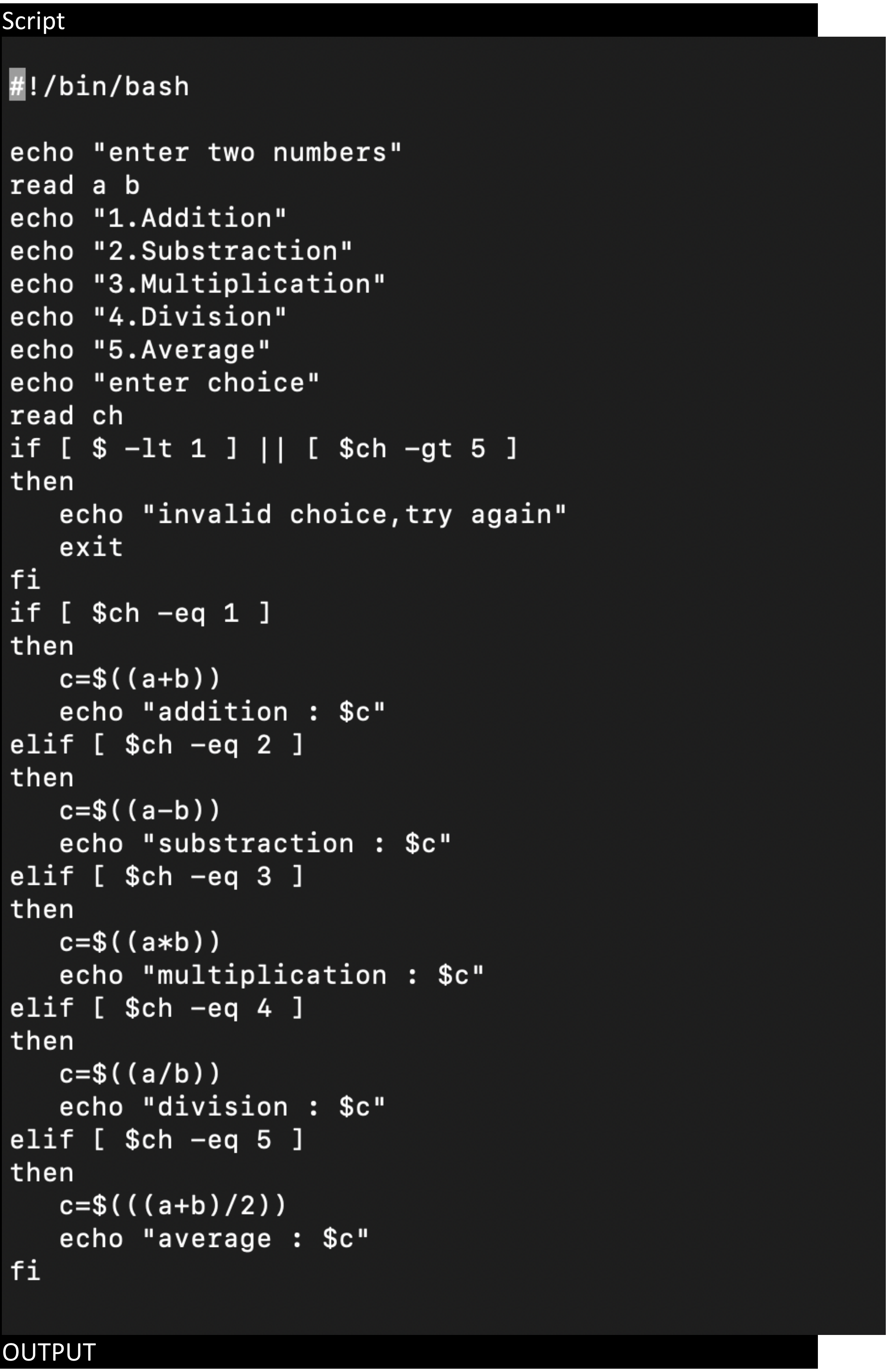
BASH SCRIPTING

1) Write a shell script to run the following operations by reading 2 numbers and one choice from the user:

