



Desenho de Algoritmos Trabalho_1

Logística Urbana para Entregas de Mercadorias

André Costa up201905916

Óscar Esteves up201906834

Pedro Nunes up201905396

Descrição

Cenário I

► Otimização do número de estafetas :

Implementação de um algoritmo que, perante um input de estafetas e encomendas, minimize o número de estafetas usados e maximize o número de encomendas entregues.

Os estafetas só podem fazer uma viagem por dia.

Os pedidos que, não possam ser transportados num dia, serão descartados retornando ao fornecedor, ou adiados para o dia seguinte (funcionalidade extra).

Solução

Cenário I

➤ Algoritmo Greedy :

Começamos por abordar o problema através de um Algoritmo Greedy.

Organizamos os estafetas por ordem decrescente da soma do volume máximo e do peso máximo.

&

Organizamos as encomendas pela ordem reversa, ou seja, ordem crescente da soma do volume e do peso.

Deste modo, iríamos ocupar primeiramente os estafetas de maior capacidade, com o maior número de encomendas.

Solução

Cenário I

Após a ordenação, passamos ao preenchimento dos estafetas.

Para isto, percorremos, para cada estafeta, todas as encomendas verificando então, se esta já estava ou não entregue e se poderia ir ou não no estafeta. Carregando/Entregando todas as encomendas, não seria necessário usar mais nenhum estafeta.

Assim, para os inputs bases (500 pedidos e 50 estafetas), obtemos o seguinte resultado:

```
No old Orders to be delivered!
```

```
There were used 23 couriers (46%) 0 for old orders and 23 for new orders  
A total of 450 (100%) orders were delivered!  
0 orders were passed to the next day!
```

Solução

Cenário I

Com esta abordagem, obtivemos uma complexidade temporal de $O(x*y)$ sendo:

x – nº de estafetas usados (n° total – nº de não usados)
 y – nº total de encomendas recebidas

E espacial de $O(n^\circ \text{ total de E} + n^\circ \text{ total de P})$

E – estafetas
P – pedidos

Perante os resultados das complexidades, começamos a pensar como melhorar estas.

Concluímos que, para baixar a complexidade temporal, podíamos simplesmente, à medida que enchiamos um estafeta e avançavamos para o próximo, começavamos a percorrer nos pedidos ainda não enviados.

Solução

Cenário I

Para isto, tínhamos de parar quando um estafeta estivesse cheio e começar, no próximo, onde este tinha parado.

Contudo, apesar de melhorarmos a complexidade temporal para $O(x + y)$, sendo:

x – nº de estafetas usados (nº total – nº de não usados)

y – nº total de encomendas enviadas

... pioramos o resultado obtido, pois as encomendas variam bastante no seu volume e peso. Umas podem ser pequenas mas pesadas e outras o inverso.

Com isto, podemos concluir que, ambas as abordagens são válidas dependendo apenas dos tamanhos dos inputs e os interesses da Empresa.

```
No old Orders to be delivered!
```

```
There were used 24 couriers (48%) 0 for old orders and 24 for new orders  
A total of 450 (100%) orders were delivered!  
0 orders were passed to the next day!
```

Descrição

Cenário II

➤ Otimização do lucro :

Implementação de um algoritmo que, perante um input de estafetas e encomendas, maximize o lucro da empresa para a entrega total ou do maior número de pedidos num dia.

(O lucro é calculado pela diferença entre o total de recompensas e o total dos gastos com os estafetas)

Os estafetas só podem fazer uma viagem por dia.

Os pedidos que, não possam ser transportados num dia, serão descartados retornando ao fornecedor, ou adiados para o dia seguinte (funcionalidade extra).

Solução

Cenário II

➤ Algoritmo Greedy :

Os estafetas são organizados por ordem ascendente do seu custo, enquanto as encomendas são organizadas por ordem descendente da recompensa associada.

Assim, os primeiros estafetas a ser utilizados vão ser os que têm menor custo, e as primeiras encomendas serão as que dão mais lucro.

Solução

Cenário II

► Algoritmo Bin Packing :

Ainda que o algoritmo Greedy já seja de, certa forma, eficiente a maximizar o lucro, não tem em conta o peso e o volume das encomendas, nem dos estafetas.

