```

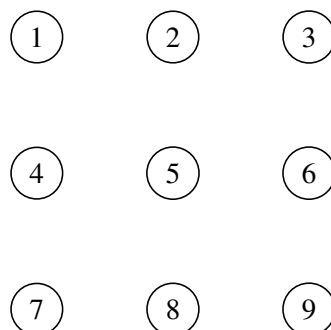
2. Considere o seguinte grafo.



- a) Desenhe a árvore geradora mínima (à esquerda) e o grafo da estrutura de dados *union-find*, sem o passo de compressão de caminho (à direita), recorrendo ao algoritmo de Kruskal. Quando necessário, ligue a raiz da árvore com menor altura à raiz da árvore com maior altura e, em caso de empate, escolha, como raiz, o vértice que apresentar a etiqueta com o menor valor. (2 pontos)

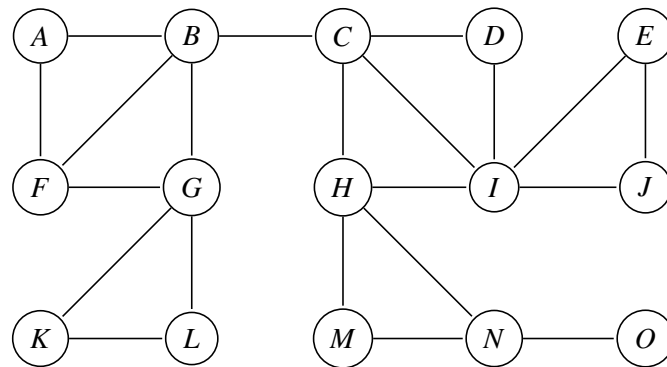


- b) Desenhe o caminho mais curto do nó 3 a cada nó do grafo usando o algoritmo de Dijkstra, indicando a distância mais curta em cada nó. Assuma que pode ir em ambos os sentidos em cada aresta. No fim da página, indique os nós visitados pela ordem que o algoritmo os seleciona (2 pontos).



Nós visitados:

3. Encontre os pontos de articulação e as pontes no grafo seguinte. Para justificação da sua resposta, reporte a árvore de procura em profundidade a partir do vértice *B*, escolhendo os vértices para a travessia de acordo com a ordem alfabética das etiquetas, e indique explicitamente os valores finais de *dfs* and *low* em cada vértice. Reporte igualmente os pontos de articulação e as pontes encontradas, ordenados pelo tempo em que foram encontrados durante a travessia em profundidade. (4 pontos)



4. Considere a seguinte recorrência para $i = 0, \dots, n$, $j = 0, \dots, n$ e $d_i \geq 0$:

$$D(i, j) = \begin{cases} \infty & \text{se } i < 1 \text{ or } j < 1 \\ d_i + \min\{D(i-1, j), D(i, j-1), D(i-1, j-1)\} & \text{caso contrário} \end{cases}$$

Tendo em conta que o valor que pretende obter é retornado por $D(n, n)$, apresente o pseudo-código de um algoritmo de programação dinâmica ascendente (*bottom-up*) que explore a recorrência acima para obter esse valor e discuta a sua complexidade computacional. (3 pontos)

