

## Parte Teórica

- 1) Duas características dos sistemas de discos RAID são permitir aumentar o débito de leitura e escrita de dados e disponibilizar protecção contra perdas de informação por falhas de hardware.
  - a) Explique em que consiste o RAID 0, *stripping*, valendo-se de um diagrama.
  - b) Explique em que consiste o RAID 1, *mirroring*, valendo-se de um diagrama.
  - c) O sistema RAID10 combina as vantagens dos sistemas RAID 0 e RAID 1. Na prática, a cada elemento (*strip*) do RAID 0 é aplicado um sistema RAID 1 autónomo. Tente expor este funcionamento recorrendo a um diagrama.
  
- 2) Relativamente a *threads* de utilizador e *threads* de *kernel*:
  - a) Por que motivo é que a comutação entre *threads* de utilizador é mais rápida?
  - b) Que faz o *kernel* quando uma *thread* de utilizador bloqueia num acesso de E/S?
  
- 3) No Windows 2000:
  - c) O *quantum* de cada *thread* varia conforme a versão do sistema: no W2K Professional é de 20 ms, no W2K Server (monoprocessador) é de 120 ms. Que consequências advêm desta diferença de *quantum*? Que tipo de processos é prejudicado ou favorecido em cada caso?
  - d) O escalonador utiliza um algoritmo de escalonamento de *threads* por prioridades, com *round-robin* das listas de *threads*. Desenhe um diagrama explicativo deste tipo de algoritmos e descreva de forma genérica e resumida o respectivo funcionamento.
  - e) O problema da inversão de prioridades, no escalonamento, é resolvido da seguinte forma: quando uma *thread* fica inactiva durante mais do que um tempo-limite, é-lhe atribuída a prioridade 15 (muito elevada, só atribuível pelo sistema) durante dois *quanta*. Explique em que consiste o problema de inversão de prioridade e por que motivo é que esta abordagem o resolve.
  
- 4) Relativamente ao algoritmo *buddy* para gestão de memória (utilizado pelo Linux):
  - a) Num sistema de 256 MB de memória, explique, recorrendo a um diagrama, quantos blocos livres existem e quais os respectivos tamanhos, após a seguinte sequência de pedidos de memória: 5 k, 25 k, 35 k, 1 MB e 20 k. Considere que inicialmente cerca de 24 MB tinham sido atribuídos para o sistema operativo.
  - b) Calcule qual o espaço desperdiçado, devido à fragmentação interna, em cada um dos blocos mencionados na alínea anterior. (Nota: o Linux faz sub-gestão da memória dentro de cada bloco, para obviar este problema, mas pode ignorar esse comportamento na resposta a esta pergunta.)
  
- 5) Distinga recursos preemptáveis de preemptáveis.