Deste modo, neste algoritmo, ordenamos tanto os estafetas como as encomendas, através de uma métrica à qual chamámos "*profit*", que corresponde ao custo/recompensa (dependendo se é estafeta ou encomenda) dividido pela soma do peso e do volume correspondentes.

No caso dos estafetas, são ordenados de forma ascendente, uma vez que menor é melhor (menor custo, menor valor de "*profit*").

Em relação às encomendas, estas são ordenadas de forma ascendente, uma vez que maior recompensa, maior valor de "*profit*".

Solução

Cenário II

Com estes 2 algoritmos, obtivemos os seguintes resultados, com as seguintes diferenças:

➡ Greedy:

```
[1] Few Couriers available
[2] All Couriers available
Choose one of the following: 1
[1] Few Orders available
[2] Some Orders available
[3] Many Orders available
Choose one of the following: 2

A total of 184 (40.8889%) orders were delivered!
The total profit was 150832 !
```

➡ Bin-Packing:

```
[1] Few Couriers available
[2] All Couriers available
Choose one of the following: 1
[1] Few Orders available
[2] Some Orders available
[3] Many Orders available
Choose one of the following: 2

A total of 218 (48.4444%) orders were delivered!
The total profit was 165952 !
Do you want to continue testing? [y/n] █
```

Solução

Cenário II

Com estas abordagens, obtivemos uma complexidade temporal de $O(x*y)$ sendo:

x – nº de estafetas usados (nº total – nº de não usados)

y – nº total de encomendas recebidas

E espacial de $O(\text{nº total de E} + \text{nº total de P})$

E – estafetas

P – pedidos

Descrição

Cenário III

► Entregas Expresso :

Implementação de um algoritmo que, perante um input de encomendas (encomendas expresso), minimize o tempo médio das entregas expresso a serem realizadas num dia, maximizando o número de pedidos entregues.

Para as entregas expresso, a empresa utiliza uma viatura com capacidade unitária, ou seja, tem de voltar sempre ao armazém após uma entrega.

Esta viatura, tem um horário comercial de 8h, das 9h-17h.

Solução

Cenário III

➡ Algoritmo Greedy :

Possuimos apenas uma viatura e o nosso objetivo principal é realizar o maior numero de tarefas possivel.

Para tal decidimos ordenar o vetor de entregas por ordem temporal e ir entregando sequencialmente as entregas de menor tempo.

Solução

Cenário III

Através desta abordagem obtivemos os seguintes resultados utilizando os inputs dados

```
Welcome to the 3rd Scenario, please choose how do you want to test the variables

[1] Few Orders to Delivery
[2] A normal amount of Orders to Delivery
[3] A lot of Orders to Delivery
Choose one of the following: 2

124 orders (27.5556%) were processed with a remaining time of 148 seconds
```

Solução

Cenário III

A complexidade temporal passa por ser $O(n)$ sendo:

n – tempo disponível

A complexidade espacial é $O(n)$ sendo:

n – número total de encomendas

Funcionalidades Extras

► Eficiência da operação da empresa :

A eficiência refere-se ao quociente entre o número de pedidos efetivamente entregues pelo número de pedidos recebidos num dia.

Esta funcionalidade foi implementada nos três cenários adaptada a cada problema, tal como podemos experienciar correndo os algoritmos e observando os outputs.

Funcionalidades Extras

► Transferência dos pedidos não entregues para o dia seguinte :

Esta funcionalidade tem como objetivo, priorizar os pedidos antigos. Para tal, teríamos de acrescentar uma "feature" ao algoritmo utilizado que, em vez de descartar os pedidos não entregues num dia, os passasse para o dia seguinte.

Estes por sua vez iriam ser priorizados em relação aos pedidos novos.

Esta funcionalidade foi implementada apenas no primeiro cenário, pois era no qual fazia mais sentido implementar.

Dificuldades encontradas

Considerámos implementar o Knapsack, porém existem vários estafetas, aumentando assim a complexidade da implementação e tornando o algoritmo bastante moroso quando existe um número significativo de estafetas e/ou encomendas.

Além disso, neste caso também existem 2 variáveis em análise, o que complicava a criação de uma tabela para aplicar ***Dynamic Programming***.

Autoavaliação do grupo

- André Costa: 33%
- Óscar Esteves: 33%
- Pedro Nunes: 33%