Name:- PATEL NIYATI

Roll No:- 565

Division:- E

1. Write a shell script to find greatest amongst three number

---

#1/bin/bash

This shell script finds the greatest three numbers.

echo "Enter the first number:" read numl echo

"Enter the second number:" read num2 echo "Enter

the third number:" read num3 if [Snum1 -gt Snum2]

&& [$num1 -gt Snum3 J; then greatest=Snum1 elif [

Snum2 -gt Snum1] && [Snum2 -gt Snum3 J; then

greatest=Snum2

else

greatest=Snum3

fi

echo "The greatest number is Sgreatest."

```
03/09/2023 11:16.19 /home/mobaxterm #!/bin/bash

03/09/2023 11:21.12 /home/mobaxterm # Get the three numbeRS.

03/09/2023 11:21.34 /home/mobaxterm echo "Enter the first number:"
Enter the first number:

03/09/2023 11:21.49 /home/mobaxterm read num1
54

03/09/2023 11:22.05 /home/mobaxterm echo "Enter the second number:"
Enter the second number:

03/09/2023 11:22.26 /home/mobaxterm read num2
67

03/09/2023 11:22.31 /home/mobaxterm echo "Enter the third number:"
Enter the third number:

03/09/2023 11:22.39 /home/mobaxterm read num3
72

03/09/2023 11:22.47 /home/mobaxterm if [ $num1 -gt $num2 ] && [ $num1 -gt $num3 ]; then
>   greatest=$num1
> elif [ $num2 -gt $num1 ] && [ $num2 -gt $num3 ]; then
>   greatest=$num2
> else
>   greatest=$num3
> fi

03/09/2023 11:23.21 /home/mobaxterm echo "The greatest number is $greatest."
The greatest number is 72.
```

Name:- PATEL NIYATI

Roll No:- 565

Division:- E

2. Write a shell script to find all prime numbers in a given range.

#1/bin/bash #

prime numbers.

echo "Enter the starting number:" read

start

echo "Enter the ending number:" read

end

Find all prime numbers in the given range.

for ((i=$start; i<=$end; i++)); do is_prime=1

for ((j=2; j<=§i/2; j++)); do if

[$((i%j)) -eq 0]; then

is_prime=0 break

fi

done if [Sis_prime -eq 1

]; then echo "$i"

fi

done

```
03/09/2023  11:33.42  /home/mobaxterm  #!/bin/bash

03/09/2023  11:33.46  /home/mobaxterm  # prime number.

03/09/2023  11:33.48  /home/mobaxterm  echo "Enter the starting number:"
Enter the starting number:

03/09/2023  11:33.52  /home/mobaxterm  read start
1

03/09/2023  11:34.01  /home/mobaxterm  echo "Enter the ending number:"
Enter the ending number:

03/09/2023  11:34.15  /home/mobaxterm  read end
54

03/09/2023  11:34.25  /home/mobaxterm  for ((i=$start; i<=$end; i++)); do
>   is_prime=1
>   for ((j=2; j<=$i/2; j++)); do
>     if [ $((i%j)) -eq 0 ]; then
>       is_prime=0
>       break
>     fi
>   done
>   if [ $is_prime -eq 1 ]; then
>     echo "$i"
>   fi
> done
1
2
3
5
7
11
13
17
19
23
29
31
37
41
43
47
53
```

