

## Sistemas Operacionais - UCSAL

Professor : Marco Antônio C. Câmara – 3ª Lista de Exercícios

**1ª Questão :** Com base no seu conhecimento sobre os modelos de gerência de memória com partições fixas e variáveis, identifique pontos de vantagens e desvantagens de um em relação ao outro, quanto à performance, ocupação de memória, e tempo de resposta:

PARTIÇÕES FIXAS

PARTIÇÕES VARIÁVEIS

### VANTAGENS

*Simplicidade de implementação*      *Redução da Fragmentação Interna*

*Performance*

*Melhor utilização da memória*

### DESVANTAGENS

*Fragmentação Interna*

*Fragmentação Externa*

*Complexidade de implementação*

**2ª Questão :** Considere que um Sistema Operacional mantém uma lista de segmentos de memória ocupados e livres ordenada por endereço. Para alocar um novo processo, o algoritmo de gerenciamento de memória precisa pesquisar os segmentos livres. Para cada um dos métodos abaixo, identifique o algoritmo associado (First Fit, Worst Fit, Best Fit ou NRA):

**Worst Fit** : escolhe o segmento livre que tiver o maior tamanho disponível que seja superior ao necessário para carga do novo processo;

**Best Fit** : escolhe o segmento livre que tenha o menor tamanho disponível que seja superior ao necessário para carga do novo processo;

**First Fit** : escolhe o primeiro segmento livre que tenha tamanho suficiente para carga no novo processo;

**NRA** : sorteia, entre os segmentos livres com tamanho suficiente para carga do novo processo, qual será o escolhido.

**3ª Questão :** Para cada uma das afirmações abaixo, assinale (C)erto ou (E)rrado. No caso da opção estar errada, assinale também logo abaixo a(s) justificativa(s) para o erro. Se nenhuma das justificativas for adequada, escreva no último espaço a sua justificativa.

**Observação :** Nesta questão, só será considerada correta a questão marcada como (C)erta e realmente certa ou a questão (E)rrada com a respectiva justificativa marcada. Todos os outros casos serão considerados erros.

a) ( ~~E~~ ) Entre os métodos de alocação de memória, temos a alocação contígua, as partições fixas e variáveis, o *Swapping* e a Paginação. Desses, apenas o *Swapping* utiliza a unidade de memória secundária para armazenar parte do conteúdo que originalmente deveria estar na memória principal (RAM).

**Justificativa caso esteja errada:**

( ) Na verdade, TODOS os métodos utilizam o disco para armazenamento de parte da memória;

( ~~X~~ ) O método da Paginação também utiliza o disco para armazenamento;

( ) Apenas o método da paginação utiliza o disco para armazenamento.

( ) \_\_\_\_\_

## Sistemas Operacionais - UCSAL

Professor : Marco Antônio C. Câmara – 3ª Lista de Exercícios

- b) ( **C** ) Um dos desafios dos métodos que utilizam o disco para armazenar parte da memória é evitar ao máximo as operações de leitura e a gravação em disco, já que essas implicam em perdas significativas de desempenho.

**Justificativa caso esteja errada:**

- ( ) Na verdade, o objetivo é o oposto, já que a utilização massiva do disco permite liberar memória para atividades mais nobres, aumento a performance;
- ( ) Os métodos que utilizam o disco na verdade oferecem acessos mais rápidos do que os métodos convencionais baseados apenas em memória;
- ( ) Graças aos avançados algoritmos de acesso ao disco, não ocorre nenhuma perda de performance desses métodos em relação aos métodos convencionais. Sendo assim, a performance é a mesma, independente do método.
- ( ) \_\_\_\_\_

- c) ( **C** ) No gerenciamento de memória, um endereço lógico só é equivalente ao físico quando a aplicação é carregada na memória a partir do endereço inicial.

