

## MAC0422 – Sistemas Operacionais

### Prova 2

22/11/2018

Prof. Alan Durham

Esta prova tem duas partes. A **primeira parte** contém 40 afirmações, você deve selecionar 13 destas que estão ERRADAS. Sua folha de resposta deve conter as questões erradas **NA ORDEM** com uma justificativa do porquê estão erradas. A justificativa deve ter apenas uma sentença (curta, não deve ter mais que 3 linhas). Respostas sem justificativa não serão consideradas. Cada resposta vale 1 ponto. Não serão consideradas mais que 13 respostas. A **segunda parte** tem três questões dissertativas, que devem ser respondidas no espaço alocado. Cada questão vale 2 pontos.

#### PARTE I

1. A técnica de overlay tem como objetivo fazer dois programas rodarem, alternadamente, no mesmo espaço de memória real.
2. No esquema de partições fixas, cada partição de memória tem sua própria fila de escalonamento.
3. No esquema de partições fixas, pode-se fazer programas rodarem em partições diferentes utilizando a técnica de tradução e carregamento relocáveis. Nesta técnica o texto do programa inclui uma tabela de endereços.
4. O esquema de partições variáveis de memória real convite em partições de vários tamanhos pré-determinados para poder melhor acomodar programas de tamanhos diferentes com menor fragmentação interna.
5. A técnica de alocação worst fit foi assim chamada por ser considerada universalmente ruim. A melhor técnica é sempre a chamada Best fit.
6. No esquema de “buddy system” para administração de lista livre em memória real as áreas livres têm todas tamanhos que são potência de 2.
7. Na técnica de Swapping em memória real programas podem entrar e sair da memória para maximizar o uso de recursos do sistema.
8. Em esquemas de memória virtual um programa não precisa estar contíguo na memória.
9. Em esquemas de paginação simples o overhead para uso de memória virtual é de um acesso e uma soma a mais.
10. TLB é uma técnica de caches dentro do processador que diminuem o tempo de acesso à valores na memória.
11. Em esquemas segmentados puros podemos fazer a divisão lógica dos blocos de código do programa, porém isso gera um problema de fragmentação externa.
12. Em esquemas de paginação simples a divisão dos programas é por blocos de tamanho fixo, isso gera um problema de fragmentação interna.
13. Esquemas de segmentação criam espaços naturais de compartilhamento, potencialmente diminuindo o uso de memória real em sistemas com muitos usuários.

14. Em esquemas de paginação segmentada podemos dividir grupos de páginas em segmentos lógicos, porém temos a desvantagem de termos fragmentação interna e externa.
15. Nos sistemas modernos de memória virtual o overhead médio para uso de memória virtual é de apenas alguns ciclos. Isso se deve a técnicas especiais de processamento direto dentro do chip de memória, executando operações memória a memória em tempo muito reduzido.
16. TLBs otimizam o uso de memória virtual, mas infelizmente são aplicadas apenas em memórias paginadas. Isso determinou o fim da memória virtual segmentada pura.
17. Com o aumento do tamanho das memórias virtuais, é necessário que todos os programas compartilhem apenas uma memória virtual. Isso evita que a tabela de blocos ocupe um espaço excessivo de memória.
18. As estratégias de colocação em memórias segmentadas puras são as mesmas encontradas em sistemas de memória real com partições variáveis.
19. Quando esperamos que vigore o princípio da localidade, não faz sentido as estratégias de “pre-fetching” de blocos (antecipação de carga de blocos).
20. Estratégias de reposição de páginas tentam escolher, dentre as páginas residentes de memória, qual a que tem menos chance de ser usada no futuro próximo. No caso geral é impossível garantir isso.
21. O esquema NUR (*not used recently* ou não usada recentemente) utiliza o dirty bit e o access bit para definir a próxima página a ser reposta. A prioridade é dada a páginas com o access bit e o *dirty bit* zerados seguidas daquelas com apenas o referende (access) bit zerado.
22. O esquema NUR tem ótimo desempenho, infelizmente necessita dos bits de acesso e dirty bit para ser executado. Sem estes, é impossível.
23. O esquema “FIFO segunda chance” é uma variação de FIFO que usa o “reference bit” e o princípio da localidade para melhorar a política FIFO.
24. A estratégia de “working set” determina uma “janela” de acessos. O working set consiste das páginas acessadas nesta janela.
25. Os processadores ARM trabalham com tamanhos de página distintos coexistentes no mesmo sistema.
26. Os processadores Intel trabalham com uma escolha de tamanhos de página, mas apenas um tamanho é utilizado de cada vez.
27. O esquema de tabelas de tabela de página invertida se implementado sem nenhuma otimização é inviável, pois acarretaria um overhead linear no tamanho da memória.
28. Arquivos podem ser organizados em registros ou em bytes. Os arquivos Unix são organizados em bytes.
29. Diretórios são sempre, em todos os sistemas, arquivos especiais que contém informações sobre cada arquivo como permissões de acesso.
30. Sistemas de servidores muitas vezes contém arquivos especiais, com registros formatados e mecanismos para acesso eficiente. Um exemplo é o ISAM (index-sequential access method), que pode se utilizado tanto para acesso eficiente sequencial, como para acesso eficiente por índice.

