

INSTITUTO SUPERIOR TÉCNICO  
Análise e Síntese de Algoritmos  
2021/2022

1º Projecto (actualizado a 2021.12.10)

Data enunciado: 03 de Dezembro de 2021  
Data Limite de Entrega: 04 de Janeiro de 2021 (17h00)

## Descrição do Problema

O projecto é composto pelos dois problemas que se descrevem em baixo.

**Problema 1** Dada uma sequência  $\vec{x} = \langle x_0, x_1, \dots, x_k \rangle$  de inteiros, pretende calcular-se o tamanho da maior subsequência estritamente crescente de  $\vec{x}$ , bem como o número de subsequências estritamente crescentes de tamanho máximo. Por exemplo, a sequência  $\vec{x} = \langle 1, 2, 6, 3, 7 \rangle$  tem duas subsequências estritamente crescentes de tamanho máximo igual a 4:  $\vec{s}_1 = \langle 1, 2, 6, 7 \rangle$  e  $\vec{s}_2 = \langle 1, 2, 3, 7 \rangle$ .

**Problema 2** Dadas duas sequências  $\vec{x} = \langle x_0, x_1, \dots, x_k \rangle$  e  $\vec{y} = \langle y_0, y_1, \dots, y_l \rangle$  de inteiros, pretende calcular-se apenas o tamanho da maior subsequência comum estritamente crescente entre  $\vec{x}$  e  $\vec{y}$ . Por exemplo, as sequências  $\vec{x} = \langle 1, 2, 6, 3, 7 \rangle$  e  $\vec{y} = \langle 1, 2, 4, 7, 3 \rangle$  têm duas subsequências comuns estritamente crescentes de tamanho máximo igual a 3:  $\vec{s}_1 = \langle 1, 2, 3 \rangle$  e  $\vec{s}_2 = \langle 1, 2, 7 \rangle$ .

## Input

O ficheiro de entrada contém a informação relativa ao problema a resolver e às sequências de inteiros correspondentes, e é definido da seguinte forma:

- uma linha contendo um inteiro que indica o problema a resolver: o inteiro  $n$  corresponde ao problema  $n$ ;

- $n$  linhas contendo cada uma delas uma sequência de inteiros separados por um único espaço em branco e não contendo qualquer outro carácter, a não ser o fim de linha.

## Output

Para o Problema 1, o programa deverá escrever no output dois inteiros  $t$  e  $c$  separados por um espaço, onde  $t$  corresponde ao tamanho da maior subsequência que respeita as restrições do problema e  $c$  corresponde ao número de subsequências de tamanho máximo.

Para o Problema 2, o programa apenas deverá escrever no output um inteiro  $t$  correspondente ao tamanho da maior subsequência que respeita as restrições do problema.

## Exemplo

### Input

```
1
1 2 6 3 7
```

### Output

```
4 2
```

### Input

```
2
1 2 6 3 7
1 2 4 7 3
```

### Output

```
3
```

## Implementação

A implementação do projecto deverá ser feita preferencialmente usando as linguagens de programação C ou C++. Submissões nas linguagens Java/Python também serão aceites, embora fortemente desaconselhadas. Alunos que o escolham fazer devem estar cientes de que submissões em Java/Python podem não passar todos os testes mesmo implementando o algoritmo correcto.

O tempo necessário para implementar este projecto é inferior a 15 horas.

### Parâmetros de compilação:

```
C++: g++ -std=c++11 -O3 -Wall file.cpp -lm
C: gcc -O3 -ansi -Wall file.c -lm
Javac: javac File.java
Java: java -Xss32m -Xmx256m -classpath . File
Python: python3 file.py
```

## Submissão do Projecto

A submissão do projecto deverá incluir um relatório resumido e um ficheiro com o código fonte da solução. 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.

**Relatório:** deverá ser submetido através do sistema Fénix no formato PDF com não mais de 2 páginas, fonte de 12pt, e 3cm de margem. O relatório deverá incluir uma 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. Atempadamente será divulgado um template do relatório.

**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 (<http://acp.tecnico.ulisboa.pt/~mooshak/>). Os alunos são encorajados a submeter, tão cedo quanto possível, soluções preliminares para o sistema Mooshak e para o Fénix. Note 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.

## 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 85% da nota final. Na segunda fase, o relatório será avaliado. A nota do relatório contribui com 15% 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. Os 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). O sistema reporta um valor entre 0 e 170.

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. 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 or 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”.