**Justificativa caso esteja errada:**

- ( ) Um endereço lógico não pode ser igual ao físico, já que precisa ser convertido durante o processo de carga da aplicação.
- ( ) Um endereço lógico sempre se equivale ao físico após a carga da aplicação na memória do computador.
- ( ) Na verdade, os endereços ficam iguais quando o processo é carregado nas últimas posições de memória.
- ( ) \_\_\_\_\_

**4ª Questão :** Para cada uma das afirmações abaixo, assinale (C)erto ou (E)rrado. No caso da opção estar errada, assinale também logo abaixo a(s) justificativa(s) para o erro. Se nenhuma das justificativas for adequada, escreva no último espaço a sua justificativa.

**Observação :** Nesta questão, só será considerada correta a questão marcada como (C)erta e realmente certa ou a questão (E)rrada com a respectiva justificativa marcada. Todos os outros casos serão considerados erros.

- a) ( **E** ) Entre os métodos de alocação de memória, temos a alocação contígua, as partições fixas e variáveis, o *Swapping* e a Paginação. Desses, apenas o *Swapping* é adequado a sistemas operacionais multitarefa. Todos os demais são adequados apenas para ambientes monotarefa.

**Justificativa caso esteja errada:**

- ( ) Na verdade, TODOS os métodos são adequados para sistemas multitarefa;
- ( **X** ) O único utilizado apenas para ambientes monotarefa é o de alocação contígua;
- ( ) Os métodos de alocação contígua e de partições são utilizados apenas para ambientes multitarefa.
- ( ) \_\_\_\_\_

- b) ( **E** ) O único objetivo dos métodos que utilizam a memória secundária para simular uma memória maior no dispositivo é permitir a execução de um número maior de múltiplos processos simultâneos.

**Justificativa caso esteja errada:**

- ( ) Na verdade, devido à perda de desempenho, sistemas que utilizam memória virtual na verdade reduzem a capacidade de execução simultânea de múltiplos processos;
- ( **X** ) O uso do recurso da estratégia de memória virtual também permite a execução de processos maiores do que o espaço disponível de memória real;
- ( ) Graças aos avançados algoritmos de acesso à memória secundária, não ocorre nenhuma perda de performance desses métodos em relação ao acesso convencional à memória principal. Sendo assim, a única necessidade de memória real é para garantir a execução dos processos, já que apenas processos armazenados na memória principal podem entrar em execução.
- ( ) \_\_\_\_\_

## Sistemas Operacionais - UCSAL

Professor : Marco Antônio C. Câmara – 3ª Lista de Exercícios

c) ( ~~E~~ ) Páginas e segmentos são partes, respectivamente, da memória principal, e do espaço de armazenamento em memória secundária.

Justificativa caso esteja errada:

- ( ) Na verdade é o contrário. Páginas são partes do espaço de armazenamento em memória secundária, e segmentos são partes da memória principal;
  - ( ) Tanto as páginas quanto os segmentos são partes do espaço de armazenamento em memória secundária;
  - ( ~~X~~ ) Tanto as páginas quanto os segmentos podem estar na memória principal ou secundária;
  - ( ) \_\_\_\_\_
- 

**5ª Questão :** Por quê os algoritmos de memória virtual muitas vezes precisam fazer dois acessos à unidade de memória secundária para recuperar uma determinada informação? Como isso é contornado em alguns sistemas operacionais?

*No gerenciamento de memória virtual, o acesso à memória secundária acontece apenas quando o conteúdo desejado não se encontra armazenado na memória principal. No caso da paginação, isto pode ocorrer tanto com a tabela de páginas (que nem sempre consegue estar completamente armazenada em memória) quanto com o conteúdo desejado, que pode ter sido transferido para a unidade secundária (page-out).*

*Quando o SO solicita o acesso a um determinado conteúdo, a primeira busca ocorre na tabela de páginas, e se aquela posição específica da tabela não estiver na memória principal, acontece aí o primeiro acesso à memória secundária. O segundo acesso pode acontecer quando é feito o acesso à página apontada na tabela, caso ela também não esteja na memória principal.*

