 <small>ESCOLA SUPERIOR DE TECNOLOGIA E GESTÃO</small>	Tipo de Prova Exame Época Normal	Ano letivo 2019/2020	Data 31-01-2020
	Curso Licenciaturas em Engenharia Informática / Segurança Informática em Redes de Computadores	Hora 14h30	
	Unidade Curricular Sistemas Operativos	Duração 02h10	

Observações

- Com consulta de documentação própria.
- O tempo previsto para responder a cada questão é apresentado entre parêntesis retos.
- A cotação atribuída a cada pergunta é apresentada entre parêntesis curvos.

PARTE I (Teórica) [70 min]

Questão 1 [10 min]

[2.0 valores]

Para cada uma das seguintes afirmações deverá indicar se as considera verdadeiras ou falsas. Caso considere alguma afirmação como falsa deverá rescreve-la, transformando-a numa afirmação verdadeira. À simples negação não será atribuída nenhuma cotação.

- Sistemas paralelos são exemplos de sistemas mono-programados [0.5 valores];
- Embora consumam mais recursos físicos, as soluções de hardware são as mais adequadas em termos de portabilidade entre sistemas para tratar de situações de competição [0.5 valores];
- Problemas de mútua exclusão e impasse ficam completamente resolvidos quando são utilizados semáforos [0.5 valores];
- Na comunicação direta entre dois processos, independentemente do tipo de endereçamento escolhido, o emissor da mensagem tem sempre que indicar o recetor da mensagem [0.5 valores];

Questão 2 [10 min]

[2.0 valores]

"A estratégia de prevenção para lidar com situações de impasse é mais eficiente em termos de utilização de recursos do que a estratégia de evitar."

Comente a afirmação, indicando também se **concorda ou não** com a mesma. Fundamente a sua resposta com um exemplo concreto.

Questão 3 [10 min]

[2.0 valores]

Assuma um sistema com **8KBytes** de memória virtual por **paging** com páginas de **512Bytes**. Indique, recorrendo à técnica MMU e à tabela seguinte:

a) a que endereço físico corresponde o endereço virtual **2990**; [1.0 valores]

b) a que endereço físico corresponde o endereço virtual **290**; [1.0 valores]


110	1
011	0
010	1
000	1
100	0
001	1

Questão 4 [10 min]

[2.0 valores]

Considere que um sistema operativo, num dado instante de tempo, dispõe da seguinte lista de partições de memória:

[p1|70K] - [h|56K] - [p3|45K] - [h|120K] - [p2|51K] - [h|82K]

 <small>ESCOLA SUPERIOR DE TECNOLOGIA E GESTÃO</small>	Tipo de Prova Exame Época Normal	Ano letivo 2019/2020	Data 31-01-2020
	Curso Licenciaturas em Engenharia Informática / Segurança Informática em Redes de Computadores	Hora 14h30	
	Unidade Curricular Sistemas Operativos	Duração 02h10	

Apresente a lista resultante da aplicação do **worst-fit** para cada um dos seguintes eventos (nota: espaços livres contíguos devem estar juntos):

1. Novo processo (P4) com 100K de tamanho;
2. Novo processo (P5) com 10K de tamanho;
3. Término dos processos (P1 e P3) e libertação dos recursos associados;
4. Novo processo (P6) com 154K de tamanho;
5. Novo processo (P7) com 20K de tamanho;
6. Término (P2 e P5) e libertação dos recursos associados;
7. Novo processo (P8) com 151K de tamanho;

Questão 5 [10 min]

[2.0 valores]

Considere o seguinte conjunto de processos. Assuma o instante de chegada e duração de cada processo aqueles indicados na tabela seguinte:

Processo	Instante de chegada	Duração
P1	0.1	0.5
P2	0.3	0.7
P3	0.7	0.9
P4	0.9	0.4
P5	1.2	0.3

Calcule o tempo médio de turnaround, considerando que o algoritmo de escalonamento é o **RR** (Round-Robin, com quantum de 0.5). Fundamente a sua resposta com todos os cálculos que sentir necessidade de efetuar.


Questão 6 [10 min]

[2.0 valores]

Assuma um sistema com os tipos de recursos (A, B, ...), processos (P1, P2, ...) e caracterização como apresentada nas tabelas seguintes:

Alocado						Necessidades máximas						Disponibilidade				
	A	B	C	D	E		A	B	C	D	E	A	B	C	D	E
P1	1	0	1	1	0	P1	1	0	1	2	2	2	1	0	1	2
P2	0	0	0	0	0	P2	0	3	2	1	0					
P3	1	0	1	1	0	P3	1	2	1	2	1					
P4	0	1	0	1	1	P4	2	1	1	2	1					
P5	1	1	0	0	1	P5	5	1	1	0	2					

Aplique o algoritmo do banqueiro, de forma a garantir uma sequência de execução que mantenha o sistema num estado seguro. Justifique a sua resposta com todos os cálculos que sentir necessidade de efetuar.

