Name: Brijesh Rameshbhai Rohit

Admission NO.: U19CS009

OS-LAB-ASSIGNMENT-03

(1) Shell script Program to accept a character and check whether it is an

Lower case alphabet

Upper case alphabet

A digit

Special symbol

Vowel

Using case control structure

```
  GNU nano 4.8                          q1.sh
#question 1:

echo -e "Enter character: \c"
read c
case "$c" in
        "a"|"e"|"i"|"o"|"u") echo "A LOWERCASE VOWEL"
        ;;
        "A"|"E"|"I"|"O"|"U") echo "A UPPERCASE VOWEL"
        ;;
        [[:lower:]]) case "$c" in
                *)echo "LOWERCASE CONSONANT"
                ;;
        esac
        ;;
        [[:upper:]]) case "$c" in
                *) echo "UPPERCASE CONSONANT"
                ;;
        esac
        ;;
        [[:digit:]]) echo "NUMERIC DIGIT"
        ;;
        [[:punct:]]) echo "SPECIAL CHARACTER"
        ;;
        *) echo "OTHER CHARACTER"
        ;;
esac
```

```
brijesh@brijesh-GF75-Thin-9SCSR:~/Documents/os/OS-ASSIGN-02$ ./q1.sh
Enter character: A
A UPPERCASE VOWEL
brijesh@brijesh-GF75-Thin-9SCSR:~/Documents/os/OS-ASSIGN-02$ ./q1.sh
Enter character: a
A LOWERCASE VOWEL
brijesh@brijesh-GF75-Thin-9SCSR:~/Documents/os/OS-ASSIGN-02$ ./q1.sh
Enter character: C
UPPERCASE CONSONANT
brijesh@brijesh-GF75-Thin-9SCSR:~/Documents/os/OS-ASSIGN-02$ ./q1.sh
Enter character: c
LOWERCASE CONSONANT
brijesh@brijesh-GF75-Thin-9SCSR:~/Documents/os/OS-ASSIGN-02$ ./q1.sh
Enter character: 1
NUMERIC DIGIT
brijesh@brijesh-GF75-Thin-9SCSR:~/Documents/os/OS-ASSIGN-02$ ./q1.sh
Enter character: @
SPECIAL CHARACTER
brijesh@brijesh-GF75-Thin-9SCSR:~/Documents/os/OS-ASSIGN-02$
```

(2) Using case esac structure

Find the number of users logged into the system

Print the calendar for current year

Print the date

```
  GNU nano 4.8                          q2.sh
#quesiton 2

echo -e "1 : NUMBER OF USERS \n2 : CALENDAR \n3 : DATE"
read op
case "$op" in
        1) who --count | grep users
        ;;
        2) cal
        ;;
        3) date
        ;;
        *) echo -e "INCORRECT INPUT!"
        ;;
esac
```

```
brijesh@brijesh-GF75-Thin-9SCSR:~/Documents/os/OS-ASSIGN-02$ ./q2.sh
1 : NUMBER OF USERS
2 : CALENDAR
3 : DATE
q
INCORRECT INPUT!
brijesh@brijesh-GF75-Thin-9SCSR:~/Documents/os/OS-ASSIGN-02$ ./q2.sh
1 : NUMBER OF USERS
2 : CALENDAR
3 : DATE
1
# users=1
brijesh@brijesh-GF75-Thin-9SCSR:~/Documents/os/OS-ASSIGN-02$ ./q2.sh
1 : NUMBER OF USERS
2 : CALENDAR
3 : DATE
2
    August 2021
Su Mo Tu We Th Fr Sa
 1  2  3  4  5  6  7
 8  9 10 11 12 13 14
15 16 17 18 19 20 21
22 23 24 25 26 27 28
29 30 31

brijesh@brijesh-GF75-Thin-9SCSR:~/Documents/os/OS-ASSIGN-02$ ./q2.sh
1 : NUMBER OF USERS
2 : CALENDAR
3 : DATE
3
Tuesday 17 August 2021 02:50:58 PM IST
```

