



Universidade Federal Rural de Pernambuco
Sistemas de Informação

Princípios de Software Básico

Lhaíslla Eduarda Cavalcanti Rodrigues da Silva

RECIFE-PE

Exercícios referente ao capítulo 3- Gerenciamento de memória do livro Sistemas Operacionais modernos Andrew . S Tanenbaum

1. O IBM 360 tem um esquema de travar blocos de 2 KB designando a cada um uma chave de 4 bits e fazendo a CPU comparar a chave em cada referência de memória a uma chave de 4 bits no PSW. Cite dois problemas desse esquema não mencionados no texto.

Dividir a memória em blocos e colocar bits de verificação para cada bloco, assim pode-se saber a quem a memória pertence na hora da alocação. outra ideia é manter dois registradores com endereço base de memória do processo e um outro com o endereço limite. A cada referência a um endereço os registradores são verificados

2. Na Figura 3.3 os registradores base e limite contêm o mesmo valor, 16.384. Isso é apenas um acidente, ou eles são sempre os mesmos? Se for apenas um acidente, por que eles estão no mesmo exemplo?

São os mesmos, ele tinha múltiplos registradores base, permitindo programar textos e dados, por exemplo, para serem realocados independentemente, mas não oferecia proteção contra referências à memória além da capacidade.

3. Um sistema de troca elimina lacunas por compactação. Presumindo uma distribuição aleatória de muitas lacunas e muitos segmentos de dados e um tempo para ler ou escrever uma palavra de memória de 32 bits de 4 ns, aproximadamente quanto tempo leva para compactar 4 GB? Para simplificar, presume que a palavra 0 faz parte de uma lacuna e que a palavra mais alta na memória contém dados válidos.

Quando as trocas de processos criam múltiplos espaços na memória, é possível combiná-los em um grande espaço movendo todos os processos para baixo, o máximo possível. Essa técnica é conhecida como compactação de memória. Em geral ela não é feita porque exige muito tempo da CPU. ela levaria em torno de 4 s para compactar toda a memória.

4. Considere um sistema de troca no qual a memória consiste nos seguintes tamanhos de lacunas na ordem da memória: 10 MB, 4 MB, 20 MB, 18 MB, 7 MB, 9 MB, 12 MB e 15 MB. Qual lacuna é pega para sucessivas solicitações de segmentos de

- (a) 12 MB
- (b) 10 MB
- (c) 9 MB

para o algoritmo primeiro encaixe? Agora repita a questão para melhor encaixe, pior encaixe e próximo encaixe.

Processo A: 20 KB, 12 KB e 18 KB - Processo B: 18 KB, 9 KB e 20 KB.
Alternativas a e b são as corretas.

5. Qual é a diferença entre um endereço físico e um endereço virtual?

Memória virtual é uma técnica que usa a memória secundária como uma cache para armazenamento secundário e endereço físico é a posição da célula na memória. Endereços são indexadores pelos quais os programas podem referenciar dados na memória.

6. Para cada um dos endereços virtuais decimais seguintes, calcule o número da página virtual e deslocamento para uma página de 4 KB e uma de 8 KB: 20.000, 32.768, 60.000.

