

INSTITUTO SUPERIOR TÉCNICO

Análise e Síntese de Algoritmos

2016/2017

2^o Projecto

Data Limite de Entrega: 02 de Maio de 2017

Descrição do Problema

O Sr. João Caracol, primeiro-ministro do Bananadistão, resolveu fazer um investimento em infra-estruturas no país por forma a integrar todas as cidades de média e grande dimensão numa rede de ligações aéreas e rodoviárias. O objectivo passa então por construir aeroportos e estradas novas que interliguem todas as cidades. Considerando que a construção de cada estrada e cada aeroporto tem um custo, o Sr. João Caracol pretende obter a solução que minimize o valor total da despesa.

Na construção da rede, considera-se que uma cidade que tenha um aeroporto consegue ligar-se a outras cidades com aeroportos. As cidades sem aeroporto, precisam de estradas para outras cidades por forma a ficarem ligadas. Considera-se que uma estrada liga duas cidades em ambos os sentidos.

O seu programa deverá decidir quais as estradas a construir e onde construir aeroportos de tal forma que todas as cidades fiquem ligadas em rede, e minimizando o custo total das obras.

Havendo soluções óptimas com o mesmo custo total, devem considerar a solução que minimiza o número de aeroportos a construir.

Input

O ficheiro de entrada contém a informação do número de cidades e do custo de construção de cada estrada ou aeroporto.

O input é definido da seguinte forma:

- Uma linha com o número N ($N \geq 2$) de cidades.
- Uma linha com o número A ($0 \leq A \leq N$) de potenciais aeroportos a construir.
- Uma sequência de A linhas, em que cada linha contém dois inteiros a e c (separados por um espaço em branco), onde c denota o custo de construir um aeroporto na cidade a .
- Uma linha com o número E de potenciais estradas a construir.
- Uma sequência de E linhas, em que cada linha contém três inteiros a , b e c (separados por um espaço em branco), onde c denota o custo de construir uma estrada entre as cidades a e b .

Assume-se que a identificação das cidades é um inteiro entre 1 e N .

Output

O seu programa deverá escrever no output a seguinte informação:

- Caso não seja possível construir a rede, o output deve consistir de uma única linha com a palavra `Insuficiente`.
- Caso seja possível construir a rede, o output deve consistir de duas linhas com a seguinte informação:
 - Uma linha com o custo total.
 - Uma linha com dois inteiros (separados por um espaço) indicando o número de aeroportos e o número de estradas a construir.

Todas as linhas do output terminam com o caracter de fim de linha.

Exemplos

input 1

```
4
3
1 1
2 5
3 1
4
1 2 1
1 3 6
2 4 2
3 4 3
```

output 1

5
2 2

input 2

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

output 2

Insuficiente

input 3

4
4
1 1
2 5
3 2
4 10
2
1 2 1
1 3 6

output 3

14
3 1

input 4

4
3
1 1

```
2 5
3 1
4
1 2 1
1 3 6
2 4 2
3 4 2
```

output 4

```
5
0 3
```

input 5

```
4
0
4
1 2 1
2 3 2
3 4 1
4 1 1
```

output 5

```
3
0 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. 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”.

3
3
1 3
2 3
3 3
3
1 2 5
1 3 5
2 3 5
9
3 0