(3) Shell Script Program to check whether given file is a directory or not.

```
  GNU nano 4.8                          q3.sh
#question 3

echo -e "SEARCH FILE BY NAME : \c"
read filename
if [ -e $filename ]
then
        echo "FILE EXISTS"
else
        echo "FILE DOES NOT EXISTS"
fi
```

```
brijesh@brijesh-GF75-Thin-9SCSR:~/Documents/os/OS-ASSIGN-02$ ls
c1.png     o1.png  q11.sh  q14.sh  q1.sh  q4.sh  q7.sh
c2.png     o2.png  q12.sh  q15.sh  q2.sh  q5.sh  q8.sh
file1.txt  q10.sh  q13.sh  q16.sh  q3.sh  q6.sh  q9.sh
brijesh@brijesh-GF75-Thin-9SCSR:~/Documents/os/OS-ASSIGN-02$ ./q3.sh
SEARCH FILE BY NAME : file1.txt
FILE EXISTS
brijesh@brijesh-GF75-Thin-9SCSR:~/Documents/os/OS-ASSIGN-02$ ./q3.sh
SEARCH FILE BY NAME : abcd.png
FILE DOES NOT EXISTS
```

(4) Shell Script Program to Count number of files in a Directory.

```
  GNU nano 4.8                          q4.sh
#question 4

if [ -d "$@" ]
then
        echo "NUMBER OF FILES IS : $(find "$@" -type f | wc -l)"
        echo "NUMBER OF DIRECTORIES IS : $(find "$@" -type d | wc -l)"
else
        echo -e "ERROR\!, No FILES OF DIRECTORY FOUND\!"
        exit 1
fi
```

```
brijesh@brijesh-GF75-Thin-9SCSR:~/Documents/os/OS-ASSIGN-02$ ls
c1.png  c4.png     o2.png  q11.sh  q14.sh  q1.sh  q4.sh  q7.sh
c2.png  file1.txt  o3.png  q12.sh  q15.sh  q2.sh  q5.sh  q8.sh
c3.png  o1.png     q10.sh  q13.sh  q16.sh  q3.sh  q6.sh  q9.sh
brijesh@brijesh-GF75-Thin-9SCSR:~/Documents/os/OS-ASSIGN-02$ ./q4.sh
NUMBER OF FILES IS : 25
NUMBER OF DIRECTORIES IS : 1
```

(5) Shell Script Program to copy contents of one file to another.

```
  GNU nano 4.8                        q5.sh
#question 5

echo -e "ENTER ORIGINAL FILENAME : \c"
read original
echo -e "ENTER NEW FILENAME : \c"
read new
echo -e  "\n\nCONTENT BEFORE COPYING : "
echo -e "\nCONTENT OF ORIGINAL FILE :"
cat $original
echo -e "\nCONTENT OF NEW FILE :"
if [ -e $new ]
then
        cat $new
else
        echo -e "NEW FILE DOES NOT EXIST!"
fi
cat $original >> $new
echo -e  "\n\nCONTENT AFTER COPYING : "
echo -e "\nCONTENT OF ORIGINAL FILE :"
cat $original
echo -e "\nCONTENT OF NEW FILE :"
cat $new
echo ""
```

```
brijesh@brijesh-GF75-Thin-9SCSR:~/Documents/os/OS-ASSIGN-02$ ls
c1.png  c4.png     o2.png  oldfile.txt  q12.sh  q15.sh  q2.sh  q5.sh  q8.sh
c2.png  file1.txt  o3.png  q10.sh       q13.sh  q16.sh  q3.sh  q6.sh  q9.sh
c3.png  o1.png     o4.png  q11.sh       q14.sh  q1.sh   q4.sh  q7.sh
brijesh@brijesh-GF75-Thin-9SCSR:~/Documents/os/OS-ASSIGN-02$ cat oldfile.txt
hello there!
my name is brijesh rohit
brijesh@brijesh-GF75-Thin-9SCSR:~/Documents/os/OS-ASSIGN-02$ ./q5.sh
ENTER ORIGINAL FILENAME : oldfile.txt
ENTER NEW FILENAME : newfile.txt


CONTENT BEFORE COPYING :

CONTENT OF ORIGINAL FILE :
hello there!
my name is brijesh rohit

CONTENT OF NEW FILE :
NEW FILE DOES NOT EXIST!


CONTENT AFTER COPYING :

CONTENT OF ORIGINAL FILE :
hello there!
my name is brijesh rohit

CONTENT OF NEW FILE :
hello there!
my name is brijesh rohit

brijesh@brijesh-GF75-Thin-9SCSR:~/Documents/os/OS-ASSIGN-02$ cat newfile.txt
hello there!
my name is brijesh rohit
```