Em um sistema de memória virtual baseada em paginação, o endereço virtual apresenta o formato (p, d), onde p = número da página e d = deslocamento dentro da página. A MMU (Memory Management Unit) deve, portanto, transformar primeiramente o endereço virtual “linear” 123456 para este formato (no exemplo da questão, o sistema possui 58 páginas virtuais ($232K/4k = 58$) e cada endereço virtual possui 18 bits, assim divididos: 6 bits para armazenar o número p da página e 12 bits para armazenar o deslocamento d). Para se determinar os valores de página e deslocamento, o endereço virtual 123456 é dividido pelo tamanho da página, que é de 4096 bytes. O resultado desta divisão implica em: página = 30, deslocamento = 576. De posse do número da página, a tabela de páginas do processo é então acessada para se descobrir se esta página está mapeada na memória. Se estiver, o número da moldura correspondente é usado para, juntamente com o deslocamento, calcular o endereço físico a ser acessado (endereço físico = número da moldura * tamanho da moldura + deslocamento). O tamanho da moldura é, obviamente, igual ao tamanho da página. Caso a página acessada não esteja na memória (o que pode ser deduzido observando-se o valor do bit de validade/presença da entrada na tabela de páginas), é gerada uma exceção (page fault – falta de página) e um algoritmo de substituição de páginas é executado para se determinar qual das páginas mapeadas na memória deve ser substituída para que a página recém-acessada seja carregada na moldura em que se encontra a página selecionada para expulsão. A tabela de páginas é atualizada. Por exemplo, supondo que a página 30 esteja mapeada na moldura 10 da memória real, a posição física correspondente seria assim calculada: $10 \times 4096 + 576$ (moldura x tamanho da moldura + deslocamento), resultando em acesso ao endereço físico 41.536. Caso o hardware possua um sistema de cache de páginas implementado via TLB - Translation Look Aside Buffer, a página a ser acessada é verificada antes se está na TLB. Caso não esteja mapeada, somente depois é examinada a tabela de páginas do processo na memória. O endereço da tabela de páginas na memória faz parte do contexto do processo.

7. Usando a tabela de páginas da Figura 3.9, dê o endereço físico correspondendo a cada um dos endereços virtuais a seguir:
- (a) 20
 - (b) 4.100
 - (c) 8.300

Cada página começa com um múltiplo de 4096 e termina 4095 endereços acima; assim, 4K a 8K na verdade significa 4096-8191 e 8K a 12K significa 8192-12287. A MMU observa que a página não está mapeada (o que é indicado por um X na figura) e faz a CPU desviar para o sistema operacional. Essa interrupção é chamada de falta de página (page fault). O sistema operacional escolhe um quadro de página pouco usado e escreve seu conteúdo de volta para o disco (se já não estiver ali). Ele então carrega (também do

disco) a página recém-referenciada no quadro de página recém-liberado, muda o mapa e reinicia a instrução que causou a interrupção.

8. O processador 8086 da Intel não tinha uma MMU ou suporte para memória virtual. Mesmo assim, algumas empresas venderam sistemas que continham uma CPU 8086 inalterada e que realizava paginação. Dê um palpite informal sobre como eles conseguiram isso. (Dica: pense sobre a localização lógica da MMU.)
Por causa do mapeamento feito pela MMU, CPU onde a presença da cache e da MMU pode ter um impacto importante sobre o desempenho, possibilitando o chaveamento de um programa para outro.
9. Que tipo de suporte de hardware é necessário para uma memória virtual paginada funcionar?
Utiliza um pedaço do HD. Com isso, ela transfere partes da memória RAM para o disco rígido, viabilizando o acesso a mais memória RAM para a execução de programas.
10. Copiar-na-escrita é uma ideia interessante usada em sistemas de servidores. Faz algum sentido em um smartphone?
Sim, da maneira que o dado é copiado somente quando absolutamente necessário.
11. Considere o programa C a seguir:

```
int X[N];  
int step = M; /* M é uma constante pré definida */  
for (int i = 0; i < N; i += step) X[i] = X[i] + 1;
```

(a) Se esse programa for executado em uma máquina com um tamanho de página de 4 KB e uma TLB de 64 entradas, quais valores de M e N causarão uma falha de TLB para cada execução do laço interno?
Não, pois ocorreria a troca de processos.

(b) Sua resposta na parte (a) seria diferente se o laço fosse repetido muitas vezes? Explique Sim, pois causa aumento de tempo e consumo de memória.
12. O montante de espaço de disco que deve estar disponível para armazenamento de páginas está relacionado ao número máximo de processos, n , o número de bytes no espaço de endereçamento virtual, v , e o número de bytes de RAM, r . Dê uma expressão para o pior caso de exigências de disco-espaco. Quão realista é esse montante?
 $(v \times r)^n$ a virtualização cresce em função do número de processos, se os processos tiverem um crescimento exponencial, a virtualização terá um crescimento exponencial.
13. Se uma instrução leva 1 ns e uma falta de página leva n ns adicionais, dê uma fórmula para o tempo de instrução efetivo se a falta de página ocorrer a cada k instruções.
 $(p)^{n-1}$
14. Uma máquina tem um espaço de endereçamento de 32 bits e uma página de 8 KB. A tabela de páginas é inteiramente em hardware, com uma palavra de 32 bits por

entrada. Quando um processo inicializa, a tabela de páginas é copiada para o hardware da memória, a uma palavra a cada 100 ns. Se cada processo for executado por 100 ms (incluindo o tempo para carregar a tabela de páginas), qual fração do tempo da CPU será devotado ao carregamento das tabelas de páginas?