*Para contornar estes acessos duplicados, alguns SOs utilizam algoritmos preditivos similares aos utilizados pela memória cache, tentando prever o acesso a partes da tabela de páginas e páginas específicas. Outros utilizam uma espécie de buffer chamado de TLB (Translation Lookaside Buffer), ou usam estratégias para reduzir o tamanho da tabela de páginas, evitando que partes da mesma sofram page-out.*

**6ª Questão :** Identifique pontualmente os problemas relacionados à fragmentação de memória e as estratégias para evitá-la:

*A gerência de memória precisa lidar com dois tipos de problema relacionados à fragmentação. Um deles é a fragmentação interna, e o*

outro a fragmentação externa. Na interna, nem todo o espaço disponível nos quadros de memória pode ser utilizado; na externa, quadros de memória ficam desocupados por não terem tamanho suficiente para abrigar os processos que precisam ser alocados.

São diversas as estratégias para evitar estes problemas. A fragmentação externa normalmente é resolvida através do uso de quadros de memória pequenos e de tamanho fixo, associados às técnicas de alocação dinâmica e fragmentada dos espaços (paginação). Já a fragmentação interna é inevitável, mas o uso de quadros menores acaba reduzindo a perda de memória causada por este problema.

**7ª Questão :** Nos modelos de gerência de memória baseados no uso de Memória Virtual, por que os processos só podem referenciar endereços virtuais?

O uso de endereços "virtuais" permite, na prática, que cada endereço virtual seja apontado para qualquer ponto na memória física, ou na memória secundária, permitindo um melhor uso da memória disponível, admitindo processos maiores do que a memória física disponível, e oferecendo a segurança e o isolamento necessário entre diferentes processos.

**8ª Questão :** Nos modelos de gerência de memória baseados no uso de Memória Virtual, como o registrador de endereços interfere no tamanho do Espaço de Endereçamento Virtual?

O registrador de endereços determina o tamanho total do espaço de endereçamento virtual. Um registrador de 64 bits, por exemplo, determina o armazenamento de até 16 Exabytes =  $2^{64}$  endereços.

( 41 ) **9ª Questão :** Um dos problemas comuns aos primeiros métodos de alocação de memória é a fragmentação externa, provocada pela variedade de tamanho dos processos a serem alocados nos blocos de memória disponíveis. A Paginação de Memória trata desse problema implementando algumas técnicas específicas. Escreva ao lado a soma das afirmativas corretas. (lembre-se que, cada afirmativa incorreta marcada como correta anula uma marcada corretamente. Logo, evite chutar)

- (01) A memória física é dividida em blocos de tamanho fixo chamados de quadros, enquanto que a memória lógica é dividida em blocos do mesmo tamanho chamados de páginas;
- (02) A compactação da memória, juntando blocos livres de memória em blocos maiores e contíguos permite reduzir o problema da fragmentação;
- (04) Os endereços gerados pela CPU são divididos em duas partes: o número da página ( $p$ ) e o deslocamento de página ( $d$ ). O número do deslocamento de página é utilizado como índice em uma tabela de páginas, que contém o endereço base do quadro correspondente na memória física;
- (08) O tamanho de cada página (que é igual ao tamanho do quadro) é definido com base em uma potência de 2 com o tamanho variável de acordo com a arquitetura do dispositivo computacional;
- (16) A eliminação da fragmentação promovida pela Paginação de Memória é válida para qualquer processo. Na prática, não ocorre mais perda de espaço de memória utilizando esse método;
- (32) Se o tamanho de uma página for  $2^m$ , e o número de quadros disponíveis na memória física for  $2^n$ , então a memória física total é de  $2^{m+n}$ .