Name:- PATEL NIYATI

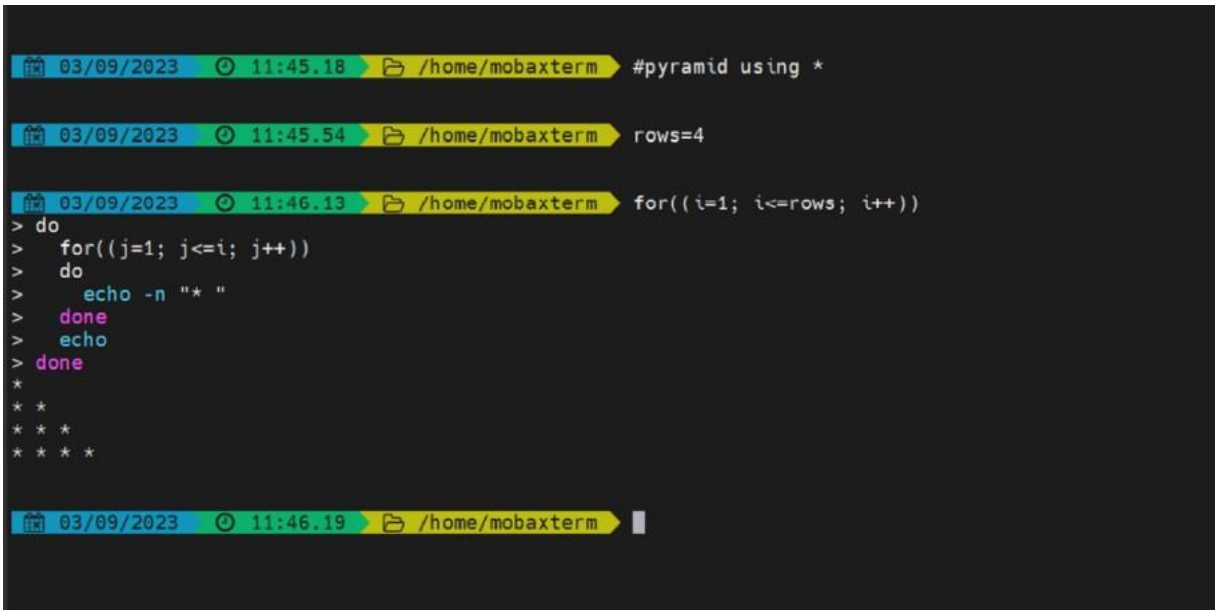
Roll No:- 565

Division:- E

3. Write a shell script to draw the following pattern.

```
#pyramid using *  rows=4
for((i=1; i<=rows; i++))
do  for((j=1; j<=i; j++))
do  echo-n"*"  done
echo

done
```



4)Write a shell script to find sum of digits of a number.

```
# Sum of digits of a number.
echo "Enter a number:"
read number  sum=0
while [ Snumber -gt 0]; do
digit=$((number %10))  sum=$((sum
+ digit))  number=$((number / 10))
done

# Print the sum of digits.  echo
"The sum of digits is $sum."
```

Name:- PATEL NIYATI

Roll No:- 565

Division:- E

```
03/09/2023  11:51.00  /home/mobaxterm  # Sum of digits of a number.

03/09/2023  11:51.20  /home/mobaxterm  echo "Enter a number:"
Enter a number:

03/09/2023  11:51.27  /home/mobaxterm  read number
54

03/09/2023  11:51.37  /home/mobaxterm  sum=0

03/09/2023  11:51.46  /home/mobaxterm  while [ $number -gt 0 ]; do
>   digit=$((number % 10))
>   sum=$((sum + digit))
>   number=$((number / 10))
> done

03/09/2023  11:51.58  /home/mobaxterm  echo "The sum of digits is $sum."
The sum of digits is 9.

03/09/2023  11:52.06  /home/mobaxterm
```

5)Write a shell script to print fibonacci series upto entered value.

Program for Fibonacci N=6

a=0 b=1 echo "The Fibonacci

series is : " for ((i=0; i<N; i++))

do echo-n"$a"

fn=$((a + b))

a=$b b=$fn

done

```
03/09/2023  12:08.45  /home/mobaxterm

03/09/2023  12:08.51  /home/mobaxterm  # Program for Fibonacci

03/09/2023  12:08.58  /home/mobaxterm  N=54

03/09/2023  12:09.02  /home/mobaxterm  a=0

03/09/2023  12:09.05  /home/mobaxterm  b=1

03/09/2023  12:09.09  /home/mobaxterm  echo "fibonacci series is :"
fibonacci series is :

03/09/2023  12:09.20  /home/mobaxterm  for (( i=0; i<N; i++ ))
> do
>     echo -n "$a "
>     fn=$((a + b))
>     a=$b
>     b=$fn
> done
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 3524
578 5702887 9227465 14930352 24157817 39088169 63245986 102334155 165580141 267914296 433494437 701408733 1134903170 1836311903 2971215073 4807526976 7778
742049 12586269025 20365011074 32951280099 53316291173

03/09/2023  12:09.28  /home/mobaxterm
```

SDJ INTERNATIONAL COLLEGE

Name:- PATEL NIYATI

Roll No:- 565

Division:- E

6)Write a menu driven shell script which accepts a basic amount as an input and displays the following options.