$$(32/8) / (100) = 0.04 \text{ ms}$$

15. Suponha que uma máquina tenha endereços virtuais de 48 bits e endereços físicos de 32 bits.

(a) Se as páginas têm 4 KB, quantas entradas há na tabela de páginas se ela tem apenas um único nível? Explique. 2^{48-n} , permite definir o grau da multiprogramação

(b) Suponha que esse mesmo sistema tenha uma TLB (Translation Lookaside Buffer) com 32 entradas. Além disso, suponha que um programa contenha instruções que se encaixam em uma página e leia sequencialmente elementos inteiros, longos, de um conjunto que compreende milhares de páginas. Quão efetivo será o TLB para esse caso? O TLB consiste em um pequeno número de entradas, oito neste exemplo, mas raramente mais do que 256. Cada entrada contém informações sobre uma página, incluindo o número da página virtual, um bit que é configurado quando a página é modificada, o código de proteção (ler/escrever/permissões de execução) e o quadro de página física na qual a página está localizada. Esses campos têm uma correspondência de um para um com os campos na tabela de páginas, exceto pelo número da página virtual, que não é necessário na tabela de páginas

16. Você recebeu os seguintes dados a respeito de um sistema de memória virtual:

(a) A TLB pode conter 1024 entradas e pode ser acessada em 1 ciclo de relógio (1 ns).

(b) Uma entrada de tabela de página pode ser encontrada em 100 ciclos de relógio ou 100 ns.

(c) O tempo de substituição de página médio é 6 ms. Se as referências de página são manuseadas pela TLB 99% das vezes e apenas 0,01% leva a uma falta de página, qual é o tempo de tradução de endereço eficiente?

$$28 - 32 \text{ k}, \text{ em um tempo médio de } 6 \text{ ms}$$

17. Suponha que uma máquina tenha endereços virtuais de 38 bits e endereços físicos de 32 bits.

(a) Qual é a principal vantagem de uma tabela de páginas em múltiplos níveis sobre uma página de um único nível?

O propósito da tabela de páginas é mapear as páginas virtuais em quadros de páginas. As vantagens desse método são que ele é direto e não exige referências de memória durante o mapeamento.

(b) Com uma tabela de página de dois níveis, páginas de 16 KB e entradas de 4 bytes, quantos bits devem ser alocados para o campo da tabela de páginas de alto nível e quantas para o campo de tabela de páginas para o nível seguinte? Explique. 32 bits em 4 níveis.

18. A Seção 3.3.4 declara que o Pentium Pro ampliou cada entrada na hierarquia da tabela de páginas a 64 bits, mas ainda assim só conseguia endereçar 4 GB de memória.

Explique como essa declaração pode ser verdadeira quando as entradas da tabela de páginas têm 64 bits

O segredo para o uso do método da tabela de páginas multinível é evitar manter todas as tabelas de páginas na memória o tempo inteiro. Em particular, aquelas que não são necessárias não devem ser mantidas. Suponha, por exemplo, que um processo precise de 12 megabytes: os 4 megabytes da base da memória para o código do programa, os próximos 4 megabytes para os dados e os 4 megabytes do topo da memória para a pilha. Entre o topo dos dados e a parte de baixo da pilha há um espaço gigante que não é usado.

