



### Prova Parcial 2

Nome: \_\_\_\_\_

- 1) (2,0 pontos) Marque verdadeiro ou falso. Justifique as falsas.
- a) (F) Sistemas em lote normalmente implementam o algoritmo de escalonamento Round-Robin por ter um bom equilíbrio entre eficiência e justiça. *Sistemas em lote normalmente usam algoritmos não-preemptivos, como FCFS e SJF. Algoritmos como Round-Robin são usados em sistema interativos.*
- b) (V) Um dos principais objetivos de algoritmos de escalonamento de sistemas operacionais interativos é a minimização do tempo de resposta médio.
- c) (F) O algoritmo de escalonamento de processos FCFS é justo e mantém as partes do sistema ocupadas, porém não faz distinção entre os processos. *O algoritmo FCFS é justo, porém se um processo é limitado por CPU e os demais por E/S, acontece o efeito “comboio”. Ou seja, ou os processos estão todos na fila da CPU ou na fila da E/S, prejudicando o equilíbrio.*
- d) (F) O algoritmo de substituição de página LRU (Least Recently Used = Menos Usado Recentemente) é o mais utilizado devido a sua eficiência e facilidade de implementação. *O algoritmo LRU é o mais eficiente, porém o mais difícil de ser implementado, pois precisa de hardware especializado.*
- e) (V) A memória virtual torna a tarefa do programador mais fácil, porque não precisa se preocupar com a quantidade de memória disponível.
- 2) (1,5 ponto) Qual o principal problema do algoritmo de escalonamento por prioridades? Qual seria uma solução para este problema?  
*O principal problema é a possibilidade de fazer um bloqueio indefinido (starvation) em um processo de baixa prioridade. Uma solução para este problema seria implementar um procedimento de envelhecimento. Aumenta-se gradualmente a prioridade de processos que esperam no sistema por muito tempo.*
- 3) (1,0 ponto) O que é fragmentação interna?  
*É o desperdício de espaço de memória dentro de uma página. Normalmente acontece apenas na última página do processo.*
- 4) (1,0 ponto) Qual a vantagem da paginação por demanda?  
*Apenas as páginas acessadas pelo processo são carregadas para a memória física. Isso permite mais eficiência, pois não gasta-se tempo carregando do disco páginas que possivelmente nunca serão usadas e permite uma maior quantidade de processos carregados na memória sem que ela torne-se cheia muito cedo.*

5) (1,5 pontos) Em um sistema operacional utiliza-se o algoritmo de escalonamento FCFS (*First Come First Served* = Primeiro a Chegar, Primeiro a ser Servido). Abaixo uma tabela ilustra o tempo de chegada dos processos ao estado de pronto em relação ao processo P1, em segundos. A tabela mostra também o tempo que necessitam de uso do processador, também em segundos. Os processos finalizam após o uso da CPU.

Processo	Tempo de chegada	Tempo de uso de CPU
P1	0	40
P2	10	80
P3	20	10
P4	25	20

a) Desenhe o Diagrama de Gantt, que ilustra a ordem e tempo dos processos em uso na CPU nesta rodada.

P1	P2	P3	P4	
0	40	120	130	150

b) Qual o tempo de retorno médio do algoritmo nesta situação?

*Tempos de retorno:*

$$P1 = 40 - 0 = 40$$

$$P2 = 120 - 10 = 110$$

$$P3 = 130 - 20 = 110$$

$$P4 = 150 - 25 = 125$$

$$\text{Tempo de retorno médio: } (40+110+110+125)/4 = 96,25 \text{ segundos}$$

6) (1,5 ponto) Um sistema possui 4 quadros de memória física, inicialmente não alocados e com SO implementando paginação por demanda. Nas tabelas abaixo, a primeira linha indica a sequência de páginas que os processos estão requisitando à memória. As 8 primeiras colunas mostram como as páginas vão sendo alocadas aos quadros e referenciadas. Entretanto, a partir na 9 coluna não há quadros suficientes para a página A. Continue o preenchimento das tabelas segundo os algoritmos mencionados, indicando na última linha se aconteceu um **page fault** naquela rodada.

Algoritmo de troca de páginas = FIFO (Primeiro a entrar, primeiro a sair)

Referência	E	D	E	H	B	D	E	D	A	E	B	E	D	E	B	G
Quadro 1	E	E	E	E	E	E	E	E	A	A	A	A	A	A	A	A
Quadro 2		D	D	D	D	D	D	D	D	E	E	E	E	E	E	E
Quadro 3				H	H	H	H	H	H	H	H	H	D	D	D	D
Quadro 4					B	B	B	B	B	B	B	B	B	B	B	G
Page Fault	X	X		X	X				X	X			X			X

Algoritmo de troca de páginas = LRU (Menos usada recentemente)

Referência	E	D	E	H	B	D	E	D	A	E	B	E	D	E	B	G
Quadro 1	E	E	E	E	E	E	E	E	E	E	E	E	E	E	E	E
Quadro 2		D	D	D	D	D	D	D	D	D	D	D	D	D	D	D
Quadro 3				H	H	H	H	H	A	A	A	A	A	A	A	G
Quadro 4					B	B	B	B	B	B	B	B	B	B	B	B
Page Fault	X	X		X	X				X							X

7) (1,5 ponto) A alocação por segmentação paginada é uma abordagem de gerenciamento que reuni as vantagens da segmentação com a paginação e está presente nos principais sistemas operacionais modernos. Dessa forma, os bits de um endereço lógico de um sistema desses são divididos em três grupos: os bits do segmento, os bits da página e os bits do deslocamento. Suponha que neste sistema, em particular, cada endereço lógico contém 18 bits, sendo que os 2 primeiros bits representam o segmento, os 5 bits seguintes representam a página e os bits restantes representam o deslocamento. Já o endereço físico contém 16 bits, sendo os 5 primeiros bits correspondentes ao número do quadro e os restantes correspondentes ao deslocamento. Calcule o endereço físico do endereço lógico 010000000000001101. (**Obs:** Anuncie caso o endereço não possa ser obtido por causa de uma *page fault*.)

Segmento 0 Código		Segmento 1 Dados		Segmento 2 Pilha	
Nº Página	Nº Quadro	Nº Página	Nº Quadro	Nº Página	Nº Quadro
0	Disco	0	9	0	Disco
1	7	1	Disco	1	Disco
2	Disco	2	3	2	2
3	0	3	Disco	3	Disco
4	Disco	4	8	4	Disco

*Os bits do endereço lógico podem ser divididos em:*

*Segmento: 01 → 1*

*Página: 00000 → 0*

*Deslocamento: 00000001101 → 13*

*No tabela do segmento 1, verificamos que o quadro correspondente a página 0 é a página 9. A solução poderia se dada no formato binário ou decimal.*

**Formato binário:**

*Une-se cinco bits do quadro (9 = 01001) com os bits do deslocamento (13 = 00000001101)*

*Ou seja, o endereço físico é 0100100000001101.*

**Formato decimal:**

*São 11 bits de deslocamento. Logo para calcular o endereço físico multiplica-se o número do quadro, pela quantidade de endereços por quadro e soma-se o deslocamento.*

*Ou seja,  $9 * 2^{11} + 13 = 18445$ .*

Boa prova e boa sorte!