a. Dearness allowance (90% of basic)

b. Provident Fund F (12% of basic) c. House Rent Allowance (20% of basic + DA)

d. Income tax deducted (5% of basic + DA + HRA)

e. Take home salary (basic + DA + HRA—IT)

```
# 90% of basic
calculate_da() {  echo "Dearness Allowance: $(bc <<<
"scale=2; 0.90 * $hasic")"
}
#12% of basic  calculate_pf() {  echo "Provident Fund:
$(bc <<< "scale=2; 0.12 * Sbasic")"
}
#20% of basic + DA  calculate_hra() {  echo "House Rent Allowance: $(bc <<<
"scale=2; 0.20 * ($hasic + $(calculate_da))")"
}
# 5% of basic + DA + HRA
calculate_it() {  echo "Income Tax Deducted: $(bc <<< "scale=2; 0.05 * (Sbasic + $(calculate_da) +
$(calculate_hra))")"
}
#basic+ DA+ HRA-IT  calculate_take_home_salary() {  echo "Take Home Salary: $(bc <<< "scale=2; $basic +
$(calculate_da) + $(calculate_hra) - $(calculate_it)")"  while true; do
echo "Menu:"  echo "a. Dearness allowance"
echo "b. Provident Fund"  echo "c. House Rent
Allowance"  echo "d. Income tax deducted”
echo "e. Take home salary"  echo "f. Exit" read -p
"Enter your choice (a/b/c/d/e/f): " choice  case
"$choice" in
a) read -p "Enter basic amount: " basic; calculate_da ;;
b) read -p "Enter basic amount: " basic; calculate_pf ;;
c) read -p "Enter basic amount: " basic; calculate_hra ;;
```

Name:- PATEL NIYATI

Roll No:- 565

Division:- E

d) read -p "Enter basic amount: " basic; calculate_it ;;

e) read -p "Enter basic amount: " basic; calculate_take_home_salary ;;

f) echo "Exiting the script. Goodbye!"; exit 0 ;;

*) echo "Invalid option. Please choose a valid option (a/b/c/d/e/f)."; ;;

esac done

```
Enter your choice (a/b/c/d/e/f): a
Enter basic amount: 54
Dearness Allowance: 48.60
Menu:
a. Dearness allowance
b. Provident Fund
c. House Rent Allowance
d. Income tax deducted
e. Take home salary
f. Exit
Enter your choice (a/b/c/d/e/f): b
Enter basic amount: 54
Provident Fund: 6.48
Menu:
a. Dearness allowance
b. Provident Fund
c. House Rent Allowance
d. Income tax deducted
e. Take home salary
f. Exit
```

7. Write a shell script to find file permissions of user, group and others file_to_check="example.txt" # Check if the file exists if [- "Sfile_to_check"]; then

Use stat to retrieve the file permissions user_permissions=$(stat

-c "%A" "$file_to_check") group_permissions=$(stat -c "%a"

"$file_to_check") other_permissions=$(stat -c "%A"

"$file_to_check" | cut -c 7-)

Display the permissions echo "File:

$file_to_check" echo "User Permissions:

Suser_permissions" echo "Group Permissions:

$group_permissions" echo "Others

Permissions: Sother_permissions”

else

echo "File not found: $file_to_check"

fi

Name:- PATEL NIYATI

Roll No:- 565

Division:- E

```
13/09/2023  14:42.08  /home/mobaxterm  file_to_check="example.txt"

13/09/2023  14:42.40  /home/mobaxterm

13/09/2023  14:42.40  /home/mobaxterm  # Check if the file exists

13/09/2023  14:42.40  /home/mobaxterm  if [ -e "$file_to_check" ]; then
>     # Use stat to retrieve the file permissions
>     user_permissions=$(stat -c "%A" "$file_to_check")
>     group_permissions=$(stat -c "%a" "$file_to_check")
>     other_permissions=$(stat -c "%A" "$file_to_check" | cut -c 7-)
>
>     # Display the permissions
>     echo "File: $file_to_check"
>     echo "User Permissions: $user_permissions"
>     echo "Group Permissions: $group_permissions"
>     echo "Others Permissions: $other_permissions"
> else
>     echo "File not found: $file_to_check"
> fi
File: example.txt
User Permissions: -rw-r--r--
Group Permissions: 644
Others Permissions: -r--

13/09/2023  14:42.41  /home/mobaxterm
```

8. Write a shell script that accepts two files are identical or not # Check if the number of arguments is not equal to 2

if ["$#" -ne 2]; then echo

"Usage: $0 filel file2" exit

1

fi

file1="$1" file2="$2"

Check if both files exist if[1-e"S$file1"]

|| [!-e"$file2"]; then echo "One or both

files do not exist."

exit 1

fi

Compare the files using the cmp command

if cmp -s "Sfile1l" "$file2"; then echo "The files

$file1 and $file2 are identical." else echo "The

files $filel and $file2 are not identical."

