

## Sistemas Distribuídos

Exame<sup>1</sup>

29 de janeiro de 2020

Duração: 2h00m

### I

Responda a cada um dos 2 grupos em folhas de teste separadas.

Não ultrapasse 1/2 página (15 linhas) em cada resposta neste grupo.

- 1 Explique as vantagens e desvantagens relativas da utilização de formatos de texto (por oposição a formatos binários) para serialização de dados em sistemas distribuídos.
- 2 Descreva a arquitetura de um sistema de invocação remota de métodos em sistemas distribuídos orientados por objetos, justificando o objetivo de cada um dos componentes.
- 3 Explique sucintamente em que consiste uma DHT e justifique a sua necessidade em termos dos objetivos (*design goals*) de sistemas distribuídos.

### II

Responda a cada um dos 2 grupos em folhas de teste separadas.

Considere um sistema de partilha de ficheiros, tal como o DropBox, que permite manter cópias locais e sincronizar alterações. De forma a evitar os conflitos que surgem quando vários utilizadores modificam o mesmo ficheiro, pretende-se um sistema que auxilie a coordenação entre utilizadores para os evitar. Este sistema deve permitir que sejam realizadas atomicamente cópias de segurança de todos os ficheiros modificados.

- 1 Pretende-se que escreva em Java, fazendo uso de primitivas baseadas em monitores, uma classe que implemente a seguinte interface para ser usada no servidor por *threads* que representam utilizadores de ficheiros:

```
interface Ficheiros {  
    void using(String path);  
    void notUsing(String path, boolean modified);  
    List<String> startBackup();  
    void endBackup();  
}
```

Quando um utilizador pretende modificar um ficheiro, deve pedir licença com o método *using*, indicando o nome do ficheiro. A licença para utilizar um ficheiro só pode ser dada a um utilizador de cada vez. O método *notUsing* assinala que já não está a usar o ficheiro, indicando o seu nome e se foi modificado.

Para iniciar uma cópia de segurança usa-se o método *startBackup*, que espera até os ficheiros não estarem a ser usados e devolve a lista de todos os ficheiros modificados desde a cópia de segurança anterior. Depois de terminada a cópia de segurança, o utilizador deve usar o método *endBackup*.

**Valorização:** *i)* Se existirem vários pedidos para o mesmo ficheiro, devem atender-se por ordem de chegada. *ii)* Permita que os ficheiros que não fazem parte de uma cópia de segurança (em curso) possam continuar a ser modificados e que serão incluídos na cópia seguinte. Indique na sua resposta quais destas garantias oferece e minimize a necessidade de acordar *threads* bloqueados.

- 2 Implemente o programa servidor usando *threads*, *sockets* TCP, e a classe desenvolvida na pergunta anterior.

<sup>1</sup>Cotação — I: 3 + 3 + 4; II: (4 + 3) + 3