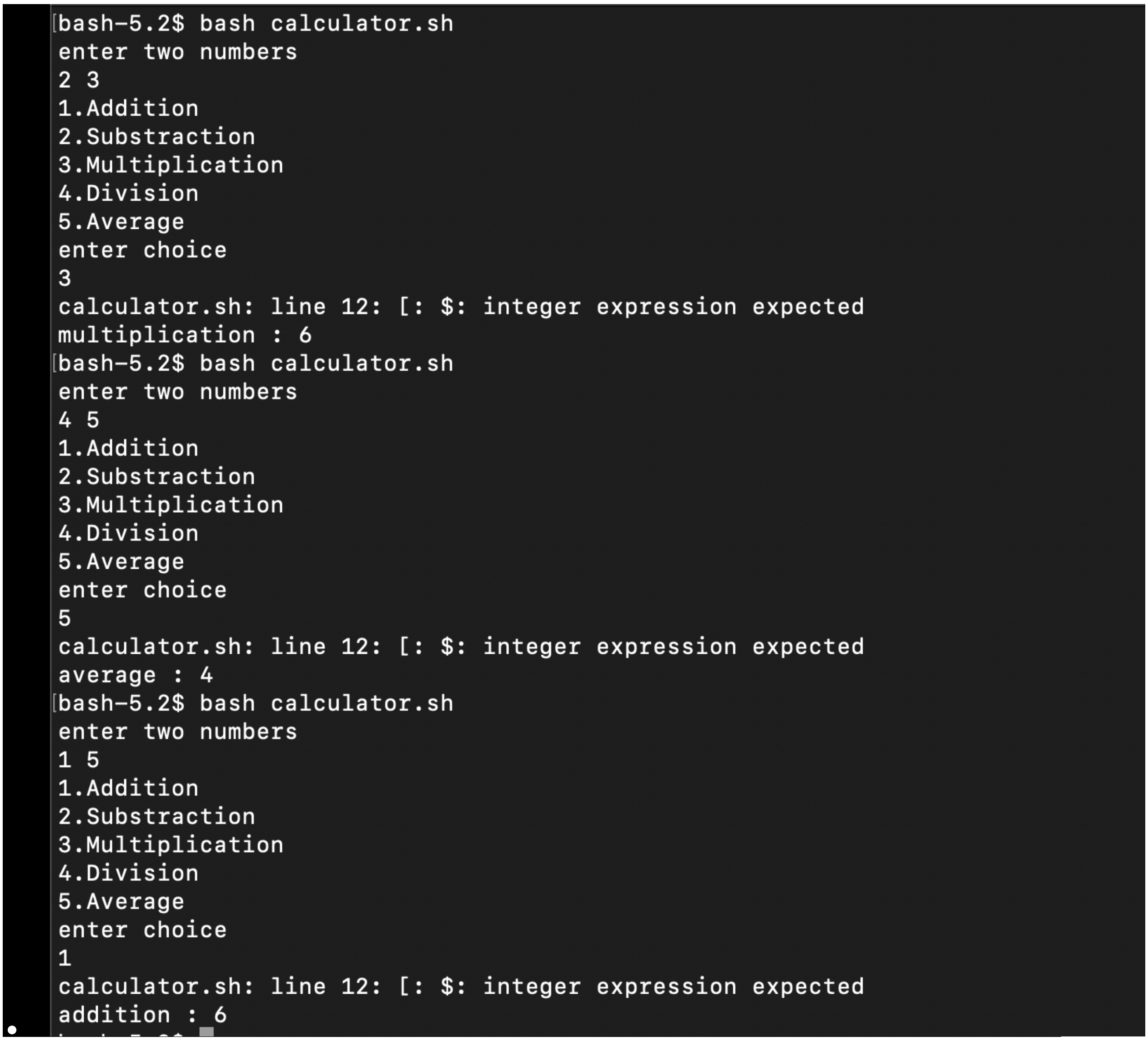
- Addition
- Subtraction
- Multiplication
- Division
- Average

