

Arquitetura e Implementação de Sistemas de Operação

Mestrado Integrado em Engenharia Informática

Exame de Recurso – 2015/06/30 – Duração: 2h30m

Teste sem consulta. Explícite nas suas respostas todas as hipóteses assumidas.

Q-1 [2.0 val.] Explique o conceito de *rotina de atendimento de interrupção* e indique que tipos de eventos podem desencadear a execução de tais rotinas.

Q-2 [2.0 val.] Considere um Sistema de Operação (S.O.) com *kernel threads*, ou seja, cuja execução é levada a cabo por *threads* que correm no contexto do próprio S.O. Em que situações tais *threads* executam código em modo supervisor?

Q-3 [2.0 val.] No contexto da implementação de um S.O.,

- a) que mecanismos de gestão de concorrência são necessários para garantir exclusão mútua a estruturas de dados do S.O. numa arquitetura *multi-core*?
- b) em que cenários os *spinlocks* são preferíveis a *locks* com fila de *threads* em espera?

Q-4 [2.0 val.] No contexto da virtualização de *hardware* explique a diferença entre instruções sensíveis e privilegiadas, e em que medida estas afetam a implementação de um monitor de máquina virtual (VMM).

Q-5 [2.0 val.] Considere um cenário em que se está a receber um fluxo de pedidos, todos com as mesmas características e aproximadamente com o mesmo tempo de execução. Que política de escalonamento melhor de adapta a este cenário? Justifique.

Q-6 [2.0 val.] Suponha que numa arquitetura *multi-core* se executam simultaneamente várias aplicações que fazem uso de sistema de execução paralelos, por exemplo OpenMP, OpenCL, Storm, MapReduce, etc. Suponha ainda que, em cada uma destas execuções, se divide o trabalho pelo número de *threads hardware* disponíveis na máquina.

- a) Na presença de um escalonador usual, como um baseado em *Multi-Level Feedback Queue*, o desempenho da execução concorrente destas aplicações pode ser bastante pior do que a sequencial. Porquê?
- b) Que novas funcionalidades ou optimizações propõe para resolver as limitações do escalonador referido na alínea anterior?

Q-7 [2.0 val.] Num S.O. com *kernel threads*, indique quais dos seguintes itens são guardados no *Process Control Block* (PCB), no *Thread Control Block* (TCB) ou em nenhum destes:

- | | |
|--|---------------------------------|
| a) apontador para a tabela de páginas | e) registos do processador |
| b) tabela de páginas | f) apontador para o <i>heap</i> |
| c) apontador para a pilha (<i>stack</i>) | g) estado para escalonamento |
| d) fila de prontos | h) ficheiros abertos |

Q-8 [2.0 val.] No contexto de gestão de memória baseada em paginação,

- a) quando é que uma página tem de ser escrita em disco?
- b) essa escrita pode ser assíncrona (ou seja realizada mais tarde no tempo), quais as vantagens?

Q-9 [2.0 val.] Defina o conceito de sistema de ficheiros baseado em *extents*, do qual o NTFS é um exemplo. Indique também uma vantagem e uma desvantagem desta abordagem relativamente às que utilizam blocos de tamanho fixo.

Q-10 [2.0 val.] Qual é o tamanho máximo para um ficheiro num sistema de ficheiros baseado em *i-nodes* com:

- blocos (lógicos) de 1MB,
- endereços de blocos de 32 bits,
- 100 apontadores para endereçamento directo, e
- 1 apontador para endereçamento indirecto simples.