(6) Write a shell script to add two numbers supplied by user and supplied as command line argument.

```
  GNU nano 4.8                          q6.sh
#question 6

if [ $# -ne 2 ]
then
        echo -e "INVALID NUMBER OF ARGUMENTS"
else
        echo "SUM : `expr $1 + $2`"
fi
```

```
brijesh@brijesh-GF75-Thin-9SCSR:~/Documents/os/OS-ASSIGN-02$ ./q6.sh 10 12
SUM : 22
brijesh@brijesh-GF75-Thin-9SCSR:~/Documents/os/OS-ASSIGN-02$ ./q6.sh 10 12 13
INVALID NUMBER OF ARGUMENTS
brijesh@brijesh-GF75-Thin-9SCSR:~/Documents/os/OS-ASSIGN-02$ ./q6.sh 10
INVALID NUMBER OF ARGUMENTS
```

(7) Write a shell script to find out biggest number form given three numbers. Numbers are supplied by command line argument.

```
  GNU nano 4.8                          q7.sh
#question 7

if [ $# == 3 ]
then
        echo -e "GREATEST OF FROM $1, $2 AND $3 IS : \c"
        if [ $1 -ge $2 ] && [ $1 -ge $3 ]
        then
                echo $1
        elif [ $2 -ge $3 ]
        then
                echo $2
        else
                echo $3
        fi
else
        echo "ERROR! INCORRECT NUMBER OF ARGUMENTS"
fi
```

```
brijesh@brijesh-GF75-Thin-9SCSR:~/Documents/os/OS-ASSIGN-02$ ./q7.sh 10 12
13
GREATEST OF FROM 10, 12 AND 13 IS : 13
brijesh@brijesh-GF75-Thin-9SCSR:~/Documents/os/OS-ASSIGN-02$ ./q6.sh 10 12
SUM : 22
brijesh@brijesh-GF75-Thin-9SCSR:~/Documents/os/OS-ASSIGN-02$ ./q7.sh 10 12
ERROR! INCORRECT NUMBER OF ARGUMENTS
brijesh@brijesh-GF75-Thin-9SCSR:~/Documents/os/OS-ASSIGN-02$ ./q7.sh 10 12
13 15
ERROR! INCORRECT NUMBER OF ARGUMENTS
```

(8) Implement simple calculator. Numbers are supplied by command line argument.

```
  GNU nano 4.8                          q8.sh
#question 8

if [ $# == 2 ]
then
        echo -e "INPUT AS FOLLOWS"
        echo -e "1 : ADD\n2 : SUBTRACT\n3 : MULTIPLY\n4 : DIVIDE"
        read op
        if [ $2 -eq 0 -a $op -eq 4 ]
        then
                echo "DIVISION NOT POSSIBLE"
        else
                case "$op" in
                        1) echo "$1 + $2 = `expr $1 + $2`"
                        ;;
                        2) echo "$1 - $2 = `expr $1 - $2`"
                        ;;
                        3) echo "$1 * $2 = `expr $1 \* $2`"
                        ;;
                        4) echo "$1 / $2 = `expr $1 / $2`"
                        ;;
                esac
        fi
else
        echo -e "ERROR\! INCORRECT NUMBER OF ARGUMENTS"
fi
```

```
brijesh@brijesh-GF75-Thin-9SCSR:~/Documents/os/OS-ASSIGN-02$ ./q8.sh
ERROR\! INCORRECT NUMBER OF ARGUMENTS
brijesh@brijesh-GF75-Thin-9SCSR:~/Documents/os/OS-ASSIGN-02$ ./q8.sh 10 12
INPUT AS FOLLOWS
1 : ADD
2 : SUBTRACT
3 : MULTIPLY
4 : DIVIDE
1
10 + 12 = 22
brijesh@brijesh-GF75-Thin-9SCSR:~/Documents/os/OS-ASSIGN-02$ ./q8.sh 10 12
INPUT AS FOLLOWS
1 : ADD
2 : SUBTRACT
3 : MULTIPLY
4 : DIVIDE
2
10 - 12 = -2
brijesh@brijesh-GF75-Thin-9SCSR:~/Documents/os/OS-ASSIGN-02$ ./q8.sh 10 12
INPUT AS FOLLOWS
1 : ADD
2 : SUBTRACT
3 : MULTIPLY
4 : DIVIDE
3
10 * 12 = 120
brijesh@brijesh-GF75-Thin-9SCSR:~/Documents/os/OS-ASSIGN-02$ ./q8.sh 10 0
INPUT AS FOLLOWS
1 : ADD
2 : SUBTRACT
3 : MULTIPLY
4 : DIVIDE
4
DIVISION NOT POSSIBLE
brijesh@brijesh-GF75-Thin-9SCSR:~/Documents/os/OS-ASSIGN-02$ ./q8.sh 10 1
INPUT AS FOLLOWS
1 : ADD
2 : SUBTRACT
3 : MULTIPLY
4 : DIVIDE
4
10 / 1 = 10
```

