

1º Teste Prático – Enunciado B

2018.11.08

Prova com Consulta

Duração: 120 minutos

Nome Completo: _____

N.º de Estudante: _____ **Regime:** [] Diurno [] Pós-laboral

IMPORTANTE

É expressamente proibido o recurso à Internet durante a prova. Qualquer utilização não autorizada da Internet leva à anulação da prova e ao reportar da situação às autoridades competentes. O mesmo sucede com outros tipos de tentativa de fraude.

- **Antes de iniciar a prova:**

- Execute os seguintes comandos:

```
cd; mkdir -p ~/Prova01/R_NUMERO/
```

(em que **R** deve ser substituído pela letra **D** se for do regime diurno e **N** se for aluno do regime pós-laboral e **NUMERO** deve ser substituído pelo seu número ESTG);

- Para garantir que o seu diretório de trabalho seja o correto, faça:

```
cd ~/Prova01/R_NUMERO/
```

- **Após ter terminado a prova:**

- Deverá proceder à criação de um arquivo TAR, fazendo uso do seguinte comando:

```
cd ~/Prova01/R_NUMERO/; tar cvf Prova01_YYYYMMDD_R_NUMERO.tar *
```

(em que YYYYMMDD corresponde à data corrente (e.g., 20181031) e R_NUMERO obedece ao formato acima indicado);

- Verifique que o arquivo “.tar” que criou não está vazio, através da execução de:

```
tar tvf Prova01_YYYYMMDD_R_NUMERO.tar
```

- Entregue o arquivo “.tar” através da plataforma moodle, no espaço reservado para o efeito. Em caso de dúvidas, pergunte ao professor;
- Informe o professor para este validar a receção dos seus ficheiros.

Pergunta 1 [10 valores]

(Escreva as suas respostas a esta pergunta no diretório ”~/Prova01/R_NUMERO/Pergunta1”. Deve indicar o seu nome completo e número de estudante IPEiria no ficheiro **README.txt** a ser criado no diretório)

NOTA 1: não é permitida a chamada a comandos externos através da função **system** ou de outra com funcionalidade similar.

NOTA 2: a solução deve ser implementada com recurso aos ficheiros existentes no arquivo **EmptyProject-Templatev3.05.zip**.

NOTA 3: código entregue que **não compile** leva à atribuição da classificação de **0 (zero) valores** à resposta.

Recorrendo à linguagem C, elabore a aplicação **abc-finder**, que deve permitir contar o número de vezes que cada letra ocorre num ficheiro de texto. A contagem deve ser *case-insensitive*, não distinguindo portanto entre maiúsculas e minúsculas.

A aplicação deve suportar, através da linha de comando, o argumento **-f** ou **--file <string>**, que deve indicar o ficheiro onde deve ser efetuada a pesquisa. No caso de este não ser passado, a aplicação deve efetuar a contagem no ficheiro `/etc/services`.

A nível de implementação, a aplicação **abc-finder** deve contar as ocorrências de cada letra, com recurso a 26 processos, em que cada processo é responsável pela contagem da respetiva letra [A-Z].

Sugestões: 1) `man ascii`; 2) função `fgetc`; 3) função `toupper`;

Considere os seguintes exemplos que ilustram o funcionamento da aplicação:

```
./abc-finder
```

```
File: /etc/services
```

```
Process [E]: found 1119 'e' or 'E' occurrences!
```

```
Process [G]: found 216 'g' or 'G' occurrences!
```

```
Process [F]: found 188 'f' or 'F' occurrences!
```

```
...
```

```
Process [A]: found 713 'a' or 'A' occurrences!
```

```
./abc-finder -f data.txt
```

```
File: data.txt
```

```
Process [R]: found 35508 'r' or 'R' occurrences!
```

```
Process [Z]: found 34650 'z' or 'Z' occurrences!
```

```
...
```

```
Process [P]: found 36619 'p' or 'P' occurrences!
```

Pergunta 2 [10 valores]

(Escreva as suas respostas a esta pergunta no diretório "~/Prova01/R_NUMERO/Pergunta2". Deve indicar o seu nome completo e número de estudante IPEiria no ficheiro **README.txt** a ser criado no diretório)

NOTA 1: não é permitida a chamada a comandos externos através da função **system** ou de outra com funcionalidade similar.

NOTA 2: a solução deve ser implementada com recurso aos ficheiros existentes no arquivo **EmptyProject-Templatev3.05.zip**.

NOTA 3: código entregue que **não compile** leva à atribuição da classificação de **0 (zero) valores** à resposta.

Considere que uma determinada ponte pode suportar um máximo de **quatro** toneladas. Pretende-se que desenvolva, em linguagem C, a aplicação **theBridge**. A aplicação deve recorrer a threads e mecanismos de sincronização de threads para simular a passagem de um conjunto de veículos na ponte de modo a que o peso total dos veículos na ponte nunca exceda o peso máximo tolerado pela ponte. Cada *thread* deve simular o comportamento de um veículo. Para efeitos de simulação, considere que cada veículo demora **três** segundos a atravessar a ponte. A cada veículo deve ser atribuído um peso aleatório entre **uma** e **três** toneladas. Considere que existem **V** veículos, sendo que cada veículo passará **P** vezes na ponte ao longo da simulação (o sentido da travessia é irrelevante). Tendo em conta as limitações da ponte, um veículo só deve passar na ponte quando o seu peso somado ao peso dos veículos que estão atualmente na ponte não exceder o peso que a ponte pode suportar.

A aplicação deve processar os parâmetros da linha de comando através do utilitário **gengetopt**, implementando as seguintes opções.

--vehicles/-v <int>: número de veículos. Parâmetro obrigatório, cujo valor tem que ser positivo.

--passages/-p <int>: número de vezes que cada veículo passa na ponte. Parâmetro obrigatório, cujo valor tem que ser positivo.

Considere os seguintes exemplos ilustrativos do funcionamento da aplicação.

Exemplo 1	
<pre>./thebridge -v 4 -p 2 Weight V1: 1 Weight V2: 3 Weight V3: 1 Weight V4: 2 ----- V4 entered the bridge (total weight: 2) V3 entered the bridge (total weight: 3) V1 entered the bridge (total weight: 4) V4 left the bridge (total weight: 2) V4 entered the bridge (total weight: 4) V3 left the bridge (total weight: 3) V3 entered the bridge (total weight: 4) V1 left the bridge (total weight: 3) V1 entered the bridge (total weight: 4) V4 left the bridge (total weight: 2) V3 left the bridge (total weight: 1) V2 entered the bridge (total weight: 4) V1 left the bridge (total weight: 3) V2 left the bridge (total weight: 0) V2 entered the bridge (total weight: 3) V2 left the bridge (total weight: 0)</pre>	

Exemplo 2	Exemplo 3
<pre>./thebridge -v 0 -p 4 [ERROR] Invalid number of vehicles: 0</pre>	<pre>./thebridge -v 4 -p -3 [ERROR] Invalid number of passages: -3</pre>