

Número de aluno:

PREENCHER EM MAIÚSCULAS

Nome:



Sistemas Operativos

2012/13

DEIS – Engenharia Informática

Prova escrita
Parte teórica
Época normal
5 / Julho / 2013

Todas as perguntas valem o mesmo. Respostas erradas descontam 1/3. Respostas múltiplas são respostas erradas.
As repostas são dadas nestas folhas. Preencha já o cabeçalho com o nome.

1 – No contexto dos sistemas estudados, qual das seguintes afirmações é correcta?

- ☐ Pode-se instalar um sistema em qualquer tipo de sistema de ficheiros desde que haja espaço.
- ☐ Faz sentido existirem mais do que uma partição activa, devendo ser seleccionada uma para arranque através de um menu.
- ☐ O sector boot determina qual a partição que está activa.
- ☐ O sector boot contém código executável.

2 – Durante a inicialização da máquina:

- ☐ O sector MBR passa o controlo para a BIOS.
- ☐ O sector boot apresenta um menu para determinar qual o sector MBR a carregar.
- ☐ O sector boot de uma partição pode carregar o sistema operativo ou passar o controlo a outro sector boot.
- ☐ A BIOS carrega directamente o sistema operativo. Os sectores MBR e boot servem para descrever a geometria do disco.

3 – O processo 200 tinha desencadeado uma operação de escrita em disco que acaba de terminar neste instante com um código de erro. O processo 300, que por acaso é menos prioritário que o 200, estava a executar no preciso instante em que a operação do processo 200 se conclui. Quais as transições de estado que ocorrem nesse instante exacto?

- ☐ O processo 200 passa para terminado por causa do erro na escrita.
- ☐ O processo 200 passa de bloqueado para execução e o processo 300 para executável.
- ☐ O processo 200 estava em execução e continua para tratar o erro, e para já mais nada acontece.
- ☐ O processo 200 passa para executável e mais nada para já.

4 – Um processo acaba de receber um sinal e vai trata-lo. O processo estava em:

- ☐ Execução.
- ☐ Bloqueado.
- ☐ Suspenso.
- ☐ Podia estar em pelo menos dois dos três anteriores.

5 – A relação entre processo e programa é:

- ☐ Um programa tem um processo.
- ☐ Um processo tem um programa.
- ☐ São termos que designam a mesma coisa.
- ☐ As anteriores estão todas certas.

6 - Qual das seguintes afirmações é correcta?

- ☐ Os sistemas multiprogramados interactivos optimizam o throughput relativamente aos multiprogramados não interactivos.
- ☐ Os sistemas multiprogramados têm uma ligeira perda de performance face aos monoprogramados devido ao maior número de comutações de processo.
- ☐ Os sistemas que obrigam um processo a sair do processador por ter passado um tempo máximo são preemptivos sendo todos os outros tipos de sistema não preemptivos.
- ☐ Os sistemas multiprogramados interactivos não têm por objectivo principal a optimização da performance global (maior numero de tarefas executadas no menor tempo possível).

7 – Observe o código à direita. Assumindo que não há erros de execução, quantas vezes aparecerá a palavra “FIM”?

- ☐ Quatro vezes.
- ☐ Oito vezes.
- ☐ Uma vez.
- ☐ Infinitas vezes.

```
int main() {  
    int a = 0;  
    while (a < 3) {  
        if (fork() != 0)  
            a++;  
    }  
    printf("FIM");  
    return 0;  
}
```

8 - Qual das seguintes afirmações é verdadeira?

- ☐ As aplicações não fazem ideia acerca do endereço onde as funções sistema se encontram.
- ☐ As chamadas às funções sistema são efectuadas da mesma forma que as chamadas às funções biblioteca.
- ☐ Genericamente, se o código das funções sistema for reproduzido numa aplicação, obter-se-ão versões “locais” à aplicação dessas funções sistema, não sendo necessário invocar as originais do sistema.
- ☐ O código das funções sistema é linkado com o código da aplicação.

9 – No contexto do processo A, a execução atinge um endereço onde se encontra uma instrução desconhecida (mais precisamente, um opcode para uma instrução inexistente). O que acontece?

- ☐ O programa gera uma interrupção.
- ☐ O processador gera uma excepção.
- ☐ O processador marca a instrução como inválida e avança para o endereço seguinte.
- ☐ A página onde a instrução se encontra é marcada como inválida.

10 - O programa "abc", a executar no processo 100, executa a seguinte linha.

```
execl("programaB", "programaB", NULL);
printf("\nEstou aqui");
```

Todas as funções invocadas são bem sucedidas. Qual das afirmações descreve melhor o que ocorre?

