**QUESTION 2 (10 marks) –**(script filename to be submitted: question2.scr)

Using the command **script** again (**question2.scr**), record the results of the following tasks. Each task should be implemented with one or two commands (including wild cards).

1. Change your working directory to **/bin/**; print your current working directory;

cd /bin/

pwd

(b) Count the number of files in the current working directory (should be **/bin/**);

ls -a1 | wc -l

(c) Count the number of files whose name begins with **a**, **l**, or **i** in **/bin/**;

ls -a1 | grep '^[ali]' | wc -l

(d) Count the number of files whose name does not begin with **a**, **l**, or **i** in **/bin/**;

ls -a1 | grep -v '^[ali]' | wc -l

(e) Show the files in **/bin/** whose names have a **c** as the second letter;

ls -a1 | grep '^.c'

(f) Show the files in **/bin/** whose names are of length 6;

ls -a1 | grep '^.\{6\}$'

(g) Count the number of files in **/bin/** whose names are of length 4;

ls -a1 | grep '^.\{4\}$' | wc -l

(h) Count the number of files in **/bin/** whose name begins with an uppercase letter;

ls -a1 | grep '^[A-Z].\*' | wc -l

### QUESTION 3 (5 marks) – (script filename to be submitted: question3.scr)

Using the command **script** again, record the results of the following tasks. Each task should be implemented with one command (including wild cards).

1. Create a directory called **public\_html**;

mkdir public\_html

1. Display the permissions of **public\_html**;

ls -ld public\_html

1. Remove your read permission and the group execute permission on **public\_html**;

chmod u-r,g-x public\_html

1. Try to show the contents of **public\_html**;

ls public\_html

(e) Change the permission of **public\_html** such that you can read, write and execute, but all other users can only execute.

chmod u+rwx,g=x,o=x public\_html

### QUESTION 4 (10 marks) – (script filename to be submitted: question4.scr)

Using the command **script** again, record the results of the following tasks. Each task should be implemented with one or two commands (including wild cards).

1. Create a directory called **MyDir1** and make it as your working directory;

mkdir Mydir1

cd Mydir1

(b) Create files with the following names in the current directory (the contents of these files do not matter):

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| intro | n1 | book2 | page1 | page3 | page3b |
| n2 | book3 | Book3 | Page1a | Page3a |  |
|  |  |  |  |  |  |

touch intro n1 book2 page1 page3 page3b n2 book3 Book3 Page1a Page3a

1. List all files whose names begin with **book** or **Book**;

ls -1 | grep '^[Bb]ook'

1. List the files whose name ends with a number only.

ls -1 | grep '[0-9]$'

### QUESTION 5 (10 marks) – (script filename to be submitted: question5.scr)

Using the command **script** again, record the results of the following tasks. Each task should be implemented with one or two one-line commands (including wild cards).

1. Display a list of all the hidden files in the current working directory, sorted in reverse lexicographical (alphabetical) order of file name.

ls -ldr .?\*

1. Use the **finger** command to find yourself on **obelix**.

finger ywu582

1. Using the **echo** command, create a one-line file in your home directory called **.plan** outlining some of your plans for today.

echo 'Finish assignment, go to class, eat lunch, sleep' >.plan

1. Change the permissions to make it readable to the world.

chmod u+r,g+r,o+r .plan

1. Use the **finger** command on yourself again.
2. finger ywu582