

Critérios de Correção

Unidade Curricular: 21111 - Sistemas Operativos

Prova: Época normal, 03 de julho de 2017

Exame: Constituído pelos grupos I e II.

p-fólio: Constituído pelo grupo I.

As cotações são indicadas por grupo e nas próprias questões.

Instruções a Verificar

- O teste deve estar resolvido na sua totalidade em folhas de respostas.
- Nas respostas, a letra deve ser legível.
- Todas as respostas devem estar escritas unicamente com caneta azul ou preta.
- O não cumprimento destas instruções implica a anulação das respetivas questões.

Tópicos de referência para as respostas

Nas questões de resposta aberta, não existem respostas únicas ou definitivas e na maioria dos casos espera-se que o aluno seja capaz de apresentar respostas mais desenvolvidas/elaboradas relativamente aos tópicos aqui fornecidos.

Nas questões de escrita de programas, a sua correção tem em conta critérios de proficiência e compreensibilidade do código tais como: legibilidade, indentação, estrutura, comentários e explicação geral do seu funcionamento.

Nota: as consultas referem-se a secções do livro recomendado MOS3e - Modern Operating Systems, 3ª ed., Andrew S. Tanenbaum.

Grupo I [12 valores]

- 1.1.** [1.2] Uma interface virtual baseada em abstrações (como modelo de um computador) é necessária para tornar praticável o desenvolvimento de aplicações sem o programador ter de conhecer todos os pormenores técnicos de operação de cada dispositivo de um computador. O SO providencia assim um conjunto claro de recursos abstratos, relativamente fáceis de entender e utilizar, em vez da interface direta com o hardware, que requer conhecimentos técnicos de elevada complexidade e é difícil de programar.
Consultar sec. 1, 1.1, 1.1.1
- 1.2.** [1.2] SO com estrutura modular onde apenas um dos módulos, o micronúcleo, executa em modo núcleo e os restantes em modo utilizador, como por exemplo os gestores de dispositivos e os sistemas de ficheiros, de modo a conseguir um elevado nível de fiabilidade.
Consultar sec. 1.7.3
- 1.3.** [1.2] Um processo que se encontre no estado bloqueado pode apenas transitar para o estado executável (ready). Esta transição é efetuada quando ocorre o evento pelo qual o processo estava à espera.
Consultar sec. 2.1.5

- 1.4.** [1.2] Mencionar pelo menos 3 razões pertinentes, nomeadamente: (i) Natureza paralela de algumas aplicações; (ii) Mais rápidas de criar/destruir que os processos; (iii) Melhor rentabilização do CPU no caso de operações de I/O frequentes em paralelo com operações de computação; (iv) Paralelismo real com aumento da velocidade de processamento em sistemas multiprocessador; (v) Partilha do espaço de endereçamento e de recursos.
Consultar sec. 2.2, 2.2.1
- 1.5.** [1.2] Este método de sincronização aplica-se a um grupo de processos. Cada processo que chega à barreira é bloqueado até que o último processo do grupo chegue à barreira, altura em que são todos desbloqueados (passam ao estado executável).
Consultar sec. 2.3.9
- 1.6.** [1.2] 1) Atribuição de recursos em exclusão mútua; 2) Processo pode reter um recurso e esperar por outro; 3) Não preempção de recursos; 4) Espera circular.
Consultar sec. 6.2.1
- 1.7.** [1.2] A TLB é uma tabela implementada por hardware, normalmente na unidade de memória, para acelerar a conversão de endereços virtuais para físicos sem ser necessário aceder à tabela de endereços residente em memória, o que seria extremamente lento. A TLB funciona como uma memória cache para a tabela de endereços.
Consultar sec. 3.3.3
- 1.8.** [1.2] Consiste num processo causar uma falha de página com elevada frequência, resultando em extrema lentidão de execução. Ocorre quando o conjunto de trabalho (working set) é demasiado grande para caber todo na memória disponível.
Consultar sec. 3.4.8
- 1.9.** [1.2] Um disco fragmentado é aquele em que os blocos livres não se encontram numa área contígua do disco. Quando é criado um novo ficheiro, os blocos que o constituem vão estar dispersos pelo disco, diminuindo o desempenho no acesso ao ficheiro.
Consultar sec. 4.4.5
- 1.10.** [1.2] Independência do dispositivo como a operação de diferentes dispositivos com a mesma interface de programação. Consultar sec. 5.2.1

Grupo II [8 valores]

- 2.1.** [3] Programa avaliado segundo a sua estrutura, simplicidade e implementação das características pedidas.
- 2.2.** [5] Programa avaliado segundo a sua estrutura, simplicidade e implementação das características pedidas.

FIM