**ASSIGNMENT 1**

**PROGRAM STATEMENT:**

Write a shell script to show the current location and list all the files which starts with letter ‘D’. Store the output file to a file ass\_1\_output.txt.

**ALGORITHM:**

OUTPUT: 1.Currrent location of the user

2. List of files and folder whose name starts with D

Steps:

1. Start
2. Write "You are currently in"
3. Get current working directory by using command: pwd
4. Write "List of files start with letter 'D'"
5. Get detailed list of files start with Letter ‘D’ using command: ls – l D\*
6. End.

**PROGRAM CODE:**

echo "You are currently in"

pwd

echo "List of files start with letter 'D'"

ls -l D\*

**OUTPUT:**

You are currently in

/mnt/d/Tasneem/shell\_script

List of files start with Letter ‘D’

-rwxrwxrwx 1 tasneemkhan tasneemkhan 11 Nov 3 00:31 D1.txt

-rwxrwxrwx 1 tasneemkhan tasneemkhan 3135536 Jul 25 2019 Digital\_Design\_5th\_Edition\_M.\_Morris\_Man.pdf

**DISCUSSION:**

1. In the terminal we have to give permission to the ‘.sh’ file for read, write and execute using chmod command.
2. To store the output to the file (ass\_1\_output.txt), in the terminal use command : ./assignment\_1.sh > ass\_1\_output.txt
3. Command pwd stands for present working directory.

**Assignment 2**

**PROGRAM STATEMENT:**

Write a shell script to convert the lower case letters of a file into upper case.

**ALGORITHM:**

INPUT: Enter the file name.

OUTPUT: All the letters of the file converted into Upper case.

Steps:

1. Start
2. Input the file name and store it in a variable ‘f1’.
3. Convert file from lower case to upper using command: tr “[a-z]” “[A-Z]”
4. Concatenate file using command: cat $f
5. End.

**PROGRAM CODE:**

echo "Enter a file name: "

read f1

echo -e "Initially file content \n"

cat $f1

echo -e "\nChanged file content \n"

cat $f1 | tr "[a-z]" "[A-Z]"

**OUTPUT:**

Enter a file name:

ass\_file1.txt

Initally file content

this is my second assignment

to convert all the lower case letters to upper case letters

Changed file content

THIS IS MY SECOND ASSIGNMENT

TO COVERT ALL THE LOWER CASE LETTERS TO UPPER CASE LETTERS

**DISCUSSION:**

1. In the terminal we have to give permission to the ‘.sh’ file for read, write and execute using chmod command. Using command :chmod 777 assignment\_2.sh
2. To use a variable in required places use ‘$’ sign in front of the variable name.

**ASSIGNMENT 3**

**PROGRAM STATEMENT:**

Write the shell script to count the frequency of each word stored in a file. Read the file name as an input.

**ALGORITHM:**

INPUT: Enter the file name.

OUTPUT: Frequency of each word of the file.

Steps:

1. Input file name and store it in a variable “file”.
2. Concatenate the file content to console using command: cat
3. Pass the file contents to tr command using pipe
4. Convert all spaces into new line character
5. Sort the lines lexicographically using command : sort.
6. Remove all the duplicate lines using: uniq
7. Count the lines based on number of occurrences using –c option
8. Print the result
9. End

**PROGRAM CODE:**

echo "Enter the file name"

read file

echo -e "File content:\n"

cat $file

echo -e "Frecuency of each word\n"

cat $file |tr ' ' '\n' |sort |uniq -c

**OUTPUT:**

Enter the file name

ass\_3.txt

File content:

I am using ubuntu.

I am writing my third assignment

In this script I am going too count the frequency of each word.

count occurrences of word in file in linux

shell script to count number of words in a file

count occurrences of all words in file linux

Frecuency of each word

3 I

1 In

1 a

1 all

3 am

1 assignment

4 count

1 each

3 file

1 frequency

1 going

4 in

2 linux

1 my

1 number

2 occurrences

4 of

2 script

1 shell

1 the

1 third

1 this

1 to

1 too

1 ubuntu.