Fi

Name:- PATEL NIYATI

Roll No:- 565

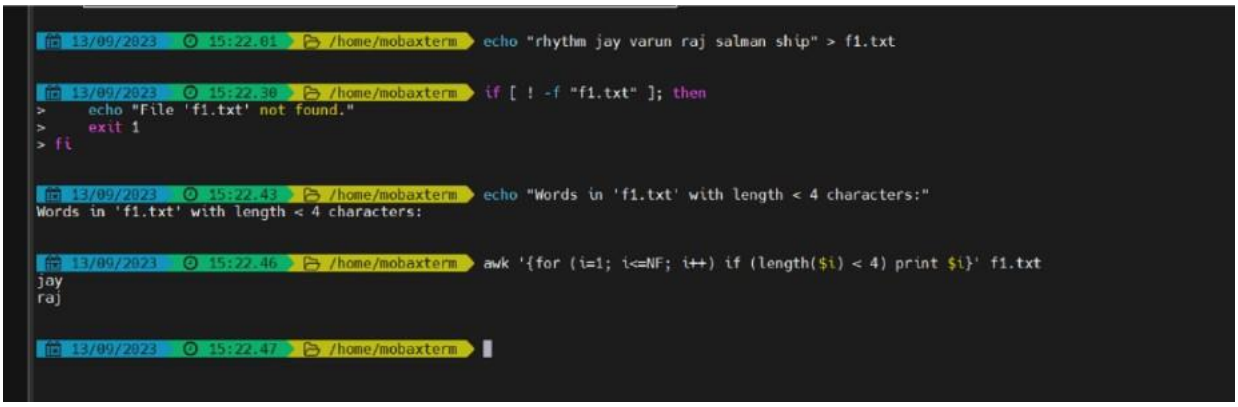
Division:- E

9)Write a shell script to display all the words, having length <4 characters, of a file f1.txt echo "rhythm jay varun cat dog fish" > f1.txt if [1-f "fLtxt"]; then echo "File 'f1.txt' not found." exit 1

fi

echo "Words in 'f1.txt' with length < 4 characters:"

awk '{for (i=1; i<=NF; i++) if (length(Si) < 4) print Si} f1.txt



Write a shell script to find total number of files and total number of directories in current working directory. file_count=0 dir_count=0 for item in *; do if [-f "Sitem"]; then

Increment the file count if it's a regular file

((file_count++)) elif [-

d "Sitem" |; then

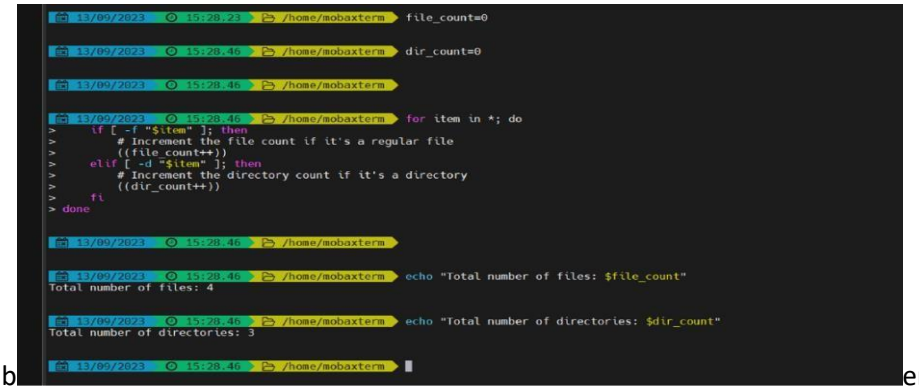
Increment the directory count if it's a directory

((dir_count++))

fi

done echo "Total number of files:

Sfile_count"



echo "Total numb er of directories: $dir_count"