- Script



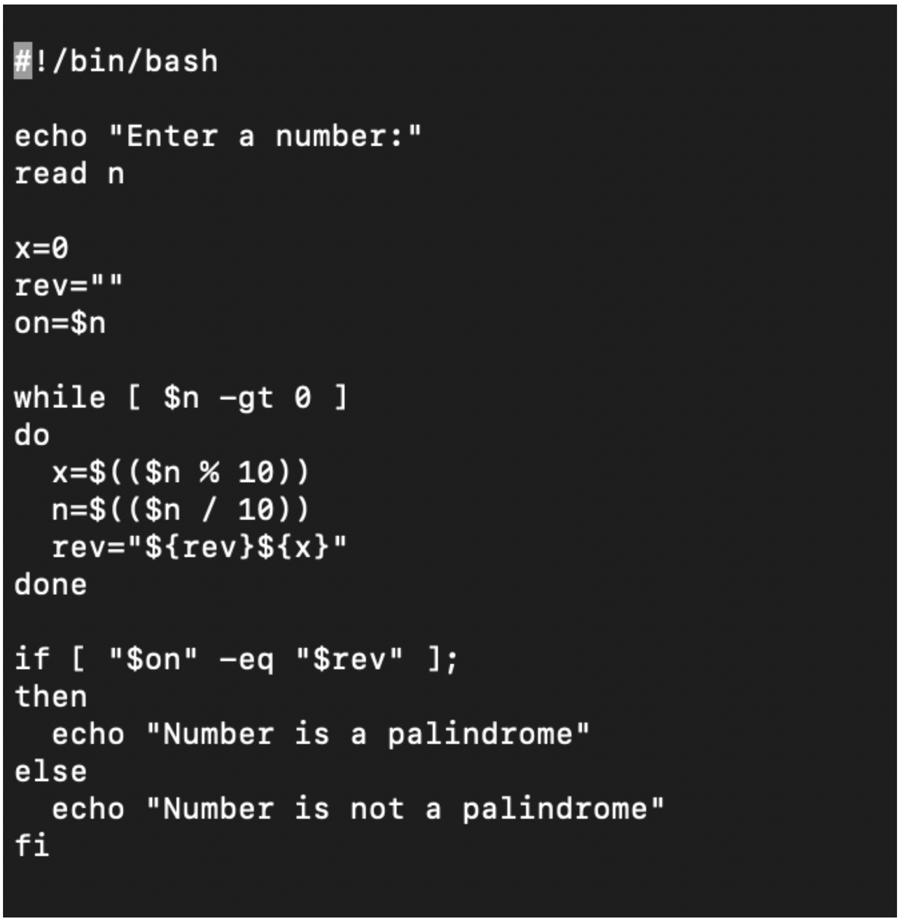
```
#!/bin/bash

echo "enter two numbers"
read a b
echo "1.Addition"
echo "2.Substraction"
echo "3.Multiplication"
echo "4.Division"
echo "5.Average"
echo "enter choice"
read ch
if [ $ -lt 1 ] || [ $ch -gt 5 ]
then
   echo "invalid choice,try again"
   exit
fi
if [ $ch -eq 1 ]
then
   c=$((a+b))
   echo "addition : $c"
elif [ $ch -eq 2 ]
then
   c=$((a-b))
   echo "substraction : $c"
elif [ $ch -eq 3 ]
then
   c=$((a*b))
   echo "multiplication : $c"
elif [ $ch -eq 4 ]
then
   c=$((a/b))
   echo "division : $c"
elif [ $ch -eq 5 ]
then
   c=$(((a+b)/2))
   echo "average : $c"
fi
```

OUTPUT



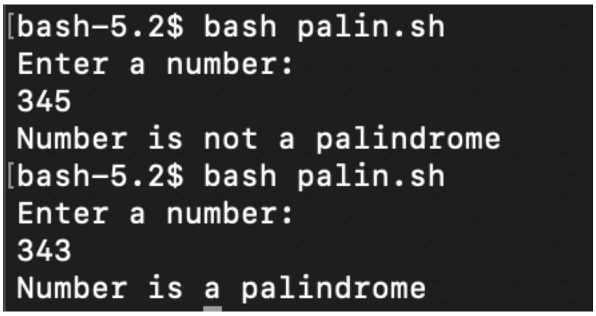
```
bash-5.2$ bash calculator.sh
enter two numbers
2 3
1.Addition
2.Substraction
3.Multiplication
4.Division
5.Average
enter choice
3
calculator.sh: line 12: [: $: integer expression expected
multiplication : 6
bash-5.2$ bash calculator.sh
enter two numbers
4 5
1.Addition
2.Substraction
3.Multiplication
4.Division
5.Average
enter choice
5
calculator.sh: line 12: [: $: integer expression expected
average : 4
bash-5.2$ bash calculator.sh
enter two numbers
1 5
1.Addition
2.Substraction
3.Multiplication
4.Division
5.Average
enter choice
1
calculator.sh: line 12: [: $: integer expression expected
addition : 6
```

