

Eng.ª Informática - Cadeira de Sistemas Operativos

Exame de 1ª Época

1 de Março de 2006

Leia atentamente as perguntas seguintes e responda de forma breve e precisa. Pode acompanhar a suas respostas com figuras de forma a torná-las mais explícitas.

1. Estrutura de Sistemas Operativos (4 valores)

- 1.1 Dê uma definição do conceito de Sistema Operativo e indique algumas das suas funcionalidades fundamentais.
- 1.2 O que entende por excepções? Indique as razões pelas quais são mecanismos essenciais para o funcionamento dos Sistemas Operativos.
- 1.3 Para que servem as Interrupções num Sistema Operativo? Explique como é que o sistema “sabe” qual a rotina que deve executar na sequência de uma interrupção.
- 1.4 Que tipos de interface utilizador fornecem os sistemas operativos para computadores destinados a uso interactivo? Dê exemplos de cada um dos tipos que referir.

2. Gestão de Processos (4 valores)

Considere o seguinte fragmento de programa:

```
int main(int argc, char *argv[])
{
    pid_t pid;

    pid = fork();
    if (pid == 0) {

        // Secção 1
    } else {

        // Secção 2
    }
}
```

- 2.1 Indique a que correspondem a Secção 1 e a Secção 2 deste programa.
- 2.2 Escreva o código de uma dessas secções de maneira que seja executado o comando correspondente ao primeiro argumento do programa acima indicado.
- 2.3 O que é a Comutação de Processos? Descreva as fases que a constituem.

3. Escalonamento de Processos (4 valores)

A Figura 1 representa o cronograma de execução de quatro processos P1, P2, P3 e P4, onde o instante de chegada respectivo é representado pela seta por baixo do número do processo.

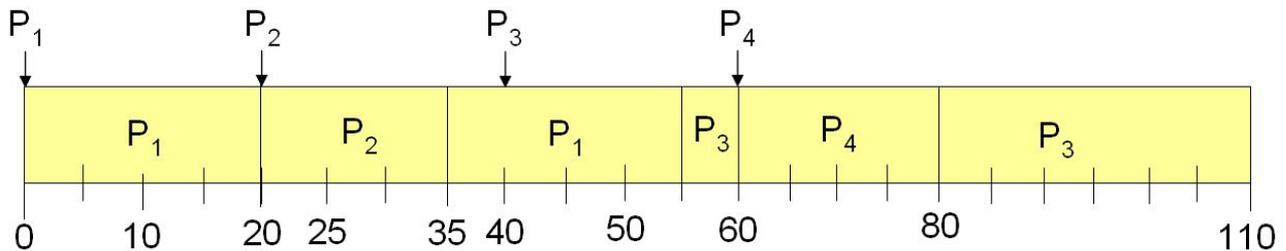


Figura 1

- 3.1 Sabendo que o algoritmo de escalonamento utilizado é o **Shortest Job First (SJF)** com **Preempção**, construa a tabela das características dos processos, na qual deverão aparecer os tempos de chegada e os respectivos ciclos de CPU previstos.
- 3.2 Desenhe o cronograma de execução dos mesmos processos, no caso do algoritmo de escalonamento ser **Round Robin** com um **Time Quantum** de 20.
- 3.3 Calcule o tempo médio de espera dos processos em ambos os casos. Qual dos algoritmos lhe parece mais eficaz?

4. Sincronização de Processos (4 valores)

Considere as seguintes funções que contam o número alunos que entram e saem de uma universidade, de forma a permitir manter sempre actualizado o número de vagas para as inscrições.

A função **contaEntradas()** é activada quando um aluno se inscreve e a **contaSaidas()** quando um aluno finaliza o curso. Finalmente a função **mostraVagas()** imprime o número de vagas livres em qualquer momento, podendo estas três operações ser executadas **simultaneamente** em threads ou processos independentes, mas que **partilham** a variável global **numLugares**.

```
global numLugares = NUM-MAX-LUGARES;
```

```
contaSaidas ()  
{  
    numLugares = numLugares + 1;  
}
```

```
contaEntradas ()  
{  
    numLugares = numLugares - 1;  
}
```

```
mostraVagas ()  
{  
    print (numlugares);  
}
```

- 4.1 Defina o tipo de concorrência existente entre estas três funções e identifique a respectiva secção crítica.
- 4.2 Introduza variáveis de sincronização e modifique o código das funções de forma a garantir a coerência dos resultados da sua execução.
- 4.3 Qual é a diferença entre um semáforo binário (ou mutex) e um semáforo contador?

