

Época Normal 2016/2017

1- Num sistema com preemptive multitasking, quando um processo começa a sua tarefa, não é necessário esperar até ao final dessa mesma tarefa para mudar de processo, isto pode tornar o escalonamento mais "justo" e executar vários processos (aparentemente) ao mesmo tempo.

2- Um long-term scheduler seleciona os processos do disco para colocar na ready queue e um short-term scheduler seleciona os processos da ready queue a colocar no CPU.

$$3- 512 \text{ MB} = 2^{29} \text{ B physical}$$

$$1 \text{ GB} = 2^{30} \text{ B virtual}$$

$$2^{20} \text{ pages}$$

a) 2^{20}

b) $\frac{2^{30}}{2^{20}} = 2^{10} \text{ B} = 1 \text{ KB}$

c) $2^{20} \cdot 2^3 = 2^{23} \text{ B} = 8 \text{ MB}$

d) Cada frame tem 1KB, então há $\frac{2^{29}}{2^{10}} = 2^{19}$ frames

$\log_2(2^{19}) = 19 = p$ $d = \log_2(2^{10}) = 10$ $19 + 10 = 29 \text{ bits}$

e) $\log_2(2^{20}) + \log_2(2^{10}) = 30 \text{ bits}$

4-

| | P | P | P | | P | PS | | PS | P | PS | |
|-----|------|------|------|------|------|------|------|------|------|------|------|
| LRU | W(1) | W(3) | R(4) | R(1) | R(5) | W(6) | R(4) | W(2) | R(1) | R(5) | W(1) |
| F0 | 1' | 1' | 1' | 1' | 1' | 1' | 1' | 2' | 2' | 2' | 2' |
| F1 | | 3' | 3' | 3' | 3' | 6' | 6' | 6' | 6' | 5 | 5 |
| F2 | | | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 |
| F3 | | | | | 5 | 5 | 5 | 5 | 1 | 1 | 1' |

PF $\rightarrow 8$

SO $\rightarrow 3$

| | P | P | P | P | PS | PS | P | | | | |
|-------|----|----|----|----|------|---------------------|---------------------|-----|-------------------|------------------|---------------------|
| CLOCK | W1 | W3 | R4 | R1 | R5 | W6 | R4 | W2 | R1 | R5 | W1 |
| F0 | 1' | 1' | 1' | 1' | (1') | 6' | 6' | 6' | 6' ¹⁰ | 6' ¹⁰ | 6' ¹⁰ |
| F1 | 0 | 3' | 3' | 3' | 3' | (3' ¹⁰) | (3' ¹⁰) | 2' | 2' ¹⁰ | 2' ¹⁰ | (2' ¹⁰) |
| F2 | | 0 | 4 | 4 | 4 | 4 ⁰ | 4 | (4) | 1 | 1 | 1' |
| F3 | | | 0 | | 5 | 5 ⁰ | 5 | 5 | (5 ⁰) | (5) | (5) |

PF → 8

SO → 2

5 a) RAID 1 - Operações de escrita são mais rápidas graças à ausência do cálculo da paridade do RAID 5

b) RAID 5 - É possível reconstruir dados perdidos graças à paridade, leitura é também mais rápida pois o striping permite leituras em paralelo

6 10 d $\text{block} = 4 \text{KB} = 2^{12} \text{B}$
 1 i $\text{ponteiro} = 64 \text{bits} = 8 \text{B} = 2^3 \text{B}$
 a) 1 ii

d - $10 \cdot 2^{12} = 40 \text{KB}$

i - Cada bloco tem $\frac{2^{12}}{2^3} = 2^9$ ponteiros $\Rightarrow (\Sigma) 1 \text{GB} + 2 \text{MB} + 40 \text{KB}$

$2^9 \cdot 2^{12} = 2^{21} = 2 \text{MB}$

ii - $2^9 \cdot 2^9 \cdot 2^{12} = 2^{30} = 1 \text{GB}$

b) $2 \text{MB} = 2^{21} \Rightarrow \frac{2^{21}}{2^{12}} = 2^9 = 512 \text{ blocos}$

10 lixos com d

502 lixos com i

1 ponteiro para bloco de ponteiros

R: 513