(9) Write a shell script to print numbers in descending order using while loop.

```
  GNU nano 4.8                          q9.sh
#question 9

echo -e "ENTER A NUMBER : \c"
read num
echo -e "THE DECREASING ORDER IS AS : \c"
while [ $num -ge 1 ]
do
        echo -e "$num \c"
        num=`expr $num - 1`
done
echo ""
```

```
brijesh@brijesh-GF75-Thin-9SCSR:~/Documents/os/OS-ASSIGN-02$ ./q9.sh
ENTER A NUMBER : 100
THE DECREASING ORDER IS AS : 100 99 98 97 96 95 94 93 92 91 90 89 88 87 86
85 84 83 82 81 80 79 78 77 76 75 74 73 72 71 70 69 68 67 66 65 64 63 62 61
60 59 58 57 56 55 54 53 52 51 50 49 48 47 46 45 44 43 42 41 40 39 38 37 36
35 34 33 32 31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11
10 9 8 7 6 5 4 3 2 1
```

(10) Write a shell script to create a simple calculator using switch-case statement.

```
  GNU nano 4.8                          q10.sh
#question 10

echo -e "ENTER 1st NUMBER : \c"
read num1
echo -e "ENTER 2nd NUMBER : \c"
read num2
echo -e "INPUT AS FOLLOW :"
echo -e "1 : ADD\n2 : SUBTRACT\n3 : MULTIPLY\n4 : DIVIDE"
read op
if [ $num2 -eq 0 -a $op -eq 4 ]
then
        echo "DIVISION NOT POSSIBLE"
else
        case "$op" in
                1) echo -e "$num1 + $num2 = `expr $num1 + $num2`"
                ;;
                2) echo -e "$num1 - $num2 = `expr $num1 - $num2`"
                ;;
                3) echo -e "$num1 * $num2 = `expr $num1 \* $num2`"
                ;;
                4) echo -e "$num1 / $num2 = `expr $num1 / $num2`"
                ;;
        esac
fi
```

```
brijesh@brijesh-GF75-Thin-9SCSR:~/Documents/os/OS-ASSIGN-02$ ./q10.sh
ENTER 1st NUMBER : 12
ENTER 2nd NUMBER : 10
INPUT AS FOLLOW :
1 : ADD
2 : SUBTRACT
3 : MULTIPLY
4 : DIVIDE
1
12 + 10 = 22
brijesh@brijesh-GF75-Thin-9SCSR:~/Documents/os/OS-ASSIGN-02$ ./q10.sh
ENTER 1st NUMBER : 12
ENTER 2nd NUMBER : 0
INPUT AS FOLLOW :
1 : ADD
2 : SUBTRACT
3 : MULTIPLY
4 : DIVIDE
4
DIVISION NOT POSSIBLE
brijesh@brijesh-GF75-Thin-9SCSR:~/Documents/os/OS-ASSIGN-02$ ./q10.sh
ENTER 1st NUMBER : 12
ENTER 2nd NUMBER : 0
INPUT AS FOLLOW :
1 : ADD
2 : SUBTRACT
3 : MULTIPLY
4 : DIVIDE
3
12 * 0 = 0
```

(11) Write a shell script to print given number in reverse order.

```
 GNU nano 4.8                          q11.sh
#question 11

echo -e "ENTER A NUMBER : \c"
read num
echo -e "REVERSE USING  REV : \c"
echo $num | rev
d=0
rev=0
while [ $num -gt 0 ]
do
        digit=`expr $num % 10`
        rev=`expr $rev \* 10`
        rev=`expr $rev + $digit`
        num=`expr $num / 10`
done
echo -e "REVERSE USING WHILE : \c"
echo "$rev"
```

```
brijesh@brijesh-GF75-Thin-9SCSR:~/Documents/os/OS-ASSIGN-02$ ./q11.sh
ENTER A NUMBER : 123456789
REVERSE USING  REV : 987654321
REVERSE USING WHILE : 987654321
brijesh@brijesh-GF75-Thin-9SCSR:~/Documents/os/OS-ASSIGN-02$ ./q11.sh
ENTER A NUMBER : 0001230103124
REVERSE USING  REV : 4213010321000
REVERSE USING WHILE : 4213010321
```

