

### Questão 1.

De forma resumida, uma fórmula que funcionaria em função de  $n$  seria  $n!$ , uma vez que com  $n!$  daria para calcular o número de permutações realizada por  $(n)$ .

### Questão 2.

A diferença entre o escalonamento com preempção e sem preempção está na capacidade do sistema operacional de interromper ou não a execução de um processo em andamento para iniciar a execução de outro processo, uma vez que se ocorrer a interrupção da execução, o processo é preemptivo, já se não ocorrer ele é não preemptivo.

### Questão 3.

As I/O serão favorecidas por conta do pico de CPU curto requerido por eles e os programas limitados por CPU não entrarão em starvation porque os programas limitados por I/O liberarão a CPU.

### Questão 4.

A distinção entre programas limitados por I/O e programas limitados por CPU é crucial para otimizar o desempenho do sistema, uma vez que na maioria dos casos as I/Os não utilizam todo o quantum, diferente da CPU que normalmente consome todo o quantum, desta forma é de grande importância para o escalonador conseguir diferenciar esses dois.

### Questão 5.

A restrição de valores  $<0$  apenas para o root está principalmente ligada a segurança do sistema, uma vez que as prioridades dos processos no Linux são definidas por uma representação numérica que pode ser de  $-20$  até  $+19$  e a prioridade dos processos se vincula à representação numérica, uma vez que ser negativo representa ter maior prioridade, e ser maior ou igual a  $0$  possui prioridade menor. Desta forma o sistema permite que o usuário padrão tenha acesso às tarefas de menor prioridade, e o usuário root pode acessar os processos de maior prioridade.

### Questão 6.

D - por prioridade. Uma vez que por priorizar o processo que possui maior prioridade, pode acontecer de que o processo que possui menor prioridade não consiga terminar seu processo por ser interrompido por um de maior prioridade.