COSC 519 -- Lab 1 -- Sara Ogaz

11.  We need the dot before the executable name because the location of the file is not known by the OS.  This is because the home directory is not part of the PATH environment variable.

12. Adding your home directory to your PATH is NOT a good idea.  It provides a security risk, as someone could easily replace one of your normal executables with something bad in your home directory, which would be run automatically when you tried to use said executable.  Example used in class:  something as simple as "ls" could be replaced, running an unknown process when ls is run.

Tested unix/linux commands:

uptime                    display system history and load

od                    display file in octal, hexadecimal or tex

head                       display from the beginning of the file

tail                          display from the end of the file

diff                        display differences of two files

sort                         sort the file

whereis                  find file

find                        find file and execute commands on it

date                        display date and time

ps                           list information on active processes

jobs                        display running background jobs

printenv                 display values of environment variables

set                          display or set values of environment variables

kill                   senda signal to the process, kill the process

bg                           continue process at background

fg                           move the process to foreground

Control-Z               kill current foreground process, log out of command shell

sleep x                    put the process on hold for x seconds

top                         displays list of most active processes

x | y                        unnamed pipe, the standard output of process x is standard input of  process y

history print command history for current

**cal displays the current month**

**time** **cat** runs a timer

|  |  |
| --- | --- |
| df | Shows the amount of disk space in use |
| du | Shows the amount of disk space used in every dir from current location |

|  |  |
| --- | --- |
| file myfile | Short summary of what type of file myfile is |
| touch | Update access and modification time of a file. |
| wc | Count words, lines, and characters |
| apropos | Locate commands by keyword lookup |
| finger | Print information about logged-in users |

30

|  |  |  |  |
| --- | --- | --- | --- |
| iostat | | Report I/O statistics | |
| script | | Keep script of terminal session | |
| [basename](http://en.wikipedia.org/wiki/Basename) |  | | Return non-directory portion of a pathname; see also dirname |
| [dirname](http://en.wikipedia.org/wiki/Dirname) |  | | Return the directory portion of a pathname; see also basename |

whatis - describe what a command is

stat - status of file (i.e. last access)

**Clear** command clear command clears the screen and puts cursor at beginning of first line.

**Nohup** command.   
nohup command if added in front of any command will continue running the command or process even if you shut down your terminal or close your session to machine. For exmaple, if I want to run a job that takes lot of time and must be run from terminal and is called update\_entries\_tonight .   
nohup update\_entries\_tonight will run the job even if terminal is shut down in middle of this job.

* **last -1 *username*** --- tells you when the user last logged on and off and from where. Without any options, **last** will give you a list of everyone's logins.

tar -zxf archive.tgz this will extract the contents of the archive called archive.tgz. kind of like unzipping a zipfile. \*\*\*