- ☐ A mensagem "Estou aqui" não aparece".
- ☐ A mensagem "Estou aqui" aparece logo.
- ☐ A mensagem "Estou aqui" aparece só depois de "programaB" terminar.
- ☐ Não é possível saber se a mensagem "Estou aqui" aparece ou não com os dados indicados.

11 - Considere o programa "abc" apresentado à direita. Todas as funções invocadas são bem sucedidas. O que aparece no ecrã?

- ☐ "1 ABC" e mais nada.
- ☐ "1 ABC" "2 ABC" "3 ABC" "4 ABC" "5 ABC"
- ☐ "1 ABC" 5 vezes.
- ☐ "1 ABC" infinitas vezes.

Programa "abc"

```
int x; /* global inicializada a 0 */
int main() {
    x++;
    printf("%d ABC ", x);
    if (x==10) exit(0);
    execl("abc", "abc", NULL);
    return 0;
}
```

12 - O processo 101 envia neste instante o sinal SIGUSR1 ao processo 102. Por sua vez, o processo 102 tinha executado com sucesso há cerca de 5 minutos a seguinte linha:

```
signal(SIGUSR1, fazEAcontece);
```

Neste instante o processo 102 está no meio de um ciclo de cálculo intensivo, sem qualquer operação de entrada e saída e vai manter-se nesse ciclo alguns minutos ainda. Qual das afirmações está correcta?

- ☐ Após o envio do sinal, o processo 101 vai ficar à espera (alguns minutos) até que o processo 102 acabe o ciclo de cálculo intensivo e execute a função fazEAcontece().
- ☐ O processo 102 vai ignorar o sinal porque estava ocupado no instante em que o sinal lhe é enviado.
- ☐ O processo 101 continua a sua execução e o processo 102 interrompe o que estava a fazer e executa a função fazEAcontece().
- ☐ Os processos 101 e 102 não têm nada a ver um com o outro e continuam a sua execução de forma inalterada.

13 - Num determinado sistema foram lançados os processos A,B,C,D,E nos instantes 0,1,2,3,4, respectivamente. Estes processos têm a seguinte duração, respectivamente: 4,2,2,4,2. Os processos são do tipo CPU-Bound e foram executados de acordo com o seguinte diagrama:

A	B	C	D	E	A	B	C	D	E	A	D	A	D
---	---	---	---	---	---	---	---	---	---	---	---	---	---

Qual o algoritmo de escalonamento usado pelo sistema?

- ☐ First Come First Served.
- ☐ Round Robin.
- ☐ Shortest Process Next.
- ☐ Shortest Remaining Time.

Número de aluno:

PREENCHER EM MAIÚSCULAS

Nome:

14 - Qual das seguintes está correcta?

- ☐ O algoritmo Round Robin é preemptivo e justo, mas a justiça não é necessariamente uma característica boa.
- ☐ O algoritmo Shortest Process Next otimiza os tempos de espera e é mais estável (fiável) do que o Round Robin.
- ☐ O algoritmo FCFS prejudica os processos grandes.
- ☐ O algoritmo HRRN tem por objectivo amentar a fluidez de interacção com os utilizadores.

15 - Considere um processo em memória virtual segmentada cujo espaço de endereçamento se encontra descrito na tabela abaixo.

Índice	P	Prot	Base	Limite
0	0	RW	10240	2048
1	0	RWX	8192	4096
2	1	RX	2048	6144

Qual o endereço real que corresponde ao endereço virtual (1, 5000)?

- ☐ Nenhum.
- ☐ Endereço 13192 ($8192 + 5000$).
- ☐ Endereço 2052 ($2048 + 4$).
- ☐ Um endereço real que não é possível saber neste momento.

16 – Em relação à memória paginada, a memória segmentada:

- ☐ Ajusta-se mais facilmente à estrutura lógica dos programas.
- ☐ É mais eficiente (mais rápida).
- ☐ É gerida da mesma forma (que a memória paginada).
- ☐ Estão todas certas.

17 – Qual o espaço de trabalho mais provável para o processo em memória paginada cujo código se apresenta à direita?

- ☐ Um terço das suas páginas.
- ☐ Dois terços das suas páginas.
- ☐ Todas as suas páginas.
- ☐ Nenhuma das anteriores. O espaço de trabalho só se aplica a memória segmentada.

```
int matriz[100000000];

main() {
    int indice;
    while (1) {
        indice = (int) (100000000 * random());
        matriz[indice] = (matriz[indice] + 30) / 2;
    }
}
```