CS308: Large Applications Practicum

Lab 1

This lab covers basic Linux commands and shell scripts. We will be using Bash, which is a popular shell for Linux systems.

A good resource for Bash documentation is here:
http://tldp.org/HOWTO/Bash-Prog-Intro-HOWTO.html

1. **Shell variables.** In the commands below, the $ at the beginning is the shell prompt.
   a) Define a Bash variable called `count` and set it to 3.
      ```
      $ count=3
      ```
   b) Print the value of count. *Use echo*.
      ```
      $ echo $count
      ```
   c) Note the use of the '$'. Type
      ```
      $ echo count
      ```
   d) Increment count by 1. Print the result. *Use double paranthesis.*
      ```
      $ ((c=c+1))
      ```
   e) Bash variables are untyped.
      ```
      $ echo $NAME1
      $ NAME1="Arun Kumar"
      $ echo $NAME1
      $ NAME2="Verma"
      $ NAME3=$NAME1$NAME2
      $ echo $NAME3
      ```

   f) Print the value of `$NAME+$NAME2`

2. **Executing a script.**
   a) Save the above Bash commands into a script file and call it `script1.sh.`
   b) Provide execute permission to owner`. chmod u+x script1.sh`
   c) Run the script: `./script1.sh`
   d) Repeat the above with `NAME1="4", NAME2="6"`

3. The following script adds two numbers.
   ```
   #!/bin/bash
   x=5
   y=10
   ans=$(( x + y ))
   echo "$x + $y = $ans"
   ```

   Modify the script to find the maximum of three numbers. Please look at the documentation for the syntax of `if..then..else`.

4. Create a text file called `movies.list` which lists your five favourite movies.
   a) Determine the number of lines in the file (*use `wc`*)
   b) Determine the number of lines by using `cat` and piping.

5. In the previous question, look at the output of `wc`. What information does it display? How can you print only the number of lines in the file?

6. Use `seq` and `shuf` to generate a random sequence of 20 numbers.

7. Sort the above sequence using `sort`. Use the `-n` flag for numeric sort.

8. Create the following file `marks.dat` which has three columns:

```
$ cat marks.dat
B12 Rajesh 34
B18 Mahesh 75
B19 Arun 55
B10 Vinod 90
B22 Priya 95
B30 Susan 85
```

   Use `shuf` to generate a random permutation of the entries. Now sort based on the marks (use `-k` flag.) Use `head` to print the records with the top three marks.

9. Use `seq` and `shuf` to generate random sequences of length 20, 30 and 100. Redirect these to a single file. How many lines are present? Use `uniq` to remove repeated entries. How many lines are present now?

10. Lookup the use of the command `xargs`.

11. Write a script that prints out the current user and his/her home directory. *Use `whoami` and `pwd`.*

12. Write a script which prints the number of items in the `/etc` directory. *Use backticks.*

13. Write a script that prints the number of directories in the `/etc` directory.

14. Make a Bash script sleep for 5 seconds. *Use sleep.*

15. Given the following C program, write a Bash script to:

a) Check if a given number is prime or not.
b) Determine the time taken by the C program.
c) Read each line in the file `data.txt` and determine if it is prime or not. Look up how to read a file line by line.