
Duração: 150 minutos

Consulta: documentação instalada no computador e "cheat sheet"

Código público: ODG501 (<http://sigex.fe.up.pt/>)

1 Introdução

A prova consiste na resolução em Java ou em C do problema que se segue.

2 O Problema

Desenvolver um programa que 1) recolhe os valores de sensores de temperatura (usando o paradigma produtor/consumidor) e que 2) fornece o valor médio das temperaturas a clientes.

Para facilitar o desenvolvimento e a avaliação farão a implementação de cada uma destas funcionalidades em alíneas separadas.

3 Parte 1: Cliente-servidor

Implemente um servidor e respetivo cliente usando **TCP** e a linguagem que preferir.

O servidor deverá suportar uma única operação:

average a qual retorna o valor médio das temperaturas (recebidas dos sensores)

e executar indefinidamente esperando pedidos dos clientes e respondendo-lhes.

O cliente deverá enviar um único pedido, esperar pela resposta e imprimi-la.

Quer o servidor quer o cliente deverão imprimir na saída padrão (`System.out/stdout`) mensagens indicando: 1) o envio duma mensagem ; 2) a receção duma mensagem. As mensagens impressas deverão incluir as mensagens recebidas/enviadas ou seus sumários.

Compete-lhe a si definir o formato das mensagens trocadas entre cliente e servidor.

Invocação dos programas em Java O servidor deverá ser invocado da seguinte maneira:

```
java Server <port> <average>
```

onde

<port> é o porto do *socket* onde o servidor receberá os pedidos dos clientes, e que usará para enviar as respostas;

<average> é o valor (do tipo `float`) que deverá ser retornado nas respostas (atendendo a que esta versão não inclui código para receber as temperaturas).

O cliente deverá ser invocado da seguinte maneira:

```
java client <addr> <port>
```

onde

<addr> é o endereço IPv4 do servidor;

<port> é o porto do *socket* usado pelo servidor.

Invocação dos programas em C Os nomes dos programas servidor e cliente deverão ser `server` e `client`, respetivamente. Os argumentos são idênticos aos dos programas correspondentes em Java. Se usar a linguagem de programação C, deverá submeter um ficheiro ASCII com nome `client-server.txt` contendo os comandos para compilação dos seus programas.

4 Parte 2: Produtor-Consumidor

Nesta parte deverá desenvolver um consumidor e um produtor usando comunicação multicast. (Se não souber usar multicast, poderá usar comunicação UDP unicast, sofrendo uma penalização de 30%.)

O **produtor** deverá fazer o multicast das temperaturas medidas por um sensor. Estas medidas são argumentos do programa e deverão ser enviadas com um período de aproximadamente 1 segundo. (Pode usar a função `Thread.sleep()` / `sleep()`). Deverá terminar após ter enviado todas as mensagens.

O **consumidor** deverá executar um ciclo sem fim, no qual recebe as mensagens enviadas pelo produtor.

Quer o produtor quer o consumidor deverão imprimir mensagens na saída padrão quando da transmissão ou receção de mensagens, que deverão incluir as mensagens enviadas/recebidas ou seus sumários.

Compete-lhe a si definir o formato das mensagens enviadas pelo produtor.

Invocação dos programas em Java O programa consumidor deverá ser invocado:

```
java Consumer <addr> <port>
```

onde

<addr> é o endereço IPv4 multicast a subscrever pelo consumidor;

<port> é o porto do *socket* do grupo multicast a usar pelo consumidor;

O programa produtor deverá ser invocado da seguinte maneira:

```
java Producer <addr> <port> <val1> ... <valn>
```

onde

<addr> é o endereço IPv4 multicast subscrito pelo consumidor do servidor;

<port> é o porto do *socket* do grupo multicast usado pelo consumidor;

<val[1..n]> é o valor (do tipo `float`) que deverá ser enviado em cada uma das mensagens).

Invocação dos programas em C Os nomes dos programas consumidor e produtor deverão ser `consumer` e `producer`, respetivamente. Os argumentos são idênticos aos dos programas correspondentes em Java.

Se usar a linguagem de programação C, deverá submeter um ficheiro ASCII com nome `producer-consumer.txt` contendo os comandos para compilação dos seus programas.

5 Documentação e Submissão

A documentação da API de JDK 1.6 pode ser encontrada no seguinte URL: `file:///opt/docs/`

Para submeter a prova deverá omprimir num ficheiro `.zip` único todos os ficheiros com a sua solução, i.e. ficheiros com o código (caso tenha desenvolvido o seu programa em C deverá também incluir os ficheiros com os comandos de compilação). **IMP.** O ficheiro `.zip` deverá incluir apenas ficheiros, não directórios.