31. Em Unix temos três tipos de arquivos: caractere, binário e archive. Eles são identificados pelo sistema utilizando o “número mágico”.
32. Em operações de leitura, o usuário deve fornecer uma área de escrita. Se isso não for feito fica difícil prever o overhead de espaço necessário para os sistema de arquivos funcionar.
33. Hard links podem ser mais seguros do que soft links.
34. Soft links podem ser mais seguros do que hard links.
35. Arquivos em discos rígidos são sempre alocados em blocos que podem ser distribuídos qualquer lugar do disco.
36. No sistema de arquivo FAT utiliza cache “write through” onde qualquer modificação na cache é imediatamente gravada em disco. O princípio da localidade garante que isso não gera overhead significativo.
37. Os stems Unix realizam operações periódicas de sincronização de caches, assim se um programa morrer inesperadamente alguns dados podem não ter sido gravados.
38. Sistemas modernos estão cada vez mais seguros, técnicas modernas de segurança garantem a inviabilidade de sistemas críticos, mesmo quando ligados à internet. Infelizmente o alto custo destes mecanismos faz com que a maioria das grandes corporações não o utilizem, o que gera as falhas de segurança noticiadas regularmente.
39. Um dos princípios de desenho de sistemas de segurança é que o esquema deve ser público, não deveria depender do sigilo sobre seu mecanismo de funcionamento.
40. Alguns sistemas criam logins de fácil invasão para tentar localizar o uso malicioso do sistema.

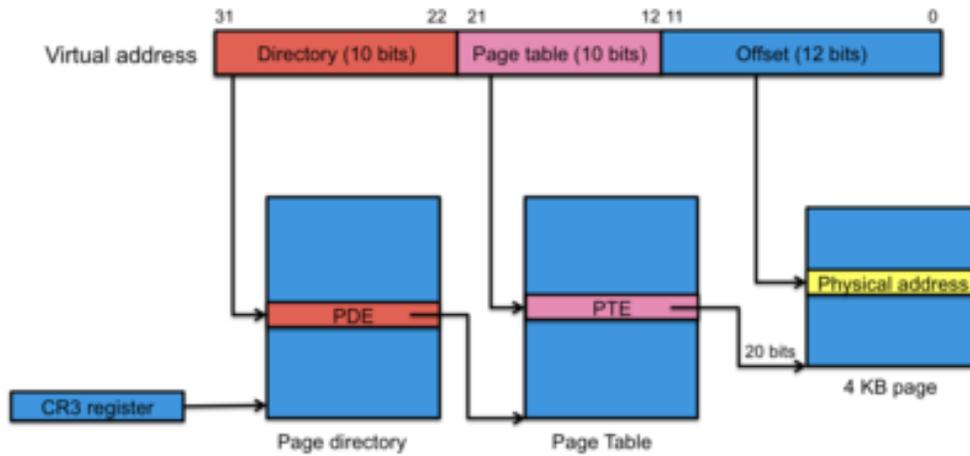
COLOQUE AQUI A LISTA DE QUESTÕES ERRADAS E A JUSTIFICATIVA (CURTA) DE CADA UMA.







3. (2 pontos) A figura abaixo mostra um dos esquemas de endereçamento de processadores intel.



Explique qual a vantagem de paginação em vários níveis. Por que, mesmo com tantas tabelas em condições normais, o *overhead* de acesso e de espaço para uso de memória virtual é pequeno?