

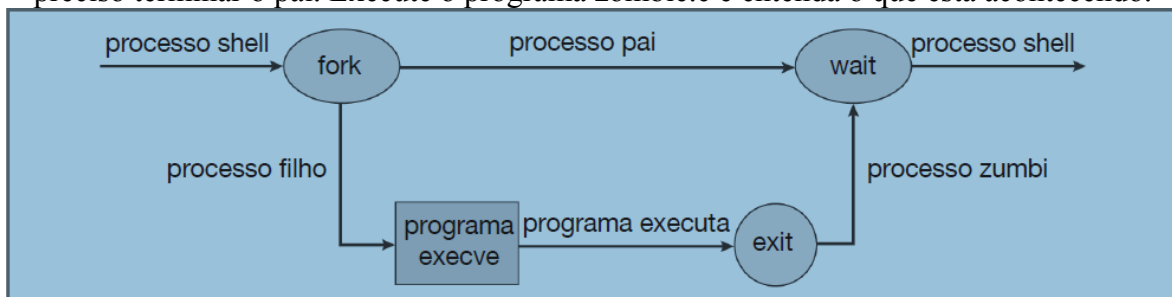
CES-33

Laboratório 1 – Gerência de Processos

O objetivo deste laboratório é a familiarização com gerência de processos no Linux.

Parte 1:

1. Na máquina virtual, baixe o arquivo *Lab1-ShellUnix.c*, compile-o e rode-o passando como parâmetro comandos básicos do Linux como *ls*, *date*, *pwd*, *cat*. Este programa é uma interface de shell: aceita comandos de usuário e então executa cada comando em um processo separado. (Para compilar: *gcc Lab1-ShellUnix.c -o Lab1-Shell.exe*.)
2. Em outro terminal, encontre o seu processo na lista de processos (*ps -aux | grep Lab1*). No prompt de seu shell, chame o navegador Firefox. Abrindo outra janela, encontre seu PID na lista de processos (*ps -aux | grep firefox*). Qual o PID do pai? Qual o PID do filho? Qual o PPID? Veja o PID do filho e a seguir pergunte o PPID dele pelo comando *ps: (ps -p <pid> -o ppid)*. Quem é o avô?
3. Crie processos “órfãos” matando o processo pai através do comando *kill: kill -9 <pid>*. O que acontece com a hierarquia dos processos? Investigue.
4. Abra um terminal e digite o comando *top*. Este comando dá uma visão dinâmica do que está rodando no sistema.
5. Processo zumbi: Se o filho termina a execução, mas o pai não está esperando por ele (*waitpid*), o filho entra em modo zumbi, quando ele não é mais executado, porém ainda ocupa um espaço na tabela de processos. Quando o pai espera a morte do filho com a função *waitpid*, a memória do filho é desalocada e seu espaço na tabela de processos é limpo. Processos zumbis quando exibidos via comando *ps*, tem um Z no campo status do processo. Eles não podem ser terminados com o comando *kill* porque já não existem. É preciso terminar o pai. Execute o programa *zombie.c* e entenda o que está acontecendo.



Altere o programa zombie.c e crie um processo que fica zumbi por apenas 20 segundos. Verifique que na lista de processos ativos nos primeiros 20 segundos o processo é zumbi, e que depois deste tempo o processo não está mais na lista de processos.

Parte 2 para casa:

Programa 1:

Faça um programa em que três processos executam paralelamente as seguintes ações:

- Pai - Imprime os números de 1 a 50, com um intervalo de 2 segundos entre cada número. Após imprimir todos os números, imprime a frase “Processo pai vai morrer”.
- Filho1 - Imprime os números de 100 a 200, com um intervalo de 1 segundo entre cada número. Antes de imprimir os números, imprime a frase “Filho 1 foi criado”, e após imprimir todos os números, imprime a frase “Filho 1 vai morrer”.
- Neto1- (filho do processo Filho1) imprime os números de 300 a 350, com um intervalo de 2 segundos entre cada número. Antes de imprimir os números, imprime a frase “Neto 1 foi criado” Após imprimir todos os números, imprime a frase “Neto 1 vai morrer”.

Em cada printf os processos devem imprimir o seu pid e o pid do seu pai.

Mostre as saídas enfatizando os processos executando em “paralelo”.

Programa 2:

Altere o programa 1 para que primeiro sejam exibidos primeiramente apenas os prints do neto, depois só os do filho e por último só os do pai.

Relatório

Registre sua experiência da seguinte maneira em um único relatório com 2 seções:

- Seção 1 relativa ao Programa 1:
 - Repita o enunciado do problema;
 - Coloque o código desenvolvido;
 - Coloque o resultado dos testes.
 - Discussão: Porque obteve este resultado? Se concluiu algo? Como verificou o paralelismo?
- Seção 2 relativa ao Programa 2:
 - Repita o enunciado do problema;
 - Coloque o código desenvolvido;
 - Coloque o resultado dos testes.
 - Discussão: Como alterou o programa e verificou a sequencia dos números?

Entregue este relato e os 2 programas fontes no dia 26/03 no G Classroom.

Dica do dia:

Pode ser útil para você criar uma pasta compartilhada entre Windows e Unix para troca de arquivos com simplicidade e rapidez. Passos para isso:

- 1) Crie uma pasta no Windows (ex: NossaPasta) e nas configurações da VM escolha esta pasta na opção Pasta Compartilhada;
- 2) No Ubuntu, na opção Dispositivo, escolha Inserir imagem de CD dos adicionais de convidado.
- 3) Crie um diretório no Unix para partilhar. (Ex: /home/aluno/Compartilha)
- 4) Monte: sudo mount -t vboxsf NossaPasta /home/aluno/Compartilha