60-256 Labs Fall 2014

Lab#1: Introduction to Unix Shell

Part 1: Follow the steps below to create a simple bash shell script.

1. Create a file named `myScript`, and type in the following line:

```
#!/bin/bash

echo Welcome to shell scripting.
myfirstvar=hello
echo $myfirstvar
myfirstvar=($myfirstvar there)
echo $myfirstvar
echo "Today is ‘date‘"
```

2. Modify this file's protection mode by using the `chmod` command to make sure the file can be executed by you. Once it is done, run the script and take note of the output.

3. Now add the following lines to the file

```
echo The program name is: $0
echo "The program name is: $0"
echo 'The program name is: $0'

echo The first argument is: $1
echo The second argument is: $2
echo All the arguments are: $*
echo The total number of argument is: $#
```

   and run the script with some arguments as shown below. Take note of the output produced.

```
myScript these are some arguments
```

4. Use man/online help to modify your script to have today's date printed as

```
Today is Monday September 22, 2014
```

   or

```
Today is Wednesday September 24, 2014
```

   for Wednesday group.

Note: You may launch the script with `-xv` options as

```
%bash -xv your_csh_script
```

to get some details of how the script is being executed, and thus help the debugging process.

Part 2:

- Write a bash shell script that will display the total number of files, within a given directory, whose names start with **b** and conist of 5 characters. The name of the directory is given as command line argument at the shell prompt.

  A sample run:

  ```
  >testscript /bin/
        4
  >testscript /usr/include/
        1
  ```

  There are 4 matches in the `/bin` directory and 1 in the `/usr/include` directory.

- Modify your script so that the starting letter (**b**) becomes the second parameter passed to the shell script at the shell prompt.

  The above sample runs become:

  ```
  >testscript /bin b
        4
  >testscript /bin f
        3
  ```