**Name : Tanishq Thuse**

**Branch : CS(AI)**

**Year : SY**

**Div : B**

**Roll no. : 60**

**Title : OS Assignment-2 Shell Scripting**

**I have written and executed these programs on my Fedora 40 Linux Distro**

**Q1)Menu Driven Programming in bash (shell scripting)**

Code :

***#!/bin/bash***

*# Function for arithmetic operations*

arithmetic\_operations() {

echo "Enter first number:"

read num1

echo "Enter second number:"

read num2

echo "Choose an operation (+, -, \*, /):"

read op

case $op in

+) result=$((num1 + num2)) ;;

-) result=$((num1 - num2)) ;;

\\*) result=$((num1 \* num2)) ;;

/) result=$((num1 / num2)) ;;

\*) echo "Invalid operation" ;;

esac

echo "Result: $result"

}

*# Function for file operations*

file\_operations() {

echo "Choose a file operation:"

echo "1. Create a file"

echo "2. Delete a file"

echo "3. List files in directory"

read file\_op

case $file\_op in

1) echo "Enter filename to create:"

read filename

touch "$filename"

echo "File '$filename' created." ;;

2) echo "Enter filename to delete:"

read filename

rm "$filename"

echo "File '$filename' deleted." ;;

3) echo "Enter directory to list files:"

read dir

ls "$dir" ;;

\*) echo "Invalid file operation" ;;

esac

}

*# Function for text processing*

text\_processing() {

echo "Choose a text processing operation:"

echo "1. Count lines in a file"

echo "2. Count words in a file"

echo "3. Search for a word in a file"

read text\_op

case $text\_op in

1) echo "Enter filename to count lines:"

read filename

line\_count=$(wc -l < "$filename")

echo "Number of lines in $filename: $line\_count" ;;

2) echo "Enter filename to count words:"

read filename

word\_count=$(wc -w < "$filename")

echo "Number of words in $filename: $word\_count" ;;

3) echo "Enter filename to search in:"

read filename

echo "Enter word to search for:"

read word

grep -o -i "$word" "$filename" | wc -l

echo "Number of occurrences of '$word' in $filename: $(grep -o -i "$word" "$filename" | wc -l)" ;;

\*) echo "Invalid text processing operation" ;;

esac

}

*# Main menu*

while true; do

echo "Main Menu"

echo "1. Arithmetic Operations"

echo "2. File Operations"

echo "3. Text Processing"

echo "4. Exit"

echo "Enter your choice:"

read choice

case $choice in

1) arithmetic\_operations ;;

2) file\_operations ;;

3) text\_processing ;;

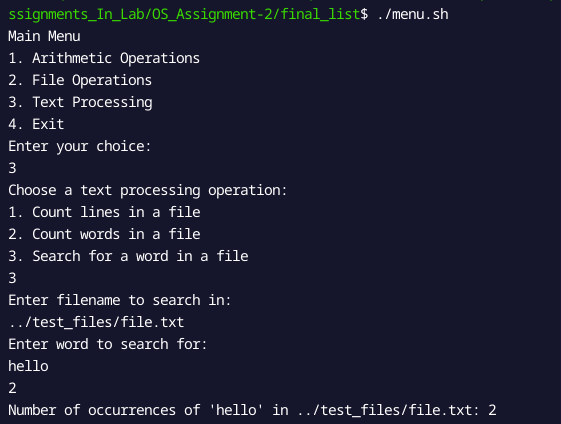
4) echo "Exiting..."; exit 0 ;;

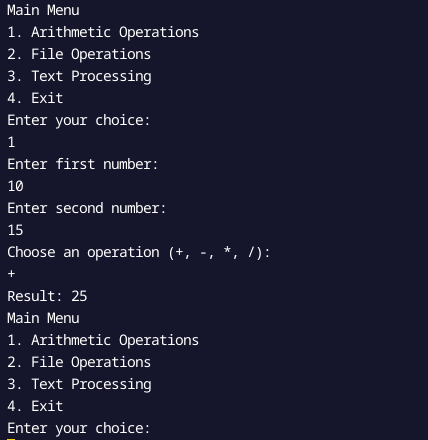
\*) echo "Invalid choice. Please enter a number between 1 and 4." ;;

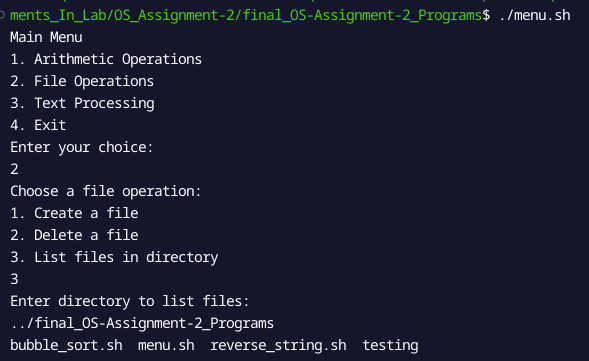
esac

done

Output :







**Q2)Bubble Sort :**

Code :

*#!/bin/bash*

*#bubble sorting function in shell scripting*

bubble\_sort() {

*local* arr=("$@")

*local* n=${#arr[@]}

for ((i = 0; i < n-1; i++)); do

for ((j = 0; j < n-i-1; j++)); do

if [ ${arr[j]} -gt ${arr[j+1]} ]; then

temp=${arr[j]}

arr[j]=${arr[j+1]}

arr[j+1]=$temp

fi

done

done

echo "Sorted Array: ${arr[\*]}"

}

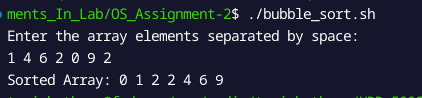
*#Main functions*

echo "Enter the array elements separated by space:"

read -a array

bubble\_sort "${array[@]}"

Output :



**Q3) Reverse String**

Code: ***#!/bin/bash***

reverse\_string() {

*local* str=$1

*local* len=${#str}

*local* rev=""

for ((i = len - 1; i >= 0; i--)); do

rev="$rev${str:i:1}"

done

echo "Reversed String: $rev"

}

echo "Enter a string:"

read str

reverse\_string "$str"

Output :