(12) Write a shell script to print sum of all digits of a given number.

```
  GNU nano 4.8                          q12.sh
#question 12

echo -e "ENTER A NUMBER : \c"
read num
digit=0
sum=0
while [ $num -gt 0 ]
do
        digit=`expr $num % 10`
        sum=`expr $sum + $digit`
        num=`expr $num / 10`
done
echo "SUM OF DIGITS OF NUMBER YOU ENTERED IS : $sum"
```

```
brijesh@brijesh-GF75-Thin-9SCSR:~/Documents/os/OS-ASSIGN-02$ ./q12.sh
ENTER A NUMBER : 123456789
SUM OF DIGITS OF NUMBER YOU ENTERED IS : 45
brijesh@brijesh-GF75-Thin-9SCSR:~/Documents/os/OS-ASSIGN-02$ ./q12.sh
ENTER A NUMBER : 09807605403201
SUM OF DIGITS OF NUMBER YOU ENTERED IS : 45
```

(13) Find the factorial value of given input number.

```
  GNU nano 4.8                          q13.sh
#question 13

echo -e "ENTER NUMBER : \c"
read n
if [ $n -lt 0 ]
then
        echo -e "PLEASE ENTER A NON-NEGATIVE INTEGER!"
else
        num=$n
        fact=1
        while [ $n -gt 1 ]
        do
                fact=`expr $fact \* $n`
                n=`expr $n - 1`
        done
        echo -e "FACTORIAL OF $num = $fact"
fi
```

```
brijesh@brijesh-GF75-Thin-9SCSR:~/Documents/os/OS-ASSIGN-02$ ./q13.sh
ENTER NUMBER : 0
FACTORIAL OF 0 = 1
brijesh@brijesh-GF75-Thin-9SCSR:~/Documents/os/OS-ASSIGN-02$ ./q13.sh
ENTER NUMBER : 10
FACTORIAL OF 10 = 3628800
brijesh@brijesh-GF75-Thin-9SCSR:~/Documents/os/OS-ASSIGN-02$ ./q13.sh
ENTER NUMBER : 5
FACTORIAL OF 5 = 120
brijesh@brijesh-GF75-Thin-9SCSR:~/Documents/os/OS-ASSIGN-02$ ./q13.sh
ENTER NUMBER : -1
PLEASE ENTER A NON-NEGATIVE INTEGER!
```

(14) Generate and display Fibonacci series.

```
  GNU nano 4.8                        q14.sh
#question 14

echo -e "ENTER NUMBER OF TERMS FOR FIBONACCI : \c"
read n
if [ $n -lt 1 ]
then
        echo -e "PLEASE ENTER A POSITIVE INTEGER"
else
        x=0
        y=1
        i=0
        while [ $i -lt $n ]
        do
                echo -e "$x \c"
                nxt=`expr $x + $y`
                x=$y
                y=$nxt
                i=`expr $i + 1`
        done
        echo ""
fi
```

```
brijesh@brijesh-GF75-Thin-9SCSR:~/Documents/os/OS-ASSIGN-02$ ./q14.sh
ENTER NUMBER OF TERMS FOR FIBONACCI : -12
PLEASE ENTER A POSITIVE INTEGER
brijesh@brijesh-GF75-Thin-9SCSR:~/Documents/os/OS-ASSIGN-02$ ./q14.sh
ENTER NUMBER OF TERMS FOR FIBONACCI : 10
0 1 1 2 3 5 8 13 21 34
brijesh@brijesh-GF75-Thin-9SCSR:~/Documents/os/OS-ASSIGN-02$ ./q14.sh
ENTER NUMBER OF TERMS FOR FIBONACCI : 30
0 1 1 2 3 5 8 13 21 34 55 89 144 233 377 610 987 1597 2584 4181 6765 10946
17711 28657 46368 75025 121393 196418 317811 514229
brijesh@brijesh-GF75-Thin-9SCSR:~/Documents/os/OS-ASSIGN-02$ ./q14.sh
ENTER NUMBER OF TERMS FOR FIBONACCI : 50
0 1 1 2 3 5 8 13 21 34 55 89 144 233 377 610 987 1597 2584 4181 6765 10946
17711 28657 46368 75025 121393 196418 317811 514229 832040 1346269 2178309
3524578 5702887 9227465 14930352 24157817 39088169 63245986 102334155 16558
0141 267914296 433494437 701408733 1134903170 1836311903 2971215073 4807526
976 7778742049
```