19. Um computador com um endereço de 32 bits usa uma tabela de páginas de dois níveis. Endereços virtuais são divididos em um campo de tabela de páginas de alto nível de 9 bits, um campo de tabela de páginas de segundo nível de 11 bits e um deslocamento. Qual o tamanho das páginas e quantas existem no espaço de endereçamento?

$$2^{11} * 2^{11} = 2^{22} = 4194304$$

20. Um computador tem endereços virtuais de 32 bits e páginas de 4 KB. O programa e os dados juntos encaixam-se na página mais baixa (0–4095). A pilha encaixa-se na página mais alta. Quantas entradas são necessárias na tabela de páginas se a paginação tradicional (de um nível) for usada? Quantas entradas da tabela de páginas são necessárias para a paginação de dois níveis, com 10 bits em cada parte?

11, 11 e 22 entradas

21. A seguir há um traço de execução de um fragmento de um programa para um computador com páginas de 512 bytes. O programa está localizado no endereço 1020, e seu ponteiro de pilha está em 8192 (a pilha cresce na direção de 0). Dê a sequência de referências de página geradas por esse programa. Cada instrução ocupa 4 bytes (1 palavra) incluindo constantes imediatas. Ambas as referências de instrução e de dados contam na sequência de referências. Carregue palavra 6144 no registrador 0. Envie registrador 0 para pilha. Chame uma rotina em 5120, empilhando o endereço de retorno. Subtraia a constante imediata 16 do ponteiro de pilha. Compare o parâmetro real com a constante imediata 4. Salte se igual a 5152.

- a. (a) Cada referência será falha de página, a menos que o número de quadros da página é de 512, o comprimento da sequência inteira.

22. Um computador cujos processos têm 1024 páginas em seus espaços de endereços mantém suas tabelas de páginas na memória. O custo extra exigido para ler uma palavra da tabela de páginas é 5 ns. Para reduzir esse custo extra, o computador tem uma TLB, que contém 32 pares (página virtual, quadro de página física), e pode fazer uma pesquisa em 1 ns. Qual frequência é necessária para reduzir o custo extra médio para 2 ns?

Os quadros de página para FIFO são os seguintes:

x0172333300

xx017222233

xxx01777722

xxxx0111177

Os quadros de página para LRU são os seguintes:

x0172327103

xx017232710

xxx01773271

xxxx0111327

FIFO rende seis falhas de página; LRU rende sete.

23. Como um dispositivo de memória associativa necessário para uma TLB pode ser implementado em hardware, e quais são as implicações de um projeto desses para sua capacidade de expansão?
- Quando um endereço virtual é apresentado para a MMU para tradução, o hardware primeiro confere para ver se o seu número de página virtual está presente na TLB comparando-o com todas as entradas simultaneamente (isto é, em paralelo). É necessário um hardware especial para realizar isso, que todas as MMUs com TLBs têm. Se uma correspondência válida é encontrada e o acesso não viola os bits de proteção, o quadro da página é tirado diretamente da TLB, sem ir à tabela de páginas. Se o número da página virtual estiver presente na TLB, mas a instrução estiver tentando escrever em uma página somente de leitura, uma falha de proteção é gerada. O interessante é o que acontece quando o número da página virtual não está na TLB. A MMU detecta a ausência e realiza uma busca na tabela de páginas comum. Ela então destitui uma das entradas da TLB e a substitui pela entrada de tabela de páginas que acabou de ser buscada. Portanto, se a mesma página é usada novamente em seguida, da segunda vez ela resultará em uma presença de página em vez de uma ausência. Quando uma entrada é retirada da TLB, o bit modificado é copiado de volta na entrada correspondente da tabela de páginas na memória. Os outros valores já estão ali, exceto o bit de referência. Quando a TLB é carregada da tabela de páginas, todos os campos são trazidos da memória.
24. Uma máquina tem endereços virtuais de 48 bits e endereços físicos de 32 bits. As páginas têm 8 KB. Quantas entradas são necessárias para uma tabela de páginas linear de um único nível?
- A página pode estar na realidade na memória, mas não na tabela de páginas do processo. Por exemplo, a página pode ter sido trazida do disco por outro processo. Nesse caso, não precisamos acessar o disco novamente, mas basta mapear a página de maneira apropriada nas tabelas de páginas. Essa é uma ausência bastante leve chamada falta de página menor (minor page fault). Segundo, uma falta de página maior (major page fault) ocorre se ela precisar ser trazida do disco. Terceiro, é possível que o programa apenas tenha acessado um endereço inválido e nenhum mapeamento precisa ser acrescentado à TLB. Nesse caso, o sistema operacional tipicamente mata o programa com uma falta de segmentação.
25. Um computador com uma página de 8 KB, uma memória principal de 256 KB e um espaço de endereçamento virtual de 64 GB usa uma tabela de página invertida para implementar sua memória virtual. Qual tamanho deve ter a tabela de dispersão para assegurar um comprimento médio da lista encadeada por entrada da tabela menor que 1? Presuma que o tamanho da tabela de dispersão seja uma potência de dois.
- A sequência de 0, 1, 2, 1, 2, 0, 3. Em LRU, página 1 será substituída pela página 3. No relógio, página 1 será substituído, uma vez que todas as páginas serão marcados e o cursor está na página 0.