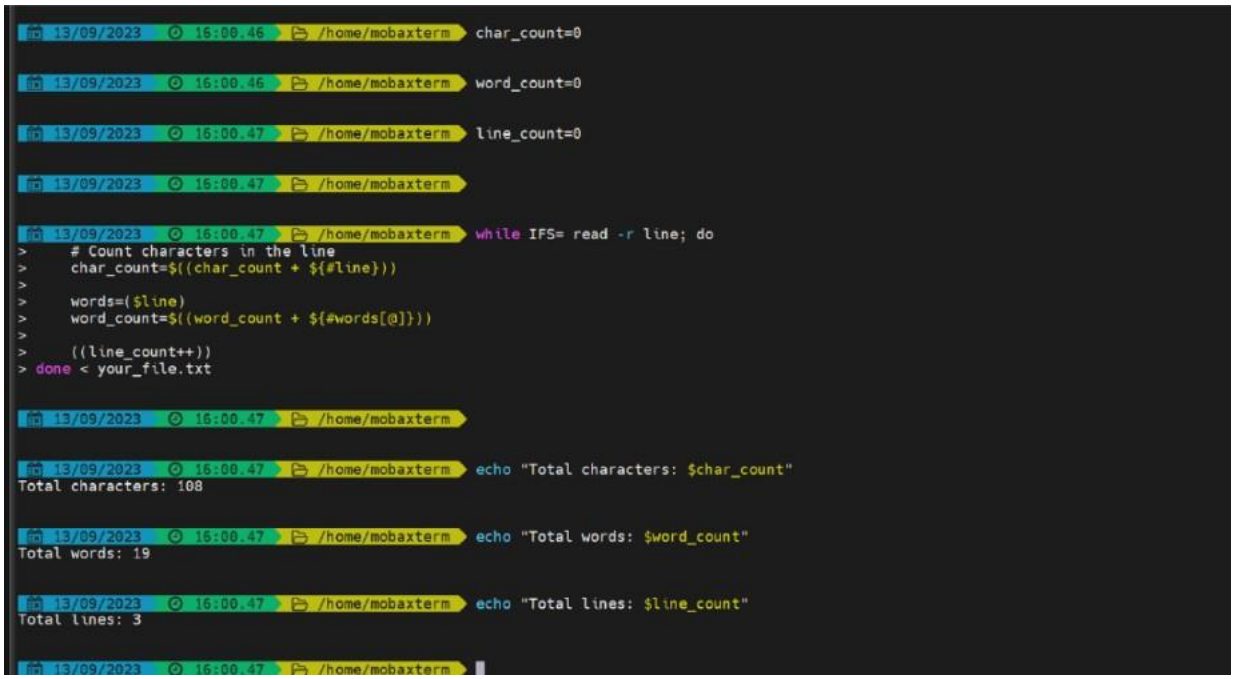
Name:- PATEL NIYATI

Roll No:- 565

Division:- E

9)Write a shell script to find total number of characters, words and lines of a file. (Do not use wc command.

```
cat <<EOL > your_file.txt
This is a sample file.
It contains multiple lines of text.
Counting characters, words, and lines in this file.
EOL  char_count=0  word_count=0
line_count=0  while IFS= read -r line; do
char_count=5$((char_count + ${#line}))
words=(Sline)  word_count=$((word_count +
${#words[@]}))
((line_count++))  done < your_file.txt
echo "Total characters: Schar_count"
echo "Total words: Sword_count"
echo "Total lines: $line_count"
```



Name:- PATEL NIYATI

Roll No:- 565

Division:- E

12. Write a shell script which accepts a username and check the entered user is currently logged in or not.

```
read -p "Enter a username: " username

# Check if the username is logged in  if

who | grep -wq "Susername"; then

echo "Susername is currently logged

in."

else  echo "Susername is not currently

logged in."

fi
```

13)Write a shell script to find total number of occurrences of SDJIC in given file. (Please provide necessary validations)

```
# Prompt the user for the filename

read -p "Enter the filename: " filename

# Check if the file exists  if [ 1 -f

"Sfilename" ]; then  # If the file

doesn't exist, create it  touch

"$filename"  echo "File

'$filename' created."

fi

# Count occurrences of "SDJIC" in the file  occurrences=$(grep

-o -w "SDJIC" "Sfilename" | wc -1)

# Display the result

echo "Total occurrences of 'SDJIC' in 'Sfilename': Soccurrences"
```

Name:- PATEL NIYATI

Roll No:- 565

Division:- E

```
13/09/2023  16:13.54  /home/mobaxterm  # Prompt the user for the filename

13/09/2023  16:13.54  /home/mobaxterm  read -p "Enter the filename: " filename
Enter the filename: RHYTHM

13/09/2023  16:13.58  /home/mobaxterm  # Check if the file exists

13/09/2023  16:14.06  /home/mobaxterm  if [ ! -f "$filename" ]; then
>     # If the file doesn't exist, create it
>     touch "$filename"
>     echo "File '$filename' created."
> fi
RHYTHM: File name too long 'SDJIC' in '$filename': $occurrences"
RHYTHM' created.rrences of 'SDJIC' in '$filename': $occurrences"

13/09/2023  16:14.06  /home/mobaxterm

13/09/2023  16:14.06  /home/mobaxterm  # Count occurrences of "SDJIC" in the file

13/09/2023  16:14.06  /home/mobaxterm  occurrences=$(grep -o -w "SDJIC" "$filename" | wc -l)
RHYTHM: File name too long 'SDJIC' in '$filename': $occurrences"

13/09/2023  16:14.06  /home/mobaxterm

13/09/2023  16:14.06  /home/mobaxterm  # Display the result

13/09/2023  16:14.07  /home/mobaxterm  echo "Total occurrences of 'SDJIC' in '$filename': $occurrences"
RHYTHM': 0l occurrences of 'SDJIC' in '$filename': $occurrences"
```

