EASTERN WASHINGTON UNIVERSITY
start something big

Processes and Jobs

Computer Science Department
Eastern Washington University
Yun Tian (Tony) Ph.D.

Last Class

- **find differs from grep command**
  - find command to search files in file system by name, size, permission etc.
  - grep to search specific content in files.
  - Different command layout
    - grep –r 'pattern' directory_name
    - find directory_name –name targetfile –print
  - By default find search folders recursively, but grep does not.

Last Class

- quotes
  - everything inside single quote has literal meaning.
  - special metacharacters $ and ` in double quotes are preserved.
  - backquotes
    - echo “The current time is `date`.”
- **find directoryName –name targetfile –print**
- ln –s ~/cscd240/mylab1.txt mylab_link

Outline for Today

- Processes and Jobs
  - Concept
  - Command of ps
  - background jobs
  - control processes

Concepts

- **What is Process?**
  - A process is a program in execution.
    - Edit files in nano
    - Listening music with iTune.
  - A process consists of resources that the program needs to run.
    - Such as program instructions, the files descriptors, the memory to be used, (Registers, Stack, Heap) IPC tools etc.
  - The OS keeps track of processes by assigning each a number, the process id or PID.

Concepts

- When invoke a system utility or an application program from a shell, one or more 'child' processes are created by the shell.
- An important process that is always present is the **init** process.
  - First process to be created when a UNIX system starts up.
  - Usually has a PID of 1.

ps command

- Show states of processes and their PID
  - ps
    - without arguments, it will list PIDs for procecces owned by the current shell.
  - ps –aux
    - shows full information about all processes on BSD machines.
    - equivalent to ps –fae
  - ps –u user
    - Shows all processes belonged to the user

kill command

- Kill a process with PID
  - Sends different level signals to the running process, requesting it shutdown and exit gracefully. (the SIGTERM signal)
    - This may not always work.
  - kill –l list all signals
    - “kill -19 PID” suspend a running process.
- Force a process to terminate abruptly.
  - **kill -9 processID**

kill command

- Note that, you can only kill processes that belong to you (unless you are the Superuser).
- **killall**
  - Kill processes by name, instead of by pid.
  - sends a signal to all processes associated with a particular command name or program name.

Jobs

- Actually the same thing as processes but restricted to the processes started by a particular shell.
  - Attached to that shell session.
  - If Shell closes, jobs in it terminate.
- Jobs can either be in the **foreground** or the **background**.
  - Only one job in the foreground at any time.
  - The foreground job has control of the shell with which you interact .

Jobs

- **jobs -l**
  - List the active jobs and its pid.

  →[1]+ 1490 Running find / -print 1>output 2>errors &
    - [1] is the job number.
    - 1490 is the process id for the corresponding process.
    - find / -print 1>output 2>errors &
      - Start a job as background job using & at the end.
  - Note that if you have more than one job you can refer to the job as %**n**
    - where **n** is the job number.

Jobs

- For example **fg %3** resumes job number 3 in the foreground.
- Kill jobs has the same syntax as kill with a PID.
  - Assume we run jobs –l we get:
    - [1]+ 1490 Running find / -print 1>output 2>errors &
  - kill -9 %1
    - Kill job 1
  - kill -9 1490
    - 1490 is the pid associated with the job find

Jobs

- The foreground job may be suspended, i.e. temporarily stopped, by pressing the Ctrl-Z key.
- A suspended job can be made to continue running in the foreground or background as needed by typing "fg" or "bg" respectively.
- Ctrl-Z differs from Ctrl-C.
  - Ctrl-C means to terminate a job or process.

Jobs

- Ctrl-C is used to interrupt a job or a process.
  - Equivalent to kill -1 PID
- Three steps to move a **running** job background
  1. suspend fore ground job using Ctrl-z,
  2. run jobs command returns its job id **n.**
  3. **bg %n**

Jobs

- How to find a big job to play with?
  - because most of jobs terminate before you pause them.
    - Not big enough to run long long time.
  - This job never done unless you kill it.
    - **cp /dev/zero /dev/null**

Shell Scripts

- **Definition**
  - A text file containing program statements for the shell command line interpreter.
  - Typically use for
    - file manipulation, automated command execution, printing text.
  - Shell scripts are interpreted.
  - Most Unix system support different type of shell scripting.

Shell Scripts

- **Definition**
  - Most Unix system support different type of shell scripting.
    - Bourne shell (sh), Bash shell(bash), C shell, Korn shell.
  - Created with a text editor.
    - before you run, add the execute permission to it.
  - Executed by invoking the script name at the command line.

To Write Scripts

- **First Line**
  - should begin with a sha-bang **#!**
    - Identifies the file as a shell script.
  - Should also specify the interpreter to be used for the commands in the script.
    - specify as the path to the interpreter:
    - /bin/bash, /usr/bin/python, /usr/bin/perl
  - Example

    #!/bin/bash

To Write Scripts

- **Comments**
  - Single line starts with **#**
  - # also could be placed at the end of line after a command.
- **Variables**
  - VariableName=value
    - No space around =
  - $VariableName is always replaced with the current value of the variable.

To Write Scripts

- **Basic I/O**
  - **echo value**
    - Screen output
- chang_name.sh
- change file name from 2007-09-24-pic.jpg to 24-09-2007-pic.jpb

hello_world.sh

```
#!/bin/bash
#Hello World shell script

string="Hello World"
echo  $string
echo program done!
```

chang_name.sh

```
#!/bin/bash

for fn in ./*.jpg
   do mv $fn `echo $fn | \
   sed -rn 's/([0-9]+)-([0-9]+)-([0-9]+)/\3-\2-\1/p'`
done
```

Summary

- Process and Jobs
- Process is an execution of a program, also means the resources that a process is assigned by the OS.
- jobs -l
- ps –aux
- kill -9 PID
- **Ctrl-z and Ctrl-c**

Summary

- Background and foreground jobs
- jobs command
- #!
- Shell Variable
  - **$NAME**

Next Class

More Scripting and for loops and condition test if.