26. Um estudante em um curso de design de compiladores propõe ao professor um projeto de escrever um compilador que produzirá uma lista de referências de páginas que podem ser usadas para implementar algoritmo ótimo de substituição de página. Isso é possível? Por quê? Existe algo que poderia ser feito para melhorar a eficiência da paginação no tempo de execução?
- Sim, pois o algoritmo de substituição de páginas é útil como uma medida-padrão pela qual outros algoritmos podem ser mensurados. podem ser mensurados. O algoritmo NRU divide as páginas em quatro classes, dependendo do estado dos bits R e M. Uma página aleatória da classe de ordem mais baixa é escolhida. Esse algoritmo é fácil de implementar, mas é muito rudimentar.
27. Suponha que a série de referências de páginas virtuais contém repetições de longas sequências de referências de páginas seguidas ocasionalmente por uma referência de página aleatória. Por exemplo, a sequência: 0, 1, ..., 511, 431, 0, 1, ..., 511, 332, 0, 1, ...consiste em repetições da sequência 0, 1, ... 511 seguida por uma referência aleatória às páginas 431 e 332.
- (a) Por que os algoritmos de substituição padrão (LRU, FIFO, relógio) não são efetivos ao lidar com essa carga de trabalho para uma alocação de páginas que é menor do que o comprimento da sequência?
- O algoritmo segunda chance é uma modificação do FIFO que confere se uma página está sendo usada antes de removê-la. Se ela estiver, a página é poupada. Essa modificação melhora muito o desempenho. O algoritmo do relógio é simplesmente uma implementação diferente do algoritmo segunda chance. Ele tem as mesmas propriedades de desempenho, mas leva um pouco menos de tempo para executar o algoritmo. O LRU é um algoritmo excelente, mas não pode ser implementado sem um hardware especial. Se o hardware não estiver disponível, ele não pode ser usado. O NFU é uma tentativa rudimentar, não muito boa, de aproximação do LRU. No entanto, o algoritmo do envelhecimento é uma aproximação muito melhor do LRU e pode ser implementado de maneira eficiente. Trata-se de uma boa escolha
- (b) Se fossem alocados 500 quadros de páginas para esse programa, descreva uma abordagem de substituição de página que teria um desempenho muito melhor do que os algoritmos LRU, FIFO ou de relógio.
- . O WSClock é uma variante que não apenas proporciona um bom desempenho, como também é eficiente de ser implementado. Como um todo, os dois melhores algoritmos são o do envelhecimento e o WSClock. Eles são baseados no LRU e no conjunto de trabalho, respectivamente. Ambos proporcionam um bom desempenho de paginação e podem ser implementados eficientemente. Alguns outros bons algoritmos existem, mas esses dois provavelmente são os mais importantes na prática.
28. Se a substituição de páginas FIFO é usada com quatro quadros de páginas e oito páginas, quantas faltas de páginas ocorrerão com relação à sequência 0172327103 se quatro quadros estiverem a princípio vazios? Agora repita esse problema para LRU.

