

INSTITUTO SUPERIOR TÉCNICO

Análise e Síntese de Algoritmos

2018/2019

2º Projecto

Data Limite de Entrega: 11 de Maio de 2019

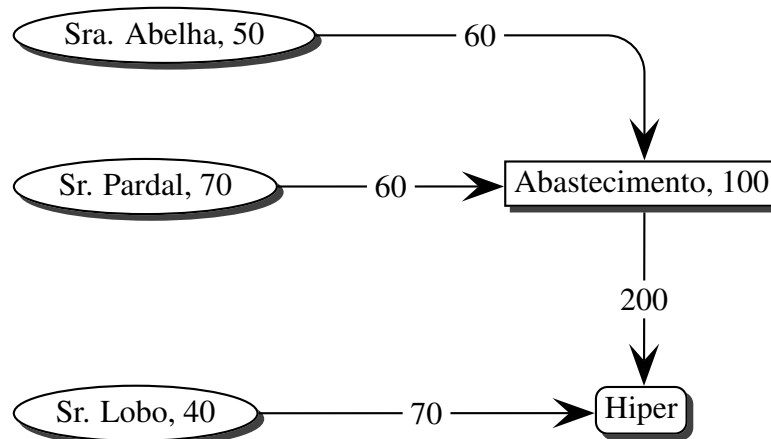
Descrição do Problema

O Sr. Caracol está num estado de euforia porque acabou de assinar um contrato para fazer a distribuição de uma grande cadeia de hipermercados, cujo nome não pode ser divulgado por razões de confidencialidade. A empresa que o Sr. Caracol gere tem ao seu dispor vários tipos de meio de transporte, carrinhas, camiões, aviões, comboios, barcos e até drones. Existem também várias estações de abastecimento e controlo.

O contrato estabelecido obriga o Sr. Caracol a providenciar o transporte de vários tipos de mercadoria dos fornecedores até ao hipermercado. Apenas existe um hipermercado, mas existem vários fornecedores. Por exemplo, a fábrica da Sra. Abelha produz mel, o talho do Sr. Lobo fornece carne e o pomar do Sr. Pardal fornece fruta.

Contudo o Sr. Caracol está preocupado com a capacidade da sua rede de transporte, dado que o contrato especifica que o Sr. Caracol terá de garantir o transporte de todo o material comprado aos fornecedores. O objectivo deste projecto é auxiliar o Sr. Caracol, desenvolvendo uma aplicação informática que determina a capacidade máxima que a sua rede é capaz de transportar. Para tal terá acesso a informação sobre a rede. Para uniformizar a informação, apenas será considerado o peso dos materiais a transportar. Os fornecedores irão indicar o peso do material que produzem. Cada meio de transporte tem um limite máximo que consegue movimentar. As estações de abastecimento são necessárias para abastecer os meios de transporte e também para fazer o controlo do transporte, garantindo que o material não é danificado ou extraviado. Contudo as próprias estações de abastecimento têm uma capacidade máxima, que limita a quantidade de material que podem processar.

O seguinte diagrama representa uma rede:



Neste caso a fábrica da Sr. Abelha produz 50 Kg de mel, o pomar do Sr. Pardal 70 Kg de fruta e o Sr. Lobo 40 Kg de carne. A rede consegue transportar 60 Kg entre a fábrica da Sr. Abelha e o posto de Abastecimento. Este posto pode processar apenas 100 Kg. Também é possível transportar 60 Kg entre o pomar do Sr. Pardal e o posto abastecimento. É possível transportar 200 Kg entre o posto de abastecimento e o Hiper. Por último é possível transportar 70 Kg entre o matador do Sr. Lobo e o Hiper. Neste caso a capacidade máxima da rede é de apenas 140 Kg, sendo que os fornecedores produzem 160 Kg. Portanto o Sr. Caracol precisa de aumentar a capacidade da sua rede, em particular precisa de aumentar a capacidade da estação de abastecimento. Por exemplo se conseguir aumentar esta estação para 110Kg isso irá aumentar a capacidade da rede para 150 Kg. Note que caso a produção do Sr. Lobo aumentasse para 70 Kg isso seria suportado pela rede de transporte, contudo essa limitação é relativa a um fornecedor e não à rede de transporte, logo a aplicação não deve reportar este tipo de informação.