1 using

1 word

1 word.

2 words

1 writing

**DISCUSSION:**

1. The command echo -e enable interpretation of backslash escapes, which helps to print a newline, tab, etc.
2. The command uniq –c filters out the repeated lines and counts the frequency of the word.

**ASSIGNMENT 4**

**PROGRAM STATEMENT:**

Write a shell script to swap the value of two numbers without using any variables.

**ALGORITHM:**

INPUT: Enter the two numbers

OUTPUT: Swap the two numbers

Steps:

1. Input both number and store it in variable ‘a’ and ‘b’.
2. Print the initial values of ‘a’ and ‘b’

[swapping]

1. Set a := a + b
2. Set b := a – b
3. Set a := a – b
4. Print the swapped values of ‘a’ and ‘b’.
5. End

**PROGRAM CODE:**

echo "Enter the first number"

read a

echo "Enter the second number"

read b

echo –e "Initial values of: \n a = $a \n b = $b"

a=$((a+b))

b=$((a - b))

a=$((a-b))

echo "Values after swapping \na = $a \nb = $b"

**OUTPUT:**

Enter the first number:

2

Enter the second number

4

Initial values of:

a = 2

b = 4

Values after swapping

a = 4

b = 2

**ASSIGNMENT 5**

**PROGRAM STATEMENT:**

Write a shell script for any base to any base conversion. The input and output bases should be taken as input.

**ALGORITHM:**

INPUT: 1. Enter the number to change the base

2. Enter the input base and output base.

OUTPUT: Conversion of a number to the given output base.

Steps:

1. Start
2. Input the input base and store the value in variable ‘inbase’
3. Input the output base and store the value in variable ‘outbase’
4. Input a number to be converted and store in the variable ‘num’
5. Use command bc for the calculation
6. Set obase := outbase
7. Set obase := inbase
8. Run bc and display output
9. End

**PROGRAM CODE:**

echo "Enter the input base"

read inbase

echo "Enter the output base"

read outbase

echo "Enter a number: "

read num

echo "The value of $num is: "

echo "obase= $outbase; ibase= $inbase; $num" | bc

**OUTPUT:**

**Case 1:**

Enter the input base

10

Enter the output base

2

Enter a number:

15

The value of 15 is:

1111

**Case 2:**

Enter the input base

2

Enter the output base

10

Enter a number:

100

The value of 100 is:

4

**Case 3:**

Enter the input base

10

Enter the output base

8

Enter a number:

10

The value of 10 is:

12

**Case 4:**

Enter the input base

8

Enter the output base

10

Enter a number:

18

The value of 18 is:

15

**Case 5:**

Enter the input base

10

Enter the output base

16

Enter a number:

14

The value of 14 is:

E

**Case 6:**

Enter the input base

16

Enter the output base

10

Enter a number:

A

The value of A is:

10

**DISCUSSION:**

1. The command bc use as an arbitrary precise calculator language.
2. To convert input base and output base we the use command ibase and obase respectively, to make the calculation more convenient first change the output base then input base.
3. Echo command is a very useful command, it can display text, calculate, which make it very reliable command.

**ASSIGNMENT 6**

**PROGRAM STATEMENT:**

Write a shell script to count the number of users currently logged into the system.

**ALGORITHM:**

OUTPUT: Gives the number of users currently logged in.

Steps:

1. Start
2. Write “Number of users currently logged in:”
3. Get the number of user logged in by using command: who –q.
4. End

**PROGRAM CODE:**

echo "Number of users currently logged in: "

who –q

**OUTPUT:**

Number of users currently logged in:

# users=1

**DISCUSSION:**

1. Linux is a multiuser system means multiple users can access system resources like memory application programs at same time.
2. The command ‘who’ use to show who is logged in, ‘-q’ counts the user logged in.