29. Considere a sequência de páginas da Figura 3.15(b). Suponha que os bits R para as páginas B até A são 11011011, respectivamente. Qual página a segunda chance removerá? A segunda chance removerá a página R.
30. Um pequeno computador em um cartão inteligente tem quatro quadros de páginas. Na primeira interrupção de relógio, os bits R são 0111 (página 0 é 0, o resto é 1). Nas interrupções de relógio subsequentes, os valores são 1011, 1010, 1101, 0010, 1010, 1100 e 0001. Se o algoritmo de envelhecimento for usado com um contador de 8 bits, dê os valores dos quatro contadores após a última interrupção.
- Quase toda a memória tem que ser copiado, o que exige que cada palavra a ser lida e depois reescrita em um local diferente. Leitura 4 bytes leva 10 nanossegundos, para ler um byte leva 2,5 ns e escrever é preciso outro 2,5 ns, para um total de 5 ns por byte compactado. Esta é uma velocidade de 200 milhões de bytes / seg.
31. Um pequeno computador em um cartão inteligente tem quatro quadros de páginas. Na primeira interrupção de relógio, os bits R são 0111 (página 0 é 0, o resto é 1). Nas interrupções de relógio subsequentes, os valores são 1011, 1010, 1101, 0010, 1010, 1100 e 0001. Se o algoritmo de envelhecimento for usado com um contador de 8 bits, dê os valores dos quatro contadores após a última interrupção.
- Com páginas de 8 KB e um de 48 bits de espaço de endereço virtual, o número de virtual páginas é $248/213$, que é de 235 (cerca de 34 milhões)
32. No algoritmo WSClock da Figura 3.20(c), o ponteiro aponta para uma página com $R = 0$. Se $\tau = 400$, a página será removida? E se ele for $\tau = 1000$?
- Se as páginas podem ser compartilhados, sim. Por exemplo, se dois usuários de um sistema de tempo compartilhado a executar o mesmo editor, ao mesmo tempo, e o texto do programa é repartido em vez de copiados, algumas dessas páginas pode serem conjunto de trabalho de cada usuário em ao mesmo tempo.
33. Suponha que o algoritmo de substituição de página WSClock use um τ de dois tiques, e o estado do sistema é o seguinte:
- onde os bits V, R e M significam Válido, Referenciado e Modificado, respectivamente.
- Se uma interrupção de relógio ocorrer no tique 10, mostre o conteúdo das entradas da nova tabela. Explique. (Você pode omitir as entradas que seguirem inalteradas.) . O WSClock é uma variante que não apenas proporciona um bom desempenho, como também é eficiente de ser implementado.
 - Suponha que em vez de uma interrupção de relógio, ocorra uma falta de página no tique 10 por uma solicitação de leitura para a página 4. Mostre o conteúdo das entradas da nova tabela. Explique. (Você pode omitir as entradas que seguirem

inalteradas.). Como um todo, os dois melhores algoritmos são o do envelhecimento e o WSClock. Eles são baseados no LRU e no conjunto de trabalho, respectivamente. Ambos proporcionam um bom desempenho de paginação e podem ser implementados eficientemente. Alguns outros bons algoritmos existem, mas esses dois provavelmente são os mais importantes na prática.

34. Um estudante afirmou que “no abstrato, os algoritmos de substituição de páginas básicos (FIFO, LRU, ótimo) são idênticos, exceto pelo atributo usado para selecionar a página a ser substituída”.

(a) Qual é o atributo para o algoritmo FIFO? Algoritmo LRU? Algoritmo ótimo? LRU (usada menos recentemente) = Excelente algoritmo, porém difícil de ser implementado de maneira exata. FIFO (primeiro a entrar, primeiro a sair) = Pode descartar páginas importantes.

(b) Dê o algoritmo genérico para esses algoritmos de substituição de páginas. O algoritmo ótimo remove a página que será referenciada por último. Infelizmente, não há uma maneira para determinar qual página será essa, então, na prática, esse algoritmo não pode ser usado.