## Sistemas Operacionais - UCSAL

Professor : Marco Antônio C. Câmara – 3ª Lista de Exercícios

**10ª Questão :** Para cada um dos impactos relacionados abaixo, identifique se ele ocorre para Páginas de Memória (G)randes, (P)equenas, ou marque a letra “X”, se não houver qualquer influência relacionada ao tamanho da Página:

- (P) Aumento do tamanho da Tabela de Páginas;
- (G) Aumento da eficiência nas operações de disco;
- (P) Redução das perdas por fragmentação interna;
- (X) A fragmentação total tende a ser igual ao tamanho de meia página multiplicado pelo número de processos em execução;
- (X) A fragmentação interna, se ocorrer, se concentra no último quadro;

(76) **11ª Questão :** Com base na figura, que representa um esquema de paginação de memória em uma suposta máquina com palavras de 32 bits, escreva ao lado a soma das afirmativas corretas. (lembre-se que, cada afirmativa incorreta marcada como correta anula uma marcada corretamente. Logo, evite chutar)

Memória Lógica	
0	a
1	b
2	c
3	d
4	e
5	f
6	g
7	h
8	i
9	j
10	k
11	l
12	m
13	n
14	o
15	p

Tabela de Página	
0	
1	
2	
3	

Memória Física	
0	i
	j
	k
	l
4	
8	
12	m
	n
	o
	p
16	a
	b
	c
	d
20	
24	
28	e
	f
	g
	h

- (01) A Tabela de Página está incompleta, pois seria necessário representar no mínimo 8 linhas, para corresponder ao número de quadros da memória física;
- (02) A representação da Memória Física está errada, pois não deveria estar ocorrendo fragmentação, já que a mesma é eliminada no esquema de paginação de memória;
- ~~(04)~~ Considerando o esquema representado, a Tabela de Página deveria estar preenchida, nas suas duas primeiras posições, com os números 4 e 7;
- ~~(08)~~ A soma dos números na Tabela de Página, se corretamente preenchidos, deve ser igual a 14;
- (16) Considerando-se o armazenamento de todas as tabelas na Memória Principal, o esquema representado obrigaria o acesso triplo à memória para recuperação de determinado conteúdo (uma vez para cada tabela);
- (32) Devido à fragmentação representada na figura, o maior processo que poderia ser instalado teria no máximo 32 Bytes;
- ~~(64)~~ O esquema representado é compatível com um ambiente de 128 Bytes de memória, e páginas de 16 Bytes;

(13) **12ª Questão :** Sobre o controle que o SO precisa manter sobre as páginas e quadros existentes, e sua disponibilidade para alocação, escreva ao lado a soma das afirmativas corretas. (lembre-se que, cada afirmativa incorreta marcada como correta anula uma marcada corretamente. Logo, evite chutar)

- ~~(04)~~ Além da Tabela de Páginas, o SO mantém uma “Tabela de Quadros”, utilizada para identificar quadros disponíveis e alocados, e no caso do mesmo estar alocado, a que página de que processo ele está associado;
- (02) Tipicamente cada processo possui sua própria Tabela de Quadros. Por outro lado, só existe uma Tabela de Páginas;
- ~~(04)~~ Com o bloqueio de um processo, o Sistema Operacional precisa guardar uma cópia da Tabela de Página no BCP, que é a mesma estrutura onde se guardam informações como o PC (Program Counter) e estruturas de pilha do processo que foi retirado de execução;
- ~~(08)~~ Com a paginação, há uma clara separação entre a visão da memória a partir do processo, e a memória física;
- (16) Como o número de páginas tipicamente é menor do que o número de blocos, é necessário implementar um mapeamento detalhado de suas associações.

## Sistemas Operacionais - UCSAL

Professor : Marco Antônio C. Câmara – 3ª Lista de Exercícios

**13ª Questão:** Para cada uma das afirmativas abaixo, assinale (C)erto ou (E)rrado. No caso da afirmativa estar errada, assinale também logo abaixo a(s) justificativa(s) para o erro. Se nenhuma das justificativas for adequada, escreva no último espaço a sua justificativa.

**Observação :** Nesta questão, só será considerada correta a questão marcada como (C)erta e realmente certa ou a questão (E)rrada com a respectiva justificativa marcada. Todos os outros casos serão considerados erros.

- a)  Uma das formas mais simples de implementar a Tabela de Páginas em *hardware*, embora não muito utilizada, é o armazenamento da mesma em registros específicos da CPU. Muito embora o armazenamento em registros dedicados de alta performance seja importante para acelerar o acesso à memória, por outro lado essa estratégia normalmente limita muito o tamanho máximo da tabela.

