universidade de aveiro theoria poiesis praxis

MIEET, MIECT-41475 **Sistemas Tempo-Real** 2009/2010

5º Ano, 1º Semestre

Exame

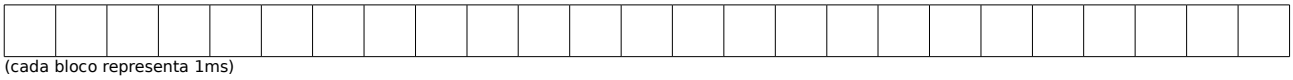
15/Jan/2010, duração: 2H00

N.º mec.: _____ Nome:_____

I) Considere o conjunto de tarefas definido na tabela abaixo, onde estão discriminados os respectivos instantes de activação, tempo de execução e deadline. Assuma que as mudanças de contexto requerem um tempo desprezável.

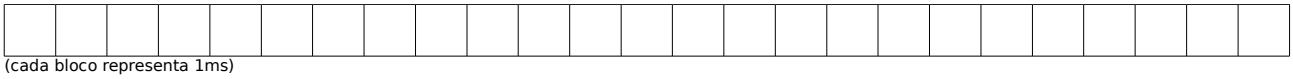
Tarefa	Activação (ms)	Tempo de execução (ms)	Deadline (ms)
T1	7	3	10
T2	0	11	15
T3	0	9	30
T4	5	2	18

I.1) Desenhe o escalonamento das 4 tarefas usando uma política de escalonamento com preempção em que se escalona em primeiro lugar a tarefa com a deadline absoluta mais curta.



(cada bloco representa 1ms)

I.2) Assuma que são alocadas às tarefas {T1, T2, T3, T4} as prioridades {5,15,10,15} em que um maior número corresponde uma maior prioridade. Desenhe o escalonamento destas 4 tarefas usando uma política que aloca o CPU à tarefa com maior prioridade. Caso duas tarefas possuam a mesma prioridade deve ser favorecida a tarefa que tiver sido activada mais cedo.



(cada bloco representa 1ms)

I.3) Considere agora que as tarefas são periódicas, com período igual à deadline. Indique, justificando, se o conjunto de tarefas é escalonável com os critérios RM e EDF.

II) Um certo sistema de tempo-real deve controlar a temperatura em dois pontos de um reactor nuclear, as quais são supostas ser sempre iguais. Se estas forem diferentes é possível que exista um mau funcionamento do sistema e deve accionar-se um alarme.
A temperatura é lida periodicamente pelo hardware e escrita em dois registos mapeados em memória (REG0 e REG1). Pode assumir-se que esta escrita é efectuada em simultâneo. Sempre que há uma nova leitura é gerado automaticamente uma interrupção ao CPU que provoca a execução da ISR ***readtemperature()***. A função ***main()*** possui um ciclo infinito em que está constantemente a comparar ambas as temperaturas e acciona um alarme se estas forem diferentes.

```
#define MEMMAPPED REG0 0x40001000
#define MEMMAPPED REG1 0x40001008
volatile int *temp0_ptr=(int *)MEMMAPPED REG0;
volatile int *temp1_ptr=(int *)MEMMAPPED REG1;
volatile int Temperature[2] ;

void irq readtemperature ( void )
{
        Temperature[0] = *temp0_ptr ; // Ler valor do hardware
        Temperature[1] = *temp1_ptr ; // Ler valor do hardware
}

void main ( void )
{
        int temp0 , temp1 ;
        while (TRUE)
        {
                temp0 = Temperature[0] ;
                temp1 = Temperature[1] ;
                if ( temp0 != temp1 ) {
                        // Ligar alarme e acordar o Homer Simpson
                }
        }
}
```

II.1) O código apresentado funciona correctamente? Caso entenda que não justifique apresentando um traço de execução em que tal seja observável.

II.2) Admita um cenário hipotético em que as funções ***main()*** e ***readtemperature()*** são tarefas de um sistema operativo tempo-real, com preempção. O código funcionaria correctamente? Justifique.

II.3) Nas condições da alínea anterior, admita que para permitir o correcto funcionamento é necessário garantir acesso atómico às variáveis Temperature{[0],[1]}. Indique, justificando, se considera mais aconselhável usar um semáforo, desligar as interrupções ou desligar a preempção.

III) Considere o seguinte conjunto de tarefas:

T1 = (10, 100, 100)
T2 = (20, 200, 200)
T3 = (150, 600, 600)
T4 = (240, 1200, 1200),

especificado na forma Ti=(Ci,Ti,Di), i={1,2,3,4}.

III.1) Qual a taxa de utilização do conjunto de tarefas?

III.2) Usando exclusivamente critérios de análise de escalonabilidade baseados em taxa de utilização, o que pode concluir-se acerca da escalonabilidade do sistema caso seja usado RM? E se for usado EDF? Justifique!

III.3) Calcule o tempo de resposta de cada uma das mensagens, admitindo escalonamento RM com preempção.

III.4) Admita que o sistema é escalonado com RM e a tarefa T1 é aperiódica, sendo servida por um “Polling Server” com (C,T)=(10,100). A escalonabilidade do sistema altera-se? Justifique!

III.5) Admita um cenário semelhante à alínea anterior, mas em que T1 é servida por um "Deferrable Server". A escalonabilidade do sistema altera-se? Justifique!

III.6) Sabendo que as tarefas acima comunicam com nó remoto através de uma porta de I/O comum a todas, e que em cada acesso esta porta de I/O é usada em modo exclusivo durante 1ms, qual o bloqueio que esse facto poderá causar às várias tarefas, se se usar semáforos com o protocolo "Priority Inheritance Protocol"?

III.7) Considere agora que as tarefas acima definidas são de instância única. Construa a árvore de escalonamentos sem preempção logicamente possíveis. Destes indique quais são praticáveis.