35. Quanto tempo é necessário para carregar um programa de 64 KB de um disco cujo tempo médio de procura é 5 ms, cujo tempo de rotação é 5 ms e cujas trilhas contêm 1 MB

(a) para um tamanho de página de 2 KB?

$$(64\text{KB} / 5 \text{ ms}) \times 2\text{KB} = 25,6 \text{ ms}$$

(b) para um tamanho de página de 4 KB? As páginas estão espalhadas aleatoriamente em torno do disco e o número de cilindros é tão grande que a chance de duas páginas estarem no mesmo cilindro é desprezível. $(64\text{KB} / 5 \text{ ms}) \times 4\text{KB} = 51,2 \text{ ms}$

36. Um computador tem quatro quadros de páginas. O tempo de carregamento, tempo de último acesso e os bits R e M para cada página são como mostrados a seguir (os tempos estão em tiques de relógio):

(a) Qual página NRU substituirá? A página 1

(b) Qual página FIFO substituirá? A página 3

(c) Qual página LRU substituirá? A página 2

(d) Qual página segunda chance substituirá? A página 3

37. Suponha que dois processos A e B compartilhem uma página que não está na memória. Se o processo A gera uma falta na página compartilhada, a entrada de tabela de página para o processo A deve ser atualizada assim que a página for lida na memória.

(a) Em quais condições a atualização da tabela de páginas para o processo B deve ser atrasada, mesmo que o tratamento da falta da página A traga a página compartilhada para a memória? Explique. Ter um algoritmo para alocar quadros de páginas para processos. Uma maneira é determinar periodicamente o número de processos em execução e alocar a cada processo uma porção igual. Desse modo, com 12.416 quadros de páginas disponíveis (isto é, sistema não operacional) e 10

processos,

(b) Qual é o custo potencial de se atrasar a atualização da tabela de páginas?

Em geral, algoritmos globais funcionam melhor, especialmente quando o tamanho do conjunto de trabalho puder variar muito através do tempo de vida de um processo. Se um algoritmo local for usado e o conjunto de trabalho crescer, resultará em ultra paginação, mesmo se houver um número suficiente de quadros de páginas disponíveis. Se o conjunto de trabalho diminuir, os algoritmos locais vão desperdiçar memória. Se um algoritmo global for usado, o sistema terá de decidir continuamente quantos quadros de páginas designar para cada processo.

38.

39. Você foi contratado por uma companhia de computação na nuvem que emprega milhares de servidores em cada um dos seus centros de dados. Eles ouviram falar recentemente que valeria a pena lidar com uma falta de página no servidor A lendo a página da memória RAM de algum outro servidor em vez do seu drive de disco local.

(a) Como isso poderia ser feito?

Com aplicação do algoritmo de substituição

(b) Em quais condições a abordagem valeria a pena? Seria factível?

Em situações que há necessidade de um controle e gerenciamento de memória.

40. Uma das primeiras máquinas de compartilhamento de tempo, o DEC PDP-1, tinha uma memória (núcleo) de palavras de 18 bits e 4K. Ele executava um processo de cada vez em sua memória. Quando o escalonador decidia executar outro processo, o processo na memória era escrito para um tambor de paginação, com palavras de 18 bits e 4K em torno da circunferência do tambor. O tambor podia começar a escrever (ou ler) em qualquer palavra, em vez de somente na palavra 0. Por que você acha que esse tambor foi escolhido?

a. O tambor PDP-1 paging tinha a vantagem de não haver latência rotacional. Este salvo meia rotação cada memória um tempo foi escrito ao tambor

41. Um computador fornece a cada processo 65.536 bytes de espaço de endereçamento divididos em páginas de 4096 bytes cada. Um programa em particular tem um tamanho de texto de 32.768 bytes, um tamanho de dados de 16.386 bytes e um tamanho de pilha de 15.870 bytes. Esse programa caberá no espaço de endereçamento da máquina? Suponha que em vez de 4096 bytes, o tamanho da página fosse 512 bytes eles caberiam então? Cada página deve conter texto, dados, ou pilha, não uma mistura de dois ou três deles.

a. Se as páginas podem ser compartilhados, sim. Por exemplo, se dois usuários de um sistema de tempo compartilhado a executar o mesmo editor, ao mesmo tempo, e o texto do programa é repartido em vez de copiados, algumas dessas páginas pode serem conjunto de trabalho de cada usuário em ao mesmo tempo.