5. Gestão de Memória (4 valores)

5.1 Acha que Sistemas Operativos multiutilizador tais como o Linux ou o Windows poderiam funcionar correctamente num computador que não dispusesse de uma Unidade de Gestão de Memória, ou *Memory Management Unit* (MMU)? Justifique.

A Figura 2 representa os espaços de endereçamento lógicos de dois processos P₁ e P₂, e a memória física onde estão carregadas as respectivas páginas. O sistema utiliza tabelas de páginas directas.

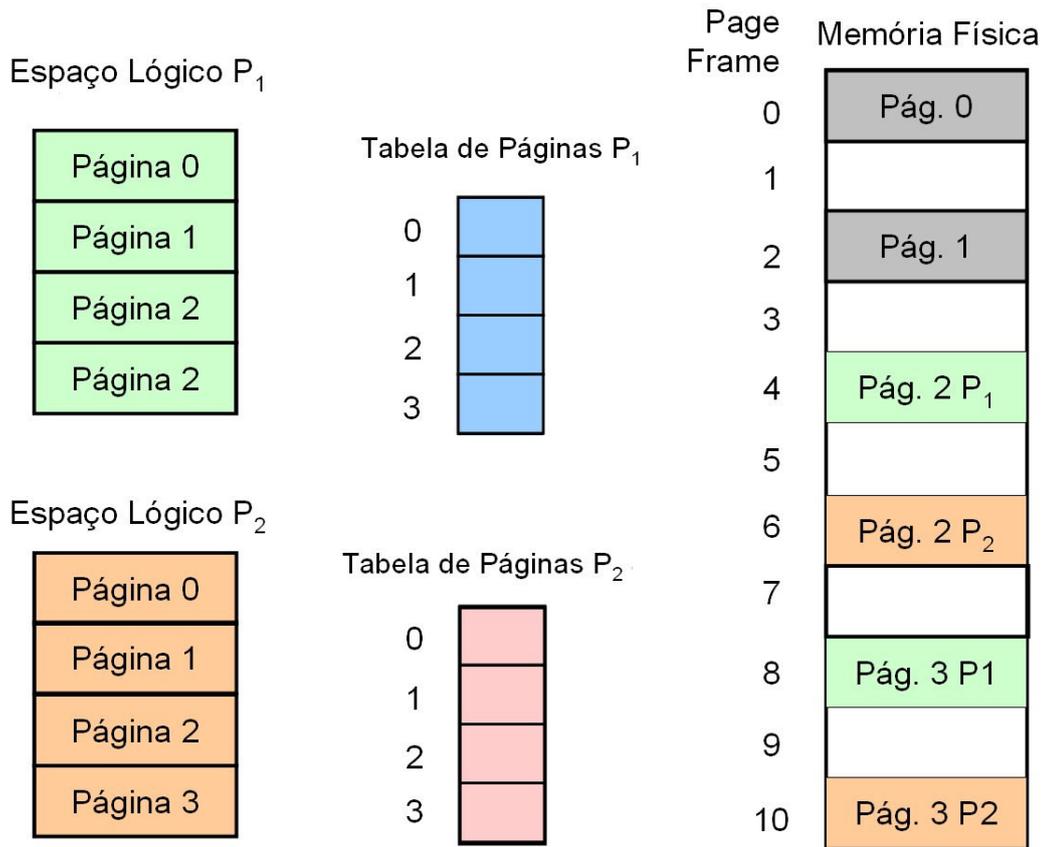


Figura 2

5.2 Tendo em conta a disposição em memória física representada na Figura 2, preencha as duas tabelas de páginas de maneira a que os dois processos partilhem o conteúdo das páginas 0 e 1 e tenham conteúdos distintos nas 3 e 4.

5.3 As tabelas de páginas directas podem ser utilizadas em sistemas com espaços de endereçamento lógico iguais ou superiores a 2³² bits? Justifique.