Input

O ficheiro de entrada contém a informação da rede e dos fornecedores. Esta informação está organizada da seguinte forma:

- Uma linha com os valores $f > 0$, $e \geq 0$ e $t \geq 0$ separados por 1 espaço, que representam, respectivamente, o número de fornecedores, o número de estações de abastecimento e o número de ligações que existem na rede do Sr. Caracol;
- Uma linha com f inteiros, separados por 1 espaço, que representam a produção de cada fornecedor;
- Uma linha com e inteiros, separados por 1 espaço, que representam a capacidade de cada estação de abastecimento;

- Uma sequência de t linhas com 3 inteiros $o \geq 2$, $d \geq 1$ e $c \geq 1$ que representam a origem de uma ligação o , o destino de uma ligação d e a capacidade c da ligação em causa;

Para identificação de uma ligação é utilizada a seguinte numeração dos vértices:

- O Hiper é sempre o número 1;
- Os números de 2 a $f + 1$ representam fornecedores;
- Os restantes números representam estações de abastecimento e controlo;

Output

O output é definido da seguinte forma:

- Uma linha com a capacidade máxima da rede;
- Uma linha com os números das estações de abastecimento que devem ser aumentadas, separadas por um espaço e por ordem crescente. Caso não haja estações a ser aumentadas, deve ser apresentada uma linha apenas com o carácter de fim de linha;
- Uma sequência de linhas com os valores o e d das ligações que devem ser aumentadas. Esta sequência deve ser ordenada por ordem crescente, considerando primeiro o e em caso de empate d . Caso não haja ligações a ser aumentadas, nada deve ser apresentado;

Nota: Podem existir várias ligações que precisam de ser aumentadas. Devem ser reportadas apenas aquelas que definem a capacidade da rede e dentro destas apenas as mais perto do Hiper. No exemplo apresentado no enunciado apenas seria reportada a estação de abastecimento e mais nenhuma ligação.

Exemplos

A rede descrita no enunciado (ver figura) está representada no exemplo 1 seguinte.

input 1

```
3 1 4
50 70 40
100
2 5 60
3 5 60
4 1 70
5 1 200
```

output 1

140
5

input 2

1 3 8
10
5 2 3
2 4 2
5 1 3
1 4 2
3 1 5
2 5 3
4 1 2
3 3 7
2 3 5

output 2

10

1 3
1 4
1 5

input 3

1 3 8
8
5 2 3
2 4 2
5 1 3
1 4 2
3 1 5
2 5 3
4 1 2
3 3 7
2 3 5

output 3

8

input 4

1 3 6
8
5 2 1
2 4 2
5 1 3
3 1 5
2 5 3
4 1 2
2 3 5

output 4

8
5
1 3
1 4

input 5

3 3 10
3 15 5
6 8 2
5 1 7
6 1 10
7 1 4
2 7 1
2 6 2
3 7 1
3 6 2
3 5 1
4 6 4
4 5 5

output 5

```
12
7
5 3
6 2
6 3
```

Implementação

A implementação do projecto deverá ser feita preferencialmente usando as linguagens de programação C ou C++. Submissões em linguagem Java também são aceitáveis, devendo no entanto ter particular atenção a aspectos de implementação.

O tempo estimado para implementar este projecto é inferior a 20 horas.

Submissão do Projecto

A submissão do projecto deverá incluir um relatório resumido e um ficheiro com o código fonte da solução. Serão utilizadas as plataformas Mooshak para a submissão do código e Fénix para submissão do relatório. Informação sobre as linguagens de programação possíveis está disponível no website do sistema Mooshak. A linguagem de programação é identificada pela extensão do ficheiro. Por exemplo, um projecto escrito em C deverá ter a extensão .c. Após a compilação, o programa resultante deverá ler do 'standard input' e escrever para o 'standard output'. Informação sobre as opções e restrições de compilação podem ser obtidas através do botão 'help' do sistema Mooshak. O comando de compilação não deverá produzir output, caso contrário será considerado um erro de compilação.