**Justificativa caso esteja errada:**

- Na verdade, esse é o único método de implementação adotado pelos Sistemas Operacionais modernos;
- Ao contrário do que foi afirmado, o armazenamento em registros dedicados torna a operação muito lenta, o que na verdade é o motivo dessa estratégia não ser muito utilizada;
- Como a Tabela de Páginas aponta para quadros da memória física, na prática ela nunca é muito grande. Por isso a estratégia não limita o tamanho máximo da tabela.
- \_\_\_\_\_

- b)  Em alguns Sistemas Operacionais, a troca da Tabela de Páginas ocorre com base no ponteiro PTBR (*Page-Table Base Register*). Ele indica qual o setor do disco que contém a Tabela a ser carregada. Ao ser escalonado um processo, a leitura do PTBR determina a carga da Tabela na Memória Física, o que acaba aumentando o tempo envolvido na troca de contexto do escalonador.

**Justificativa caso esteja errada:**

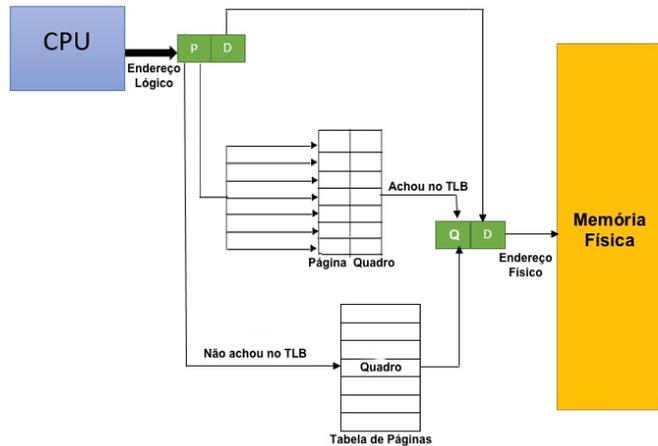
- A Tabela de Páginas contém as informações relativas a todos os processos já carregados, logo não há necessidade de carga de nenhuma tabela durante as trocas de contexto;
- O PTBR aponta para a posição da Memória onde está armazenada a Tabela de Página do processo que foi escalonado, reduzindo o tempo envolvido na troca de contexto;
- Na verdade, a Tabela de Páginas permanece sempre armazenada em disco, já que seu acesso é eventual, não implicando em nenhuma perda de performance.
- \_\_\_\_\_

- c)  A depender da implementação da Tabela de Páginas, podem ser necessários múltiplos acessos à memória para recuperar um simples Byte. Isso provoca um atraso considerável, que não pode ser tolerada em boa parte das aplicações. Uma das soluções adotadas para esse problema é a implementação de uma estrutura intermediária, chamada de TLB (*Translation Look-Aside Buffers*) constituída de uma área de disco especialmente reservada em praticamente todos os sistemas operacionais.

**Justificativa caso esteja errada:**

- Muito embora essa seja realmente uma das soluções adotadas para o problema do acesso múltiplo à memória, ela só é adotada em alguns poucos Sistemas Operacionais modernos extremamente ágeis;
- Não há qualquer perda de performance associada às múltiplas leituras na memória;
- A estrutura intermediária está armazenada em uma área específica de memória, e não do disco.
- \_\_\_\_\_

**14ª Questão :** Dada a figura, que esquematiza o acesso à memória paginada com uso do TLB (*Translation Look-Aside Buffers*), avalie as afirmativas abaixo:



- I. O TLB armazenada apenas uma parte da Tabela de Páginas, visando aumentar a performance da busca.
- II. A TLB mantém os itens da Tabela de Páginas mais acessados.
- III. O tempo efetivo de acesso à memória é calculado com base na probabilidade de localização da informação na TLB.

- (a) Apenas a segunda afirmativa está correta;
- (b) As duas primeiras afirmativas estão corretas;
- ~~(c)~~ Todas as afirmativas estão corretas;
- (d) Nenhuma das afirmativas está correta.