14. Write a shell script which accepts filename as input and reverse individual words from it. (Please provide necessary validations)

```
echo "i am rhythm 054" >input.txt

filename="input.txt"  if [ | -f

"$filename" ]; then  echo "File

'$filename' not found."  exit 1

fi

# Create an empty output file

output_file="reversed_Sfilename"  touch

"Soutput_file"

# Read the file, reverse individual words, and write to the output file

while read -r -a words; do  for word in "${words[@]}"; do

reversed_word=$(echo "Sword" | rev)  echo -n "$reversed_word "

>> "Soutput_file"  done  echo "" >> "Soutput_file"  done <

"$filename"

echo "Individual words reversed and saved to 'Soutput_file'."
```

Name:- PATEL NIYATI

Roll No:- 565

Division:- E

```
13/09/2023  16:38.20  /home/mobaxterm  echo "i am rhythm 054" > input.txt

13/09/2023  16:38.29  /home/mobaxterm  filename="input.txt"

13/09/2023  16:39.23  /home/mobaxterm

13/09/2023  16:39.23  /home/mobaxterm  if [ ! -f "$filename" ]; then
>     echo "File '$filename' not found."
>     exit 1
> fi

13/09/2023  16:39.23  /home/mobaxterm

13/09/2023  16:39.23  /home/mobaxterm  output_file="reversed_$filename"

13/09/2023  16:39.23  /home/mobaxterm  touch "$output_file"

13/09/2023  16:39.23  /home/mobaxterm

13/09/2023  16:39.24  /home/mobaxterm  while read -r -a words; do
>     for word in "${words[@]}"; do
>         reversed_word=$(echo "$word" | rev)
>         echo -n "$reversed_word " >> "$output_file"
>     done
>     echo "" >> "$output_file"
> done < "$filename"

13/09/2023  16:39.25  /home/mobaxterm

13/09/2023  16:39.25  /home/mobaxterm  echo "Individual words reversed and saved to '$output_file'."
Individual words reversed and saved to 'reversed_input.txt'.

13/09/2023  16:39.25  /home/mobaxterm  cat reversed_input.txt
i ma mhtyhr 450
i ma mhtyhr 450

13/09/2023  16:39.38  /home/mobaxterm
```

15. Write a shell script to display all the lines from a file (11.txt), which starts with text “unix”. (not case sensitive)

---

echo "Unix is an operating system." > 11.txt echo

"UNIX is versatile." >> 11.txt echo "Linux is built

on Unix principles." >> 11.txt echo "UNIX-like

systems include macOS." >> 11.txt echo

"Windows is not Unix-based." >> 11.txt

filename="11.txt" if [| -f "Sfilename"]; then

echo "File '$filename’ not found."

exit 1

fi

grep -i "Munix" "Sfilename"

Name:- PATEL NIYATI

Roll No:- 565

Division:- E

```
13/09/2023  16:44.14  /home/mobaxterm  echo "Unix is an operating system." > 11.txt

13/09/2023  16:44.15  /home/mobaxterm  echo "UNIX is versatile." >> 11.txt

13/09/2023  16:44.16  /home/mobaxterm  echo "Linux is built on Unix principles." >> 11.txt

13/09/2023  16:44.16  /home/mobaxterm  echo "UNIX-like systems include macOS." >> 11.txt

13/09/2023  16:44.16  /home/mobaxterm  echo "Windows is not Unix-based." >> 11.txt

13/09/2023  16:44.16  /home/mobaxterm  filename="11.txt"

13/09/2023  16:44.32  /home/mobaxterm

13/09/2023  16:44.33  /home/mobaxterm  if [ ! -f "$filename" ]; then
>     echo "File '$filename' not found."
>     exit 1
> fi

13/09/2023  16:44.33  /home/mobaxterm

13/09/2023  16:44.33  /home/mobaxterm  grep -i "^unix" "$filename"
Unix is an operating system.
UNIX is versatile.
UNIX-like systems include macOS.
```

16. Write grep command to perform following actions:

a. Count number of blank lines in file f1.txt

b. print all lines containing sdjic

c. print the lines that start with sdjic.

