

- Não é permitida a consulta de livros ou de apontamentos. Por norma não se esclarecem dúvidas durante a prova. Se tiver dúvidas, indique na folha de teste a sua interpretação. Utilize uma caligrafia legível.

Grupo A: Gestão de Memória (10 Valores)

- Os objetivos fundamentais dum sistema de Gestão de Memória dum Sistema Operativo incluem a realocização e partilha de memória. Explique porquê estas funcionalidades são necessárias e ou desejáveis.
- Segurança e Eficácia do OS
 - Data Execution Prevention (DEP) é uma funcionalidade de segurança incluída nos sistemas operativos Microsoft Windows. O seu objetivo é o de impedir a execução de instruções vindas de certas zonas do espaço de endereçamento dum processo. Explique como é que este mecanismo poderia ser implementado usando um sistema de memória paginada e quais são as zonas de memória que devem ser protegidas.
 - Explique o conceito de COW (Copy-on-Write) usando no Linux na criação dum novo processo usando fork(). Para que serve e como é que poderá ser implementado usando paginação ?
- Considere um sistema de gestão de memória com locação contígua de memória com partições múltiplas. Dadas partições de memória disponíveis de 50K, 300K, 200K e 100K (nesta ordem) como é que o SO colocará os processos com 270K, 20K, 80K, 40K, 120K (nesta ordem) usando os algoritmos de (i) First-Fit e (ii) Best-Fit.
- Um sistema de memória virtual tem um tamanho de página de 32 palavras, 8 páginas virtuais e 3 páginas físicas. Um endereço virtual neste sistema tem 8 bits, sendo que os primeiros 3 bits indicam o número de página. A tabela de páginas está inicialmente no estado apresentado em baixo:

Página virtual	Página física	Valido/Invalido
0	1	1
1	2	1
2		0
3		0
4		0
5		0
6		0
7	0	1

- 111 00010
- 000 00011
- 001 01100
- 010 10000
- 000 01111
- 011 01000
- 100 11100

- Indique se o resultado dos endereços lógicos indicados em cima e referenciados por esta ordem decrescente é uma page hit (sucesso), uma page miss (falha) ou uma trap (uma interrupção devido um erro). Se for uma page-miss (falha) será **invocado** o algoritmo de substituição de paginas (LRU-Least Recently Used / Menor usada recentemente) – quer dizer que a tabela de paginas poderá **mudar**!
 - Calcule o endereço físico resultante (exceto no caso dum trap) em valor decimal.
 - Mostre a tabela de páginas no fim desta sequência de endereços.
- Substituição de Páginas.
 - Num sistema de memória virtual paginada, quantas faltas de página aconteceriam usando o algoritmo de substituição de pagina "FIFO (First In First Out)" com a seguinte string de referência R:
R = 1, 2, 3, 4, 5, 1, 2, 3, 6, 1, 2, 3, 4, 5, 6
nos seguintes casos (i) quatro molduras (ii) cinco molduras ?
 - Este algoritmo com este R exhibe a anomalia de Belady ? (Nota: todas as molduras estão inicialmente vazias)

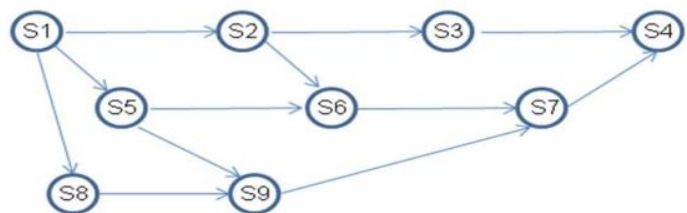
Grupo B: Threads, Concorrência e Sincronização (10 valores)

6. Explique os (três) Requisitos fundamentais para os algoritmos que fazem a manutenção da consistência de dados partilhados ?
7. Responda as perguntas seguintes considerando o seguinte programa em baixo. Nota que a instrução "x = x + k" onde "x" é uma variável inteira global e k um constante em *assembler* (x86 GNU) é a sequência de três instruções: `movl x, %eax ; addl $k, %eax ; movl %eax, x`

```
int x=0;
void *maisx(void *args) {
    int id = *(int*)args ;
    x=x+id;
}
main() {
    pthread_t th[3];
    int i,ids[3]={1,3,5};
    for (i=0; i<3; i++) pthread_create( &th[i],NULL, maisx, &ids[i]);
    for (i=0; i<3; i++) pthread_join( th[i], NULL );
    printf("x=%d\n",x);
}
```

- a) Quais são os outputs possíveis do programa ?
- b) Explique detalhadamente (faça uso dum *program trace*, pseudo-código, fluxograma etc) como é que o output poderá ser "x=6"
8. O grafo acíclico dirigido represente a execução dum programa onde as arestas representam uma regra de precedência. O código seguinte é executado no ordem especificado por duas threads concorrentes, A e B. Usando um numero mínimo de semáforos insere código para que a execução das threads A e B é consistente com o grafo. Deverá especificar os valores de inicialização dos semáforos e utilizar a Posix sintaxe.

Thread A	Thread B
S1	S2
S5	S3
S8	S6
S9	S4
S7	



9. a) Explique as condições necessárias (recursos e processos) para existir Deadlock.
- b) Considere a seguinte situação com 3 processos (P1,P2 e P3) e 3 recursos (R1,R2 e R3) - o recursos R2 tem apenas uma instância mas o R1 e R3 tem duas instâncias do mesmo recurso. O processo P1 está atribuído uma das instâncias do recurso R1 e uma do recurso R3 e pediu e está à espera do R2. O processo P2 está atribuído o recurso R2. O processo P2 pediu e está à espera da atribuição duma instância do R1. O processo P3 está atribuído uma instancia do recurso R3 e pediu e está a espera duma das instâncias do R1.
- i) Desenhe o grafo de atribuição/alocações de recursos para esta situação.
- ii) Existe uma situação de bloqueio mútua (*Deadlock*) entre os processos ou não? Justifique.