42. . Observou-se que o número de instruções executadas entre faltas de páginas é diretamente proporcional ao número de quadros de páginas alocadas para um programa. Se a memória disponível for dobrada, o intervalo médio entre as faltas de páginas também será dobrado. Suponha que uma instrução normal leva 1 ms, mas se uma falta de página ocorrer, ela leva 2001 μ s (isto é, 2 ms) para lidar com a falta. Se um programa leva 60 s para ser executado, tempo em que ocorrem 15.000 faltas de páginas, quanto tempo ele levaria para ser executado se duas vezes mais memória estivesse disponível?
- a. A metade do tempo gasto inicialmente para um processo.
43. Um grupo de projetistas de sistemas operacionais para a Frugal Computer Company está pensando sobre maneiras para reduzir o montante de armazenamento de apoio necessário em seu novo sistema operacional. O chefe deles sugeriu há pouco não se incomodarem em salvar o código do programa na área de troca, mas apenas paginá-lo diretamente do arquivo binário onde quer que ele seja necessário. Em quais condições, se houver, essa ideia funciona para o código do programa? Em quais condições, se houver, isso funciona para os dados?
- a. Há quatro momentos em que o sistema operacional tem de se envolver com a paginação: na criação do processo, na execução do processo, em faltas de páginas e no término do processo.
44. . Uma instrução em linguagem de máquina para carregar uma palavra de 32 bits em um registrador contém o endereço de 32 bits da palavra a ser carregada. Qual o número máximo de faltas de páginas que essa instrução pode causar?
- a. A pesquisa deve ser feita em cada referência de memória.
45. Explique a diferença entre fragmentação interna e fragmentação externa. Qual delas ocorre nos sistemas de paginação? Qual delas ocorre em sistemas usando segmentação pura?
- a. Fragmentação interna: O espaço extra que é desperdiçado na página
Fragmentação externa: A divisão da memória em uma série de pedaços, alguns contendo segmentos e outros lacunas.
46. Quando tanto segmentação quanto paginação estão sendo usadas, como em MULTICS, primeiro o descritor do sistema precisa ser examinado, então o descritor de páginas. A TLB também funciona dessa maneira, com dois níveis de verificação?
- a. Sim, também existe a verificação de páginas
47. Consideramos um programa que tem os dois segmentos mostrados a seguir consistindo em instruções no segmento 0, e dados leitura/escrita no segmento 1. O segmento 0 tem proteção contra leitura/execução, e o segmento 1 tem proteção apenas contra leitura/escrita. O sistema de memória é um sistema de memória virtual paginado por demanda com endereços virtuais que tem números de páginas de 4 bits e um deslocamento de 10 bits. Para cada um dos casos a seguir, dê o endereço de real memória real (efetivo) que resulta da tradução dinâmica de endereço, ou identifique o tipo de falta que ocorre (falta de página ou proteção).

- (a) Busque do segmento 1, página 1, deslocamento 3.
Quadro de página 14
- (b) Armazene no segmento 0, página 0, deslocamento 16.
No disco
- (c) Busque do segmento 1, página 4, deslocamento 28.
No disco
- (d) Salte para localização no segmento 1, página 3, deslocamento 32.
Quadro de página 6

48. Você consegue pensar em alguma situação onde dar suporte à memória virtual seria uma má ideia, e o que seria ganho ao não dar suporte à memória virtual? Explique.
- a. Quando cada programa tem seu próprio espaço de endereçamento, o qual é dividido em blocos chamados de páginas.
49. A memória virtual fornece um mecanismo para isolar um processo do outro. Quais dificuldades de gerenciamento de memória estariam envolvidas ao permitir que dois sistemas operacionais fossem executados ao mesmo tempo? Como poderíamos lidar com essas dificuldades?
- a. Em vez de ter realocações separadas apenas para os segmentos de texto e dados, todo o espaço de endereçamento pode ser mapeado na memória física em unidades razoavelmente pequenas. A memória virtual funciona bem em um sistema de multiprogramação, com pedaços e partes de muitos programas na memória simultaneamente. Enquanto um programa está esperando que partes de si mesmo sejam lidas, a CPU pode ser dada para outro processo.