**15ª Questão:** Considerando um tempo de acesso típico à DRAM (Memória RAM dinâmica) de 60ns, e um tempo de acesso ao TLB de 5ns, pode-se dizer que o tempo efetivo de acesso à memória, considerado uma probabilidade de “TLB Hit” igual a 95% seria:

- (a) Independente da probabilidade de “TLB Hit”, seria de 120ns;
- (b) 65ns;
- (c) 57ns;
- ~~(d)~~ 67,75ns;  $0,95 * (5 + 60) + 0,05 * (60 + 60) = 67,75ns$
- (e) 7,75ns;
- (f) N.R.A.;

**( 05 ) 16ª Questão:** Com base no seu conhecimento sobre a proteção de memória no método de Paginação de Memória, escreva ao lado a soma das afirmativas corretas. (lembre-se que, cada afirmativa incorreta marcada como correta anula uma marcada corretamente. Logo, evite chutar)

- ~~(01)~~ A proteção de memória é implementada a bits que atribuem direitos específicos, e que estão associados a cada quadro da memória física;
- (02) As tentativas de escrita sobre um bloco que está marcado com um bit de “apenas leitura” deve ser interpretada pelo programador da aplicação, que deverá tratar o erro dentro de sua aplicação;
- ~~(04)~~ O esquema de proteção geralmente inclui também um bit de “válido/inválido”, que permite limitar o número as páginas que um processo pode ter acesso;
- (08) O esquema de proteção de memória do método de Paginação possui um algoritmo específico para o controle de acesso de um processo às páginas de outros referenciadas na Tabela de Páginas comum entre eles;
- (16) O armazenamento dos bits de proteção de memória na Tabela de Páginas tem como objetivo garantir o mesmo grau de proteção também às Páginas do Processo.

## Sistemas Operacionais - UCSAL

Professor : Marco Antônio C. Câmara – 3ª Lista de Exercícios

**17ª Questão:** Qual das afirmativas abaixo acerca de Paginação Multinível está correta?

- a) ( ) Os esquemas de Paginação Multinível resolvem o problema do tempo de acesso, já que eliminam os múltiplos acessos à memória típicos em esquemas de Paginação com apenas um nível;
- b) ( ) A Paginação Multinível estabelece uma Tabela de Páginas com níveis diferenciados de acesso para processos internos do Sistema Operacional e Aplicações de Usuários;
- c) ( ) Os esquemas de Paginação Multinível sempre trabalham com dois níveis de Paginação;
- d) (X) A Paginação Multinível permite o armazenamento seletivo de partes da Tabela de Páginas na memória física, devido ao tamanho excessivo da Tabela completa;
- e) ( ) N.R.A.;

**18ª Questão:** Para cada uma das afirmativas abaixo, identifique a técnica associada com a letra correspondente: Tabela de Página (I)nvertida, Páginas (C)ompartilhadas, (S)egmentação. Use a letra (X) caso a afirmativa não esteja relacionada a nenhuma das técnicas apresentadas.

- ( I ) Permite reduzir o tamanho da Tabela de Paginação, identificando apenas os endereços dos quadros de memória e respectivas páginas alocadas e processos associados;
- ( X ) Permite o compartilhamento da memória principal entre dispositivos computacionais conectados em rede;
- ( S ) Estende o conceito de paginação, permitindo que o espaço de endereçamento lógico seja dividido em uma coleção de segmentos, o que lembra a visão que um usuário típico tem da memória;
- ( S ) Permite a extensão do conceito de proteção de memória, já que permite, por exemplo, atribuir especificamente a um trecho de memória de um processo que contém código executável o atributo de “apenas leitura”, impedindo a modificação indesejada do código;
- ( C ) Habilita o reaproveitamento de segmentos de código entre diferentes processos ou múltiplas instâncias do mesmo processo;
- ( X ) Habilita a busca mais rápida de uma determinada página, por oferecer dois roteiros de busca: o normal e o invertido;