**Exercícios Dierone**

[Exercícios](http://www.deinfo.uepg.br/~dierone/so_2018/S12_A.T_12.1_08.05.2018__Exercicios_GerenciaMemoria.pdf)

1) Um processo pode executar sem ter todas as instruções e dados dentro da memória principal. O espaço de memória disponível ao programa pode exceder o tamanho da memória principal. Existem duas formas de implementação da memória virtual: paginação e segmentação.  
Paginação: invisível ao programador, serve para prover um espaço maior de endereçamento. Memória dividida em páginas de igual tamanho, com qualquer conteúdo.  
Segmentação: em geral visível ao programador, serve para organizar programas de dados, associando atributos de privilégio e proteção a instruções e dados. Memória dividida em segmentos de programas e segmentos de dados de tamanhos variados e dinâmicos.  
  
2) Alocação particionada acontece quando a eficiência da multiprogramação exige que vários programas estejam na memória ao mesmo tempo, tendo necessidade de organizar a memória. Divide-se em estática e dinâmica.   
Alocação particionada estática: -Divisão da memória em tamanhos fixos (partições) definidos na inicialização do Sistema em função dos programas que executariam no ambiente.  
 - A alteração do tamanho de uma partição necessita a inicialização do Sistema Operacional.  
 - Os programas só podiam executar em uma das partições, mesmo com outras disponíveis.  
  
Alocação particionada dinâmica:

* Aumento do grau de compartilhamento diminuindo o problema da fragmentação.
* Partições sem tamanho fixo, onde cada programa utiliza o espaço que necessita.
* Existe ainda o problema de fragmentação, conforme os programas vão terminando e deixando espaços cada vez menores.  
    
  Alocação contígua foi implementada nos primeiros Sistemas Operacionais e ainda existentes em alguns sistemas monoprogramáveis.   
  Características:
* Memória dividida em duas partes, Sistema Operacional e programa do usuário.
* O programador tem controle sobre toda a memória principal, podendo acessar qualquer posição da memória, inclusive onde está residente o Sistema Operacional.
* Um mecanismo de proteção utilizado é delimitar a área do Sistema Operacional que delimita a área do mesmo.
* Fácil implementação e código reduzido, porém Ineficiência no uso do processador e da memória pois apenas um usuário pode dispor desse recurso.
* Programas limitados ao tamanho da memória disponível.
* Overlay (sobreposição) – Solução encontrada para dividir o programa em partes (módulos), de forma que pudessem executar independentemente uma da outra, utilizando uma mesma área de memória.  
    
  3) A paginação por demanda é baseada na paginação simples, ou seja, a memória lógica é dividida em páginas que podem ser colocadas em qualquer quadro da memória física; além de a TP ser usada para conversão de endereços lógicos em físicos.   
  Na paginação por demanda apenas as páginas que o processo acessa são carregadas para a memória física. O bit de válido/inválido indica se a página já está presente na memória ou se ainda está no disco.  
  Na segmentação por demanda FALTA   
    
    
  4) A falta de página é uma interrupção causada quando um processo acessa uma página cujo bit de validade vale zero.   
  Tratamento de falta de página:  
   Gera um trap (interrupção por SW) para o núcleo;  
  • Salvamento do contexto de P, que passa a ser bloqueado;  
  • Possivelmente no meio de uma instrução!  
  • O núcleo pega a CPU;  
  • Identificação do IRQ (vetor de interrupções);  
  • Localização do endereço da página no disco;  
  • Solicitação de E/S ao controlador do disco  
  • Inicia-se a transferência de dados disco/RAM  
  • O núcleo devolve a CPU a um processo Q.  
  • Quando a E/S tiver terminado, gera uma interrupção de HW.  
  • O núcleo pega a CPU e atualiza a tabela de páginas de P.  
  • O núcleo passa P do estado bloqueado para o estado pronto.  
    