d. Search the files in CCROGRAMS directory which has the string "include"

e. print lines having exactly 50 characters in file f1.txt

f. Count number of blank lines in file f1.txt

g. Display lines having at least one characters in file f1.txt

h. Display lines having sdjic text in any case in file f1.txt

i. Display line of file f1.txt having exactly 3 characters

j. Display lines of file f1.txt which begin with any alphabet

k. Display lines whose last word is “UNIX” in file f1.txt

l. Display filenames having last character as digit [0-9]

m. Display list of filenames that only consist digits

n. Display line of file f1.txt which only consist digits

o. Display lines of file f1.txt which only consist capital alphabets

p. Search all lines in file f1.txt which ends with “”

Ans. a) grep -c 'A$' fl.txt

Name:- PATEL NIYATI

Roll No:- 565

Division:- E

#a. Count number of blank lines in file f1.txt

```
18/09/2023 05:39.37 /home/mobaxterm grep -c '^$' f1.txt
1

18/09/2023 05:40.11 /home/mobaxterm
```

Ans. b) grep 'sdjic' f1.txt

b. Print all lines containing "sdjic" in file f1.txt

```
18/09/2023 05:40.11 /home/mobaxterm grep 'sdjic' f1.txt
Another line with sdjic.

18/09/2023 05:42.49 /home/mobaxterm
```

Ans. c) grep 'Asdjic' f1.txt

c. Print the lines that start with "sdjic" in file f1.txt

```
18/09/2023 05:47.37 /home/mobaxterm grep '^sdjic' f1.txt
sdjic ans 3.
```

Ans. d) grep -rl 'include' CPROGRAMS/

d. Search for files in the CPROGRAMS directory that have the string "include"

Ans. E) grep -E 'A.{50}$' fL.txt

e. Print lines having exactly 50 characters in file f1.txt

Ans. F) grep - 'AS$' fl.txt

#1{. Count the number of blank lines in file f1.txt

```
19/09/2023 05:39.15 /home/mobaxterm grep -c '^$' f1.txt
0
```

Ans. G) grep "' fl.txt

g. Display lines having at least one character in file f1.txt

```
19/09/2023 05:39.15 /home/mobaxterm grep '.' f1.txt
This is a sample line.
This line contains sdjic.
Another line with sdjic at the start.
Line with exactly 50 characters: 12345678901234567890123456789012345678901234567890
A line with 3 characters.
123
ABC
The last word in this line is UNIX.
This is a blank line.
```

Ans. H) grep -i 'sdjic' fl.txt

h. Display lines containing "sdjic" (case-insensitive) in file f1.txt

Name:- PATEL NIYATI

Roll No:- 565

Division:- E

```
19/09/2023 05:39.16 /home/mobaxterm grep -i 'sdjic' f1.txt
This line contains sdjic.
Another line with sdjic at the start.
```

Ans. 1) grep 'A...$" f1.txt

#i. Display lines of file f1.txt having exactly 3 characters

```
19/09/2023 05:39.16 /home/mobaxterm grep '^...$' f1.txt
123
ABC
```

Ans. J) grep 'A[A-Za-z]' fl.txt

. Display lines of file f1.txt which begin with any alphabet

```
19/09/2023 05:39.16 /home/mobaxterm grep '^[A-Za-z]' f1.txt
This is a sample line.
This line contains sdjic.
Another line with sdjic at the start.
Line with exactly 50 characters: 12345678901234567890123456789012345678901234567890
A line with 3 characters.
ABC
The last word in this line is UNIX.
This is a blank line.
```

Ans. K) grep "\<UNIXS' f1.txt

k. Display lines whose last word is "UNIX" in file f1.txt

Ans. L) Is | grep '[0-9]$'

#1. Display filenames in the current directory with the last character as a digit [0-9]

Ans. M) Is | grep 'A[0-9]*$'

m. Display a list of filenames in the current directory that consist only of digits

Ans. N) grep 'M[0-9]*$" f1.txt

n. Display lines of file f1.txt that consist only of digits grep

'~[0-9]*$' f1.txt

```
19/09/2023 05:39.18 /home/mobaxterm grep '^[0-9]*$' f1.txt
123
```

Ans. O) grep 'M[A-Z]*$' fl.txt

0. Display lines of file f1.txt that consist only of capital alphabets

```
19/09/2023 05:39.18 /home/mobaxterm grep '^[A-Z]*$' f1.txt
ABC
```

Ans. P) grep '\.$' fl.txt

p. Search for all lines in file f1.txt that end with "."