O relatório deverá ser entregue no formato PDF com não mais de 4 páginas, fonte de 12pt, e 3cm de margem. O relatório deverá incluir uma introdução breve, a descrição da solução, a análise teórica e a avaliação experimental dos resultados. O relatório deverá incluir qualquer referência que tenha sido utilizada na realização do projecto. O texto do relatório deve ser cuidado. Serão aplicados descontos por erros gramaticais ou erros ortográficos. Relatórios que não sejam entregues em formato PDF terão nota 0. O código fonte deve ser submetido através do sistema Mooshak e o relatório (em formato PDF) deverá ser submetido através do Fénix. O código fonte será avaliado automaticamente pelo sistema Mooshak. Observe que apenas a última submissão será considerada para efeitos de avaliação. Todas as submissões anteriores serão ignoradas; tal inclui o código fonte e o relatório.

Os alunos são encorajados a submeter, tão cedo quanto possível, soluções preliminares para o sistema Mooshak e para o Fénix. Note que também é possível submeter várias vezes no Fénix e que não serão aceites relatórios fora de prazo e não haverá extensão de prazo.

O sistema Mooshak indica o tempo disponível para o projecto ser submetido. Os projectos têm que ser submetidos para o sistema Mooshak; não existe outra forma de submissão do

projecto. Os relatórios têm que ser submetidos no sistema Fénix; não existe outra forma de submissão dos relatórios.

Avaliação

O projecto deverá ser realizado em grupos de um ou dois alunos e será avaliado em duas fases. Na primeira fase, durante a submissão, cada implementação será executada num conjunto de testes, os quais representam 80% da nota final. Na segunda fase, o relatório será avaliado. A nota do relatório contribui com 20% da nota final.

Avaliação Automática

A primeira fase do projecto é avaliada automaticamente com um conjunto de testes, os quais são executados num computador com o sistema operativo **GNU/Linux**. É essencial que o código fonte compile sem erros e respeite os standards de entrada e saída indicados anteriormente. Os projectos que não respeitem os formatos especificados serão penalizados e poderão ter nota 0, caso falhem todos os testes. Um conjunto reduzido de testes utilizados pelo sistema Mooshak serão públicos. A maior parte dos testes **não** serão divulgados antes da submissão. No entanto, todos os testes serão disponibilizados após o deadline para submissão do projecto. Além de verificar a correcção do output produzido, o ambiente de avaliação restringe a memória e o tempo de execução disponíveis. A maior parte dos testes executa o comando `diff` da forma seguinte:

```
diff output result
```

O ficheiro `result` contém o output gerado pelo executável a partir do ficheiro `input`. O ficheiro `output` contém o output esperado. Um programa passa num teste e recebe o valor correspondente, quando o comando `diff` não reporta quaisquer diferenças (i.e., não produz qualquer output). Existem 16 testes. Assim, o sistema reporta um valor entre 0 e 16.

A nota obtida na classificação automática poderá sofrer eventuais cortes caso a análise do código demonstre recurso a soluções ajustadas a inputs concretos ou outputs aleatórios/constantes.

Detecção de Cópias

A avaliação dos projectos inclui um procedimento para detecção de cópias, através do sistema moss (<https://theory.stanford.edu/~aiken/moss/>). A submissão de um projecto implica um compromisso de que o trabalho foi realizado exclusivamente pelos alunos. A violação deste compromisso ou a tentativa de submeter código que não foi desenvolvido pelo grupo implica a reprovação na unidade curricular, para todos os alunos envolvidos (incluindo os alunos que disponibilizaram o código). Qualquer tentativa de fraude, directa ou indirecta,

será comunicada ao Conselho Pedagógico do IST, ao coordenador de curso, e será penalizada de acordo com as regras aprovadas pela Universidade e publicadas em “Diário da República”.