 <small>ESCOLA SUPERIOR DE TECNOLOGIA E GESTÃO</small>	Tipo de Prova Exame Época Normal	Ano letivo 2019/2020	Data 31-01-2020
	Curso Licenciaturas em Engenharia Informática / Segurança Informática em Redes de Computadores	Hora 14h30	
	Unidade Curricular Sistemas Operativos	Duração 02h10	

PARTE II (Prática) [60 min]

Questão 7 [30 min]


[4.0 valores]

Considere o seguinte excerto de código:

```

1 class Tempo {
2     static void sleep( long millis ) {
3         try { Thread.sleep(millis); } catch ( InterruptedException ie ) {}
4     }
5 }
6
7 public class EpocaNormal implements Runnable {
8     Sinal s;
9     int i;
10
11     public EpocaNormal( Sinal s, int i ) {
12         this.s = s;
13         this.i = i;
14     }
15
16     public void run() {
17         int i = 0;
18
19         do {
20             s.doWait();
21             s.setNumber( s.getNumber() + 1 );
22         } while ( i++ < this.i );
23
24         Tempo.sleep( 100 );
25         System.out.println( Thread.currentThread().getName() + ":" + s.getNumber() );
26
27         s.doRun();
28     }
29
30     public static void main( String args[] ) {
31         final Sinal s = new Sinal();
32
33         for ( int i = 0; i < 5; i++ ) {
34             ( new Thread( new EpocaNormal( s, i ), "Th" + i ) ).start();
35         }
36
37         Tempo.sleep( 100 );
38         s.doRun();
39
40         System.out.println( "Todas as threads terminaram" );
41     }
42 }
43
44

```

 <small>ESCOLA SUPERIOR DE TECNOLOGIA E GESTÃO</small>	Tipo de Prova Exame Época Normal	Ano letivo 2019/2020	Data 31-01-2020
	Curso Licenciaturas em Engenharia Informática / Segurança Informática em Redes de Computadores	Hora 14h30	
	Unidade Curricular Sistemas Operativos	Duração 02h10	

```

45 class Sinal {
46     private int number = 0;
47
48     protected synchronized void doRun() {
49         this.notifyAll();
50     }
51
52     protected synchronized void doWait() {
53         try { this.wait(); } catch ( InterruptedException ie ) {}
54     }
55
56     protected void setNumber( int number ) {
57         this.number = number;
58     }
59
60     protected int getNumber() {
61         return this.number;
62     }
63 }

```

- [5 min] Qual será o resultado esperado da sua execução? [1.0 valores]
- [5 min] Existe a possibilidade de surgir outro resultado diferente daquele que indicou na alínea anterior? Justifique a sua resposta. [1.0 valores]
- [15 min] Mantendo a estrutura base e **recorrendo a semáforos**, altere o código apresentado de forma a:
 - eliminar possíveis *race conditions*; 2) garantir que a linha 40 só é executada quando todos os (child) threads já terminaram. [2.0 valores]

Questão 8 [30 min]

[4.0 valores]

Escreva três classes, **Printer** e **Desktop**, que partilham um **Job** para leitura e escrita de trabalhos de impressão entre elas (Desktop registra/escreve trabalho de impressão em Job, enquanto Printer lê trabalho de impressão de Job). Job pode conter até 5 trabalhos de impressão. Um trabalho de impressão é uma string (qualquer). **Em cada instante estão em execução uma Printer (i.e., um thread Printer) e vários Desktops (i.e., vários threads Desktop).** Pretende-se que o seguinte funcionamento seja implementado (em ciclo):

- Printer está adormecida, sendo acordada quando um Desktop regista um pedido (i.e., escreve um trabalho de impressão em Job);
- quando acordada, Printer imprime o trabalho, ou seja, imprime a string (pode ser um número, para facilitar a implementação), e volta a adormecer se não houver mais pedidos;
- Printer atende os pedidos de impressão por ordem de chegada (o mais antigo primeiro e o mais recente em último);
- apenas um Desktop de cada vez pode fazer registar um trabalho de impressão em Job num determinado momento;
- um Desktop só regista o seu trabalho de impressão se houver menos de 5 trabalhos por imprimir registados até ao momento;

Nota: o código deve garantir a não existência de situações de competição (*race conditions*).