Name:- PATEL NIYATI

Roll No:- 565

Division:- E

```
19/09/2023  05:39.19  /home/mobaxterm  grep '\.$' f1.txt
This is a sample line.
This line contains sdjic.
Another line with sdjic at the start.
A line with 3 characters.
The last word in this line is UNIX.
This is a blank line.
```

17. Write sed command to perform following tasks

a. To print only the last line of f1.txt

b. To print line number 1-3, 6-7 and 10 of f1.txt

c. To print lines beginning with SDJIC of f1.txt

d. Print three lines starting from the fourth line of f1.txt

e. Print all blank lines of file f1.txt

f. Print lines having either of “sdjic” or “sdjyc”

g. Lines beginning with either alphabet or digit

h. To insert a line “additional line” before every line

i. To replace every occurrence of | with : of first three lines

j. To replace every occurrence of “|” with “:” of every line

k. To remove all the lines having word “fail” from file f1.txt (delete command)

---

a) sed-n'Sp'fl.txt

Task a. Print only the last line of f1.txt

```
19/09/2023  05:59.08  /home/mobaxterm  sed -n '$p' f1.txt
Line with a digit: 99999
```

b)

sed-n'1,3p;6,7p;10p' fl.txt

Task b. Print line number 1-3, 6-7, and 10 of f1.txt

```
19/09/2023  05:59.09  /home/mobaxterm  sed -n '1,3p;6,7p;10p' f1.txt
This is the first line.
SDJIC Line 1
Line with | symbol.
Line with the word "fail."
This is a blank line.
SDJIC Line 2
```

c)

sed-n'/ASDIIC/p' fl.txt

Task c. Print lines beginning with "SDJIC" (case-sensitive) of f1.txt

Name:- PATEL NIYATI

Roll No:- 565

Division:- E

```
19/09/2023  05:59.09  /home/mobaxterm  sed -n '/^SDJIC/p' f1.txt
SDJIC Line 1
SDJIC Line 2
SDJIC Line 3
SDJIC Line 4
```

```
19/09/2023  05:59.10  /home/mobaxterm  sed -n '4,6p' f1.txt
Another line with sdjyc.
Line with a digit: 12345
Line with the word "fail."
```

d) sed-n'/AS/p' flixt

Task d. Print three lines starting from the fourth line of f1.txt

e) sed -n'/sdjic\|sdjyc/p' fl.txt

Task e. Print all blank lines of file f1.txt

f) sed-n'/A[A-Za-z0-9]/p' f1.txt

Task f. Print lines having either "sdjic" or "sdjyc" (case-sensitive) in f1.txt

```
19/09/2023  05:59.11  /home/mobaxterm  sed -n '/^[A-Za-z0-9]/p' f1.txt
This is the first line.
SDJIC Line 1
Line with | symbol.
Another line with sdjyc.
Line with a digit: 12345
Line with the word "fail."
This is a blank line.
This is the second line.
SDJIC Line 2
Line with | symbol.
Another line with sdjic.
Line with a digit: 67890
This is also a blank line.
This is the third line.
SDJIC Line 3
Line with | symbol.
Another line with sdjyc.
This is a third blank line.
This is the fourth line.
This line does not start with SDJIC.
This line does not contain | symbol.
Yet another line with sdjic.
Line with a digit: 55555
Another line with | symbol.
This is the fifth line.
SDJIC Line 4
Another line with sdjyc.
A line with | symbol.
Yet another line with sdjic.
Line with a digit: 99999
```

g) sed's/A/additional line\n/' f1.txt

Task g. Lines beginning with either an alphabet or a digit

h) sed'1,3s/|/:/g' fl.txt

Task h. Insert the line "add ional line" before every line

Name:- PATEL NIYATI

Roll No:- 565

Division:- E

i) sed's/|/:/g' fl.txt

Task j. Replace every occurrence of "|" with ":" in every line j)

sed'/fail/d' fl.txt

Task k. Remove all the lines containing the word "fail" from file f1.txt

```
19/09/2023  05:59.12  /home/mobaxterm  sed '/fail/d' f1.txt
This is the first line.
SDJIC Line 1
Line with | symbol.
Another line with sdjyc.
Line with a digit: 12345
This is a blank line.

This is the second line.
SDJIC Line 2
Line with | symbol.
Another line with sdjic.
Line with a digit: 67890
This is also a blank line.

This is the third line.
SDJIC Line 3
Line with | symbol.
Another line with sdjyc.
This is a third blank line.

This is the fourth line.
This line does not start with SDJIC.
This line does not contain | symbol.
Yet another line with sdjic.
Line with a digit: 55555
Another line with | symbol.

This is the fifth line.
SDJIC Line 4
Another line with sdjyc.
A line with | symbol.
Yet another line with sdjic.
Line with a digit: 99999
```