    
  5) Os algoritmos de substituição de páginas são políticas definidas para escolher qual(is) página(s) da memória dará lugar a página que foi solicitada e que precisa ser carregada. Isso é necessário quando não há espaço disponível para armazenar a nova página. O algoritmo ótimo é um algoritmo em cada página será marcada com o número de instruções que serão executadas antes que a página seja referenciada, além disso, ele retira da memória a página que tem menos chance de ser referenciada (maior número de instruções faltantes). É um algoritmo impraticável, praticamente impossível de se saber. Utilizado em simulações para comparações com outros algoritmos. Ele não é implementável pois seria necessário prever o futuro da cadeia de referência.  
    
  6)FALTA  
    
    
  7)   
  - Utiliza dois bits de status: R(referenciado) e M(modificado).  
  - Quando o processo inicia, suas páginas ainda não estão presentes na memória. Assim que uma delas é referenciada, o bit R é colocado em 1.  
    
  - Em seguida, se esta página é modificada, o bit M é colocado em 1.  
  - Ao ocorrer uma page fault o sistema operacional separa todas as páginas em quatro categorias:  
   -> Classe 0: não referenciada, não modificada.  
   -> Classe 1: não referenciada, modificada.  
   -> Classe 2: referenciada, não modificada.  
   -> Classe 3: referenciada, modificada.  
  - O NRU então remove uma página aleatória da classe mais baixa que não esteja vazia.  
  - Entre as vantagens está a baixa complexidade de entendimento e implementação e a boa aproximação para o algoritmo ótimo.  
    
    
  8)   
  - SO mantém uma fila das páginas correntes na memória.  
  - A página no início da fila é a mais antiga e a página no final é a mais nova.  
  - Quando ocorre um page fault a página do início é removida e a nova é inserida ao final da fila.   
  - Simples, mas pode ser ineficiente, pois uma página que está em uso constante pode ser retirada.  
    
  9)FALTA  
    
    
  10)Algoritmo de segunda chance é a implementação do FIFO que analisa o bit de referência de cada página candidata, para saber se ela foi acessada recentemente. Caso essa página tenha sido acessada recentemente (bit=1), ela recebe um "segunda chance" e volta para o fim da fila com o bit ajustado para 0.  
  Pode ser visualizado como uma lista circular, ou relógio. A sua vantagem é que ao invés de excluir a página direto ele verifica se a página foi acessada ou não.  
    
  11) FALTA  
    
    
  13)  
    
  a) Não usada recentemente   
  b) Página usada a mais tempo   
  c) Ótimo  
  d) FIFO  
  e) Segunda chance   
    
  ( a ) Cada página apresenta dois bits formando um número binário (1º bit - referenciado, 2º bit modificado) que menor seu valor maior será a possibilidade da mesma ser substituída.  
  ( c ) A página que será usada mais tarde é a candidata a substituição.   
  ( e ) As páginas fazem parte de uma fila em que a primeira página a ser transferida para a memória é a primeira de uma fila, candidata a ser substituída, a não ser que seja utilizada, quando então irá para o fim da fila deixando de ser candidata.  
  ( b ) Páginas referenciadas pela última vez a mais tempo são candidatas a substituição.  
  ( d ) A primeira página a ser transferida para a memória principal será a primeira a ser substituída.

14)

15)

1. ( **V** ) A fragmentação interna ocorre entre as páginas de memória.
2. ( **F** ) Na segmentação ocorre a fragmentação interna.

**Na segmentação ocorre fragmentação externa**

1. ( **F** ) A finalidade principal da memória virtual é de permitir que arquivos maiores do que o espaço de memória principal disponíveis sejam abertos.

**A finalidade principal da memória virtual é permitir o compartilhamento seguro e eficiente da memória entre vários programas e remover os transtornos de programação de uma quantidade pequena e limitada na memória principal.**

1. ( **F** ) Páginas grandes minimizam a fragmentação.

**Páginas grandes aumentam a fragmentação**

16)

