

## **Lista de Exercícios 04**

### **Processos**

Nome: Wescley Júnior Gonçalves Navarro

**Exercício 01)** O comando `ps` é comumente usado em conjunto com as opções `aux` (sem começar com traço) que interpreta essas opções no estilo BSD. Contudo, `ps` também suporta opções que começam com hífen. Tente obter um resultado similar ao estilo BSD usando opções com hífen.

**\$ `ps -ef`**

**Exercício 02)** Um usuário abriu o editor de texto `gedit` pela linha de comando, mas percebeu que a *shell* não devolveu o *prompt* para que ele pudesse dar outros comandos.

**\$ `gedit`**

- a) Sem fechar o editor, o que o usuário pode fazer para liberar o terminal para que possa dar mais comandos?

Ctrl + z ira parar o programa sem fechá-lo.

- b) O usuário percebeu que, ao resolver o item (a), o editor de texto parou de responder. Como ele pode fazer para retomar a execução desse programa e ainda manter o *prompt* da *shell*?

**\$ `bg [Número do processo q pode ser adquirido via 'jobs']`**

**Exercício 03)** Para evitar toda a burocracia do segundo exercício, o usuário poderia inicializar o editor de texto diretamente em plano de fundo. Como ele poderia fazer isso?

**\$ `gedit &`**

**Exercício 04)** O comando `top` é usado para exibir os processos que mais consomem *CPUs*, mas aceita comandos **internamente** (dentro do `top`, após ele ser executado) para executar outras tarefas. Por exemplo, o comando `q` é usado para sair do programa. Indique a sequência de comandos necessários para executar as ações a seguir. Dica: use o manual do `top` para descobrir esses comandos.

- a) Mostrar somente os processos do usuário `aluno`.

Pressionar U, digitar aluno

b) Enviar o sinal HUP para o processo cujo *pid* é 1254.

Pressionar k, digitar o PID 1254, e depois digitar o sinal HUP

**Exercício 05)** Indique **uma situação** em que os seguintes sinais podem ser emitidos por um programa.

a) SIGILL (4)

Caso seja passado uma instrução desconhecida ou que o computador não estava esperando ele pode lançar esse sinal e parar o programa.

b) SIGTRAP (5)

Caso esteja debugando um código e coloque um ponto de parada(breaking point) o programa emite esse sinal avisando q o programa parou propositalmente, sendo usado para debug.

c) SIGABRT (6)

Caso um programa encontre um erro grave ele mandara esse sinal encerrando o programa imediatamente.

d) SIGALRM (14)

Usado por exemplo em funções como sleep, ele lança um alerta caso uma certa condição seja atendida como um determinado tempo tendo passado.