AirShuttle - Transfer em aeroportos

Turma 4 Grupo F

24/04/2019

Afonso Sá – up201604605@fe.up.pt

André Serralheiro – up201604566@fe.up.pt

Carlos Vieira – up201606868@fe.up.pt

Índice

Descrição do tema 3

Formalização do problema 4

* Dados de entrada
* Dados de saída
* Restrições
* Função Objetivo 5

Solução Implementada 6

Descrição do tema

A empresa AirShuttle presta serviços de transfer entre o aeroporto Francisco Sá Carneiro e hotéis, ou outros locais da região, por um preço bastante acessível.

Para o efeito dispõe de um certo número de carrinhas, uma ou mais, com capacidade para transportar um certo número de passageiros cada. Ao viajar para o porto, os passageiros têm a opção de realizar uma reserva do serviço pela Internet, indicando hora de chegada do voo, e destino para onde desejam ir.

Dado um conjunto de pedidos de serviços a prestar num determinado dia, e respetivos destinos, a empresa deve elaborar o seu plano de deslocações. A ideia é reunir o maior número possível de passageiros por deslocação, de forma a evitar que passageiros que chegaram mais cedo fiquem muito tempo à espera dos próximos.

Primeira Parte - Uma carrinha

Primeiramente no problema que vamos abordar a empresa tem uma só carrinha, como tal, de modo a maximizar a quantidade de deslocações estas devem ser breves.

De tal modo, a nossa carrinha vai realizar viagens de uma em uma hora, para cumprir este horário as viagens podem durar, no máximo, uma hora. Para se poder aproveitar ao máximo cada viagem é necessário ver as reservas e calcular o melhor caminho para se poder levar a maior quantidade de pessoas.

Segunda Parte – Frota de carrinhas

Na segunda parte a empresa possui uma frota de carrinhas, como tal as viagens efetuadas por cada carrinha podem ser maiores, uma vez que há mais carrinhas para preencher o horário.

Assim, uma carrinha chega ao aeroporto de 30 em 30 minutos e a duração máxima de uma viagem é de 4h. De tal modo, existem 8 carrinhas para tornar este horário possível.

Formalização do problema

Dados de entrada:

G (V, A) – Grafo Não Dirigido Pesado

V – Vértices, representam paragens entre o caminho realizado pela carrinha. Estas paragens são o aeroporto, hotéis e paragens predefinidas.

* P – Quantidade de passageiros nessa paragem

E – Arestas, são o caminho a ser percorrido entre vértices.

* T – Peso da aresta, que é medido em minutos (Tempo)

I – Vértice inicial, indica o local onde o caminho vai ser iniciado;

F – Vértice Final, indica o local onde o caminho vai terminar.

Dados de saída:

Os dados de saída são todas as paragens pelo qual a carrinha passa durante a viagem, a quantidade de tempo que esta demorou, bem como a quantidade de pessoas transportadas.

Caminho percorrido: {Vi}, i = 1, …, n ; n≠0

Tempo da viagem: (soma do tempo de cada arresta percorrida)

Fluxo de passageiros:

Restrições:

Em relação à viagem, na primeira parte, há a restrição de tempo de viagem, sendo este uma hora e para a segunda parte, quatro horas.

Quanto à quantidade de pessoas a serem transportadas de uma só vez, a restrição é de 10 pessoas, sem contar com o condutor, como tal, durante uma viagem não é possível transportar mais do que 10 pessoas ao mesmo tempo.

Função Objetivo:

De modo a otimizar a resolução do problema vamos necessitar que as viagens sejam as mais curtas possíveis, tanto em tempo como em distância, e com o maior número de passageiros transportados. Como tal a função objetivo deste problema requer a otimização de:

, de tal modo queremos maximizar o

, logo queremos o minimizar o

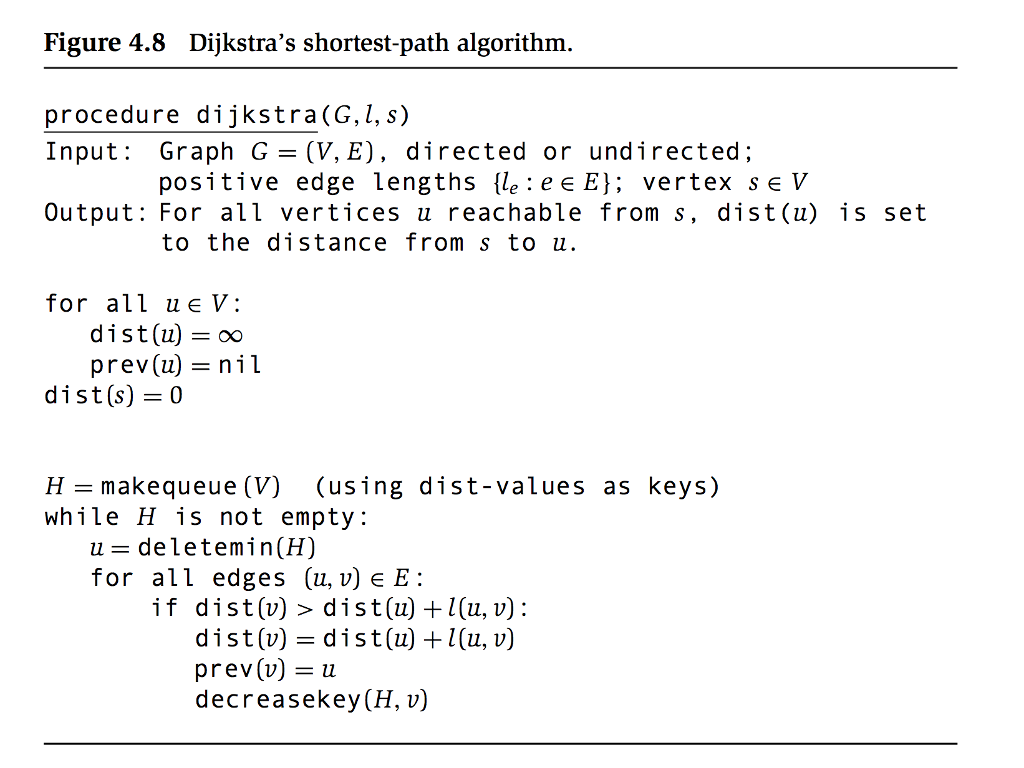
Solução Implementada

Algoritmos a utilizar:

Os algoritmos a implementar na resolução do problema serão:

* Dijkstra: Este algoritmo vai ser implementado de modo a obter o caminho mais curto, de tal modo obtendo assim o caminho mais rápido e maximizando a eficiência de cada transporte.

Como o grafo que é estudado é pesado é então necessário usar a variação gananciosa do algoritmo de Dijkstra, assim é possível obter uma maximização da eficiência em cada passo. O algoritmo possui então um tempo de execução de .

Exemplo em pseudo-código: