# Introdução

O principal objectivo deste projecto visava a concretização de um algoritmo que devolvesse o menor custo da construção de uma rede de aeroportos e estradas de forma a ligar um determinado conjunto de cidades. Devolve também o número total de aeroportos e estradas a construir. O algoritmo deve ainda detectar situações em que não é possível a construção de uma rede que ligue todas as cidades.

A solução abaixo descrita foi desenvolvida em C e demorou um total de XX horas a finalizar.

# Solução

Como referido, foi utilizada a linguagem C no desenvolvimento deste projecto.

[ESCREVER AQUI]

# Análise Teórica

Para determinar a complexidade do algoritmo, analisámos cada um dos seus procedimentos em separado:

* **Inicialização do grafo** dado um número V de vértices: Ѳ(V) (o vetor é percorrido depois de alocado para que todas as posições sejam inicializadas a NULL).
* **Adição de E arestas ao grafo inicializado**: Ѳ(E) (as arestas são adicionadas uma de cada vez à lista ligada do vértice origem).
* **Execução de uma DFS** no grafo construído: O(V + E) (se não for detetado um ciclo, o grafo inteiro é atravessado).
  + **Verificação de unicidade da ordem topológica**: O(V + E) (no pior caso, é percorrido todo o grafo).
  + **Impressão da lista ordenada**: Ѳ(V) (a lista ordenada tem tamanho V).
  + **Libertação de memória**: Ѳ(V + E) (necessário percorrer todo o grafo e toda a lista ordenada).

A descoberta de ciclos no grafo, como é realizada durante a execução da DFS, tem complexidade de tempo O(V) (pois, num DAG, V >> E).

Em suma, a complexidade total do algoritmo é O(V + E)

# Avaliação Experimental

[ESCREVER AQUI]

Referências

**Não existem fontes no documento atual.**