

Lista 1 - Sistemas Operacionais

Aluno: Henrique Oliveira da Cunha Franco

Q1) Letras a, c, d.

Q2) De acordo com o guia do usuário, e conforme as permissões fornecidas, ele poderá ler e executar o arquivo, mas não escrever (r-x). Somente o dono é capaz de fazer isso.

Q3) SIGALARM é um sinal enviado a um processo quando seu timer expira, e é normalmente usado para limitar o tempo de execução do processo. O SIGALARM pode ser ignorado dependendo da situação, por exemplo quando o programa utiliza funcionalidades opcionais e não quer que ele seja completamente interrompido. Alternativamente, o alarme NÃO deve ser ignorado se o programa depende dele para controle de tempo, já que, se ele for desconsiderado, pode ocorrer um loop infinito.

Q4) As informações necessárias seriam os estados da memória e do processador, registradores da CPU, contexto do kernel e a fila de processos prontos, para que o hardware saiba gerenciar os processos eficientemente. O funcionamento desse processo por hardware envolveria um timer fisicamente embutido no sistema computacional, de modo a não depender do timer de interrupção do SO. Também seria necessária uma memória dedicada à tabela de processos, para que eles sejam escalonados no hardware.

Q5) Existe uma pilha para cada thread, pois isso garante isolamento, segurança na execução e permite que várias threads possam sem sobrepor dados umas das outras.

Q6) Nesse cenário, o tempo gasto será de $(n \cdot T)$ segundos, pois é necessário

aguardar a execução das instruções de todos os outros processos que antecederem um específico processo até que o processador possa se dedicar a ele. Assim, cada instrução do processo será executada n vezes mais devagar devido a essa forma especial de escalonamento.

Q7) Se um processo aparecer mais de uma vez, ele terá o dobro de tempo dedicado a ele pelo processador, já que ele terá duas oportunidades de execução. Esse fato pode ser permitido quando se deseja priorizar um processo sobre os outros presentes na fila.

Q8) $E = \text{eficiência da CPU}$ / $S = \text{overhead}$ / $Q = \text{Quantum}$ / $\alpha = \text{arbitrário}$

$\alpha \cdot Q = \alpha \Rightarrow E = \frac{\min(T, \alpha)}{\min(T, \alpha) + S}$

♦ Performance

- q grande \Rightarrow FIFO
- q pequeno $\Rightarrow q$ deve ser grande em relação ao tempo de troca de contexto, ou o overhead será muito alto.

b. $Q \gg T \Rightarrow E = \frac{T}{T+S}$

↓
permitirá que o processo seja executado até T não permitir

d. $Q \leq S \Rightarrow E = \frac{1}{2} \quad 50\%$

↓
metade do tempo
executa, outra metade
troca

c. $S < Q < T \Rightarrow E = \frac{Q}{Q+S}$

↓
 Q menor que T
gera mais trocas

e. $Q \approx 0 \Rightarrow$ overhead altíssimo, eficiência próxima a 0 \rightarrow muita troca de contexto

Q9) Round-Robin

$OV=0$

	3	5	2	1	4
$Q=5$	P_1	P_2	P_3	P_4	P_5
	10	6	2	4	8

\Rightarrow	P_4	P_3	P_1	P_5	P_2
	4	2	10	8	6

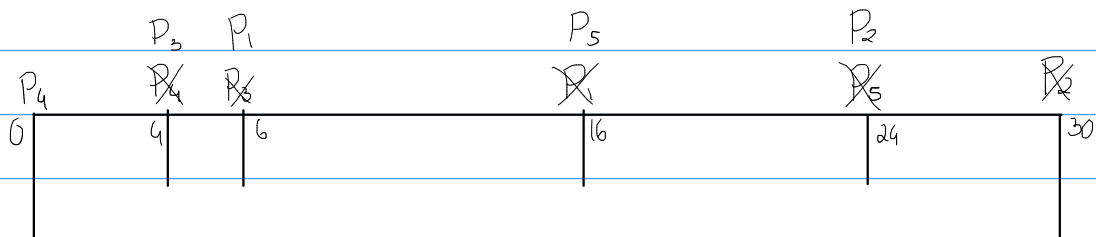
P_1	P_2	P_3	P_4	P_5
26	30	6	4	29

	4	2	5	5	5
\Leftarrow	-	-	5	3	1
	-	-	-	-	-

$(26 + 30 + 6 + 4 + 29) / 5 = 19$

Q9 Escalonamento com Prioridade

P_4	P_3	P_1	P_5	P_2
4	2	10	8	6

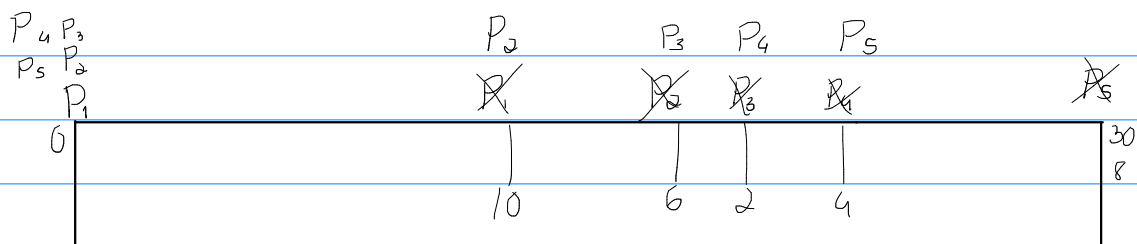


P_1	P_2	P_3	P_4	P_5
16	30	6	4	24

$(4+6+16+24+30) / 5 = 16$

FCFS (10 → 6 → 2 → 4 → 8)

P_1	P_2	P_3	P_4	P_5
10	6	2	4	8



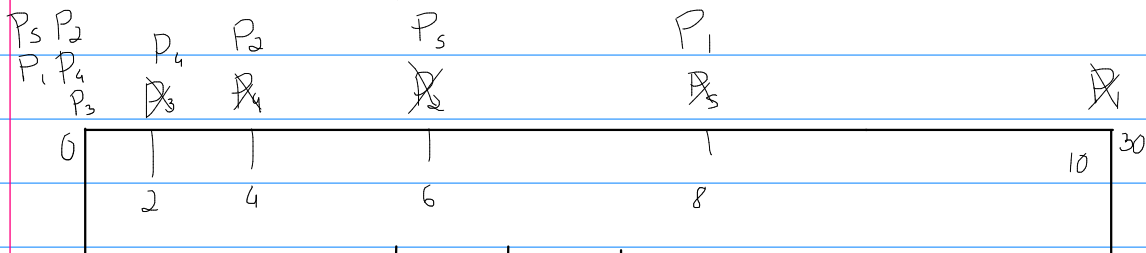
P_1	P_2	P_3	P_4	P_5
10	16	18	22	30

$(10+16+18+22+30) / 5 = 19,2$

SJF (Shortest Job First)

P_1	P_2	P_3	P_4	P_5
10	6	2	4	8

→ $P_3 - P_4 - P_2 - P_5 - P_1$



P_1	P_2	P_3	P_4	P_5
30	12	2	6	20

$(30+12+2+6+20) / 5 = 14$