

# Exame de Recurso — Redes de Computadores de 2018/2019

(Preencha sem falta todos os campos abaixo)

Nome: \_\_\_\_\_ N.º \_\_\_\_\_

Sala do exame: \_\_\_\_\_ Edifício VII Versão: **B** Venho repetir o teste 2 (S/N): \_\_\_\_\_

Duração: 2 horas e 15 minutos acrescidos de 15 minutos de tolerância

A interpretação das questões é da sua responsabilidade.

Não pode usar dispositivos electrónicos de qualquer tipo.

Nas respostas de escolha múltipla com várias hipóteses, as hipóteses erradas descontam. Se seleccionar todas as hipóteses, a cotação final é nula.

## RESPOSTA ÀS QUESTÕES - PODE USAR LÁPIS

- 1.1) **2**  
**outros diminuir 25%**
- 1.2) **4**
- 1.3) **3**
- 1.4) **1**
- 1.5) **1,7,8 cada vale 33,3%**  
**descontar 15% por errada**
- 1.6) **2,6,7 idem anterior** (medianas)
- 1.7 a) **8** 1.7 b) **10** 1.7 c) **6 ou 7 - 100%** 1.7 d) **2** (medianas)  
**8 - 10%** **resto nada**
- 1.8 a) **40** 1.8 b) **14 ou 15 ou 16** (difícil)
- 2.1) **3** 2.2) **2** 2.3) **3** 2.4) **4** (medianas)
- 2.5) **3 vale 100%** 2.6) **2,3 cada vale 50**  
(2.5 difícil) **outras descontar 20%** **outras, descontar 50%** (2.6 mediana)
- AS PERGUNTA 2.7 e 2.8 DEVEM SER RESPONDIDAS NO RESPECTIVO ENUNCIADO NO VERSO
- 2.7) (2.7 difícil)  
2.8) (2.8 mediana)

**2.7 - VERSÃO B - vale 2 valores)** Considere a rede da pergunta **2.6** com as seguintes hipóteses: o tempo de propagação do canal que liga R3 a R5 e vice-versa é 10 vezes **maior** que o dos outros canais; a rede usa um protocolo de encaminhamento baseado no algoritmo Bellman-Ford com anúncios periódicos, **com memorização dos anúncios e split horizon with poison reverse**.

É possível entrar-se num ciclo de contagem para o infinito nesta rede? Justifique a sua resposta.

**SIM podem (acertar vale 20%)**

**Se o canal que liga R1 a R2 for abaixo, todos os routers recebem o aviso de que a distância agora é infinito.**

**Mas R5 e R4 conhecem um caminho pior (em backup) via o outro pois como não o usam, é anunciado ao outro.**

**Como o tempo de propagação do link R5/R4 é muito grande, R4 e R5 não têm tempo de se aperceberem que esses caminhos também já não estão disponíveis, e podem reintroduzi-los na rede o que conduz ao ciclo.**

**a justificação vale 80%**

**2.8 - VERSÃO B - vale 2 valores)** Numa rede, os computadores têm vários vizinhos, a cada um dos quais estão ligados directamente por um canal distinto ponto a ponto. Por hipótese, os **N** destinos existentes na rede são os computadores, que são identificados por um número inteiro de **1 a N**. Na rede usa-se o algoritmo Bellman-Ford na versão **split horizon with poison reverse** para actualizar as tabelas de encaminhamento.

Um anúncio é um vetor com **N** entradas em que cada entrada indica o **custo** para chegar ao destino correspondente à mesma. Por exemplo, **costs ( i )**, indica o custo com que o emissor deste anúncio chega ao destino **i**.

Um computador tem uma **RIB** que suporta vários métodos, entre os quais:

**int getGateway ( destination )** - devolve o vizinho que encaminha para **destination** ou null se **a RIB** desconhece **destination**

**int getDistance ( destination )** - devolve o custo do caminho deste computador até **destination** ou **infinity** se **a RIB** desconhece **destination**

**put ( destination, distance, gateway)** - cria ou actualiza a entrada de **RIB** referente a **destination**.

Escreva o pseudo código usado pelo computador para processar um anúncio que acabou de receber do vizinho **K** à distância **D**.

```
RIB rib = new RIB(N); // a tabela de encaminhamento do computador contendo N entradas
int[] costs = new int[N]; // o anúncio a processar
```

```
for ( int i = 1; i <= N; i++ ) { // ciclo vale 10%

    if ( i == myself ) continue ; // se esquecer não desconta nada pois rib.getDistance(myself) deve ser 0

    if ( rib.getGateway(i) == K ) rib.put ( i, costst[i] + D, K ) // I use gateway so I always update // 45%

    else if ( rib.getDistance(i) > costs[i]+D ) rib.put( i,costst[i]+D, K ) // I use this better path // 45%
        // if i is a new destination getDistance(i) = infinity;
        // as duas condições não se podem juntar

}
```