(15) Display all even numbers within given range.

```
  GNU nano 4.8                          q15.sh
#question 15

echo -e "ENTER LOWER LIMIT : \c"
read ll
echo -e "ENTER UPPER LIMIT : \c"
read ul
if [ $ll -ge $ul ]
then
        echo "INVALID INPUT!"
fi
rem=`expr $ll % 2`
if [ $rem == 1 ]
then
        ll=`expr $ll + 1`
else
        ll=`expr $ll + 2`
fi
while [ $ll -lt $ul ]
do
        echo -e "$ll \c"
        ll=`expr $ll + 2`
done
echo ""
```

```
brijesh@brijesh-GF75-Thin-9SCSR:~/Documents/os/OS-ASSIGN-02$ ./q15.sh
ENTER LOWER LIMIT : 100
ENTER UPPER LIMIT : 200
102 104 106 108 110 112 114 116 118 120 122 124 126 128 130 132 134 136 138
 140 142 144 146 148 150 152 154 156 158 160 162 164 166 168 170 172 174 17
6 178 180 182 184 186 188 190 192 194 196 198
brijesh@brijesh-GF75-Thin-9SCSR:~/Documents/os/OS-ASSIGN-02$ ./q15.sh
ENTER LOWER LIMIT : 101
ENTER UPPER LIMIT : 200
102 104 106 108 110 112 114 116 118 120 122 124 126 128 130 132 134 136 138
 140 142 144 146 148 150 152 154 156 158 160 162 164 166 168 170 172 174 17
6 178 180 182 184 186 188 190 192 194 196 198
brijesh@brijesh-GF75-Thin-9SCSR:~/Documents/os/OS-ASSIGN-02$ ./q15.sh
ENTER LOWER LIMIT : 100
ENTER UPPER LIMIT : 201
102 104 106 108 110 112 114 116 118 120 122 124 126 128 130 132 134 136 138
 140 142 144 146 148 150 152 154 156 158 160 162 164 166 168 170 172 174 17
6 178 180 182 184 186 188 190 192 194 196 198 200
brijesh@brijesh-GF75-Thin-9SCSR:~/Documents/os/OS-ASSIGN-02$ ./q15.sh
ENTER LOWER LIMIT : 101
ENTER UPPER LIMIT : 201
102 104 106 108 110 112 114 116 118 120 122 124 126 128 130 132 134 136 138
 140 142 144 146 148 150 152 154 156 158 160 162 164 166 168 170 172 174 17
6 178 180 182 184 186 188 190 192 194 196 198 200
```

(16) Find out number of characters, words and lines from a given file.

```
  GNU nano 4.8                        q16.sh
#question 16

echo -e "SEARCH FILE NAME : \c"
read filename
if [ -e $filename ]
then
        lines=`wc -l < $filename`
        words=`wc -w < $filename`
        chars=`wc -m < $filename`
echo "No. of Characters: $chars"
echo "No. of Words: $words"
echo "No. of Lines: $lines"
else
echo "FILE DOES NOT EXIST"
fi
```

```
brijesh@brijesh-GF75-Thin-9SCSR:~/Documents/os/OS-ASSIGN-02$ ls
c10.png  c3.png     newfile.txt  o2.png  oldfile.txt  q1.sh  q8.sh.save
c11.png  c4.png     o10.png      o3.png  q10.sh       q2.sh  q9.sh
c12.png  c5.png     o11.png      o4.png  q11.sh       q3.sh
c13.png  c6.png     o12.png      o5.png  q12.sh       q4.sh
c14.png  c7.png     o13.png      o6.png  q13.sh       q5.sh
c15.png  c8.png     o14.png      o7.png  q14.sh       q6.sh
c1.png   c9.png     o15.png      o8.png  q15.sh       q7.sh
c2.png   file1.txt  o1.png       o9.png  q16.sh       q8.sh
brijesh@brijesh-GF75-Thin-9SCSR:~/Documents/os/OS-ASSIGN-02$ cat file1.txt
hey there!

how do you do?

hope you are fine @

buddy!
love the way you lie



end of session!
brijesh@brijesh-GF75-Thin-9SCSR:~/Documents/os/OS-ASSIGN-02$ ./q16.sh
SEARCH FILE NAME : file1.txt
No. of Characters: 96
No. of Words: 20
No. of Lines: 12
```