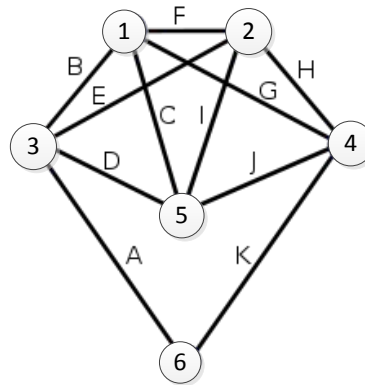


1. Para o seguinte grafo:



Apresente os vértices encontrados a partir do vértice 6, segundo uma visita em:

- a) Largura 6, 3, 4, 1, 2, 5
  - b) Profundidade 6, 3, 1, 2, 5, 4
2. Um grafo diz-se conexo (ou ligado) se para cada dois vértices X e Y há um caminho de X para Y ou de Y para X. Construa um algoritmo que verifique se um grafo é ou não conexo, e no caso de ser não conexo, apresente as ligações que não são possíveis entre vértices.
3. Pretende-se representar um mapa com várias cidades e vias de ligação entre as mesmas que poderão ser de dois tipos, autoestradas e estradas nacionais. Ambos os tipos de vias têm associado os quilómetros e as autoestradas o valor da portagem.
- a) Apresente a definição das classes não template necessárias para representar este mapa recorrendo à classe graphStl (não é necessário indicar os métodos das classes, só os seus atributos).
  - b) Implemente o método maiorLigacao, que indica a(s) cidade(s) cuja ligação direta apresenta maior distancia em kms.
  - c) Implemente o método maiorCaminhoDistancia que indica o maior caminho em distância entre um local origem e um local destino.
  - d) Implemente o método menorCaminhoCusto que indica o caminho com o menor custo de portagem entre um local origem e um local destino.

4. Considere um parque de diversões com várias atividades de entretenimento para as quais se sabe o seu código de identificação, tempo médio de espera e tempo de duração. Para além disto sabe-se também o tempo de percurso entre os pavilhões das várias atividades. Supondo que este parque é representado por um grafo dirigido da classe GraphStlPath:
- a) Apresente a definição das classes necessárias à modelação deste problema (não é necessário indicar os métodos das classes).
  - b) Apresente o **caminho mínimo** entre uma atividade inicial e uma atividade final, e indique o tempo total do caminho encontrado separado nas suas várias componentes: tempo total de duração das atividades, tempo total de espera e tempo total de percurso.