Salgado Berardo, Joao Antonio 100280950 A9 Jul/16/2017 Mark:_____

Chapter 7: 1, 2, 3, 4, 6, 8

Chapter 7:

1. The parameter $$ expands to the process ID of the current shell, while $! stores the process ID of the last background job.

When you run a script, a new process is created by the current shell, so if you use $$ inside a script, it will store the process ID of the new process running the script, not the process ID of the shell that have spawned the script.

2. Processes are organized in a hierarchy similar to files, with the first process on the top, like the root directory.

All processes have a parent process (except init process). Files also have parents, that go up in the structure until the top (root in case of files and init in case of processes).

Also, a process can have multiple children , like a directory can have multiple files in it. But a child process can have only one parent (same for files).

Finally, attributes are stored in a separated data structure in memory, called the process table. Files have a data structure called inode.

3. Yes. When a user executes a program, a process is created. If two users execute the same program, each running program is a different process, so each one have its own memory allocation, so the Operating System needs to allocate more memory.

4. When a process uses fork to spawn another process, it usually ‘waits’ for the child to terminate and get the exit code (wait system call).

While the first case is more common, the child process can also continue to run the same code as its parent if the exec system call isn’t called.

6. Daemons are system processes created at boot time by init, like web server, mail server, and so on.

They don’t have a terminal attached, so they can’t use the keyboard and monitor for standard output and standard input. Also, a user can’t press <Ctrl-c> to terminate a system process.

You can display daemons with the ps command (for example, ps -e). To identify them, one could use ps -e to see all processes, and the TTY (terminal attached to the process) column will have an “?” instead of a terminal file.

8. Because cd is a built-in program. This means that ‘cd’ is part of the kernel, and no process is created when it is executed (nor a file called cd).