- 2) Write a shell script to check whether a number is a palindrome or not



```
#!/bin/bash

echo "Enter a number:"
read n

x=0
rev=""
on=$n

while [ $n -gt 0 ]
do
  x=$(($n % 10))
  n=$(($n / 10))
  rev="${rev}${x}"
done

if [ "$on" -eq "$rev" ];
then
  echo "Number is a palindrome"
else
  echo "Number is not a palindrome"
fi
```

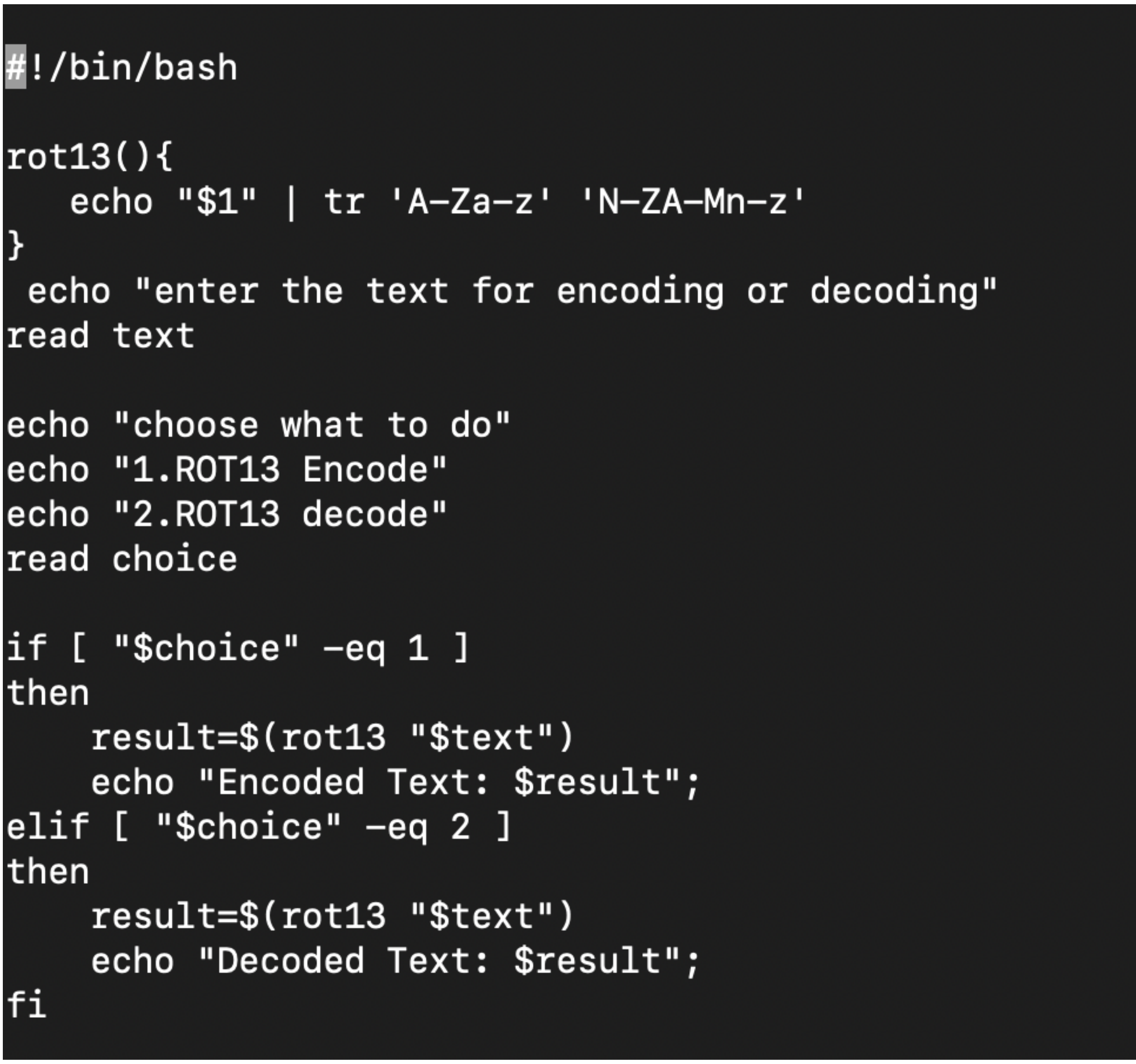
-
- OUTPUT



```
[bash-5.2$ bash palin.sh
Enter a number:
345
Number is not a palindrome
[bash-5.2$ bash palin.sh
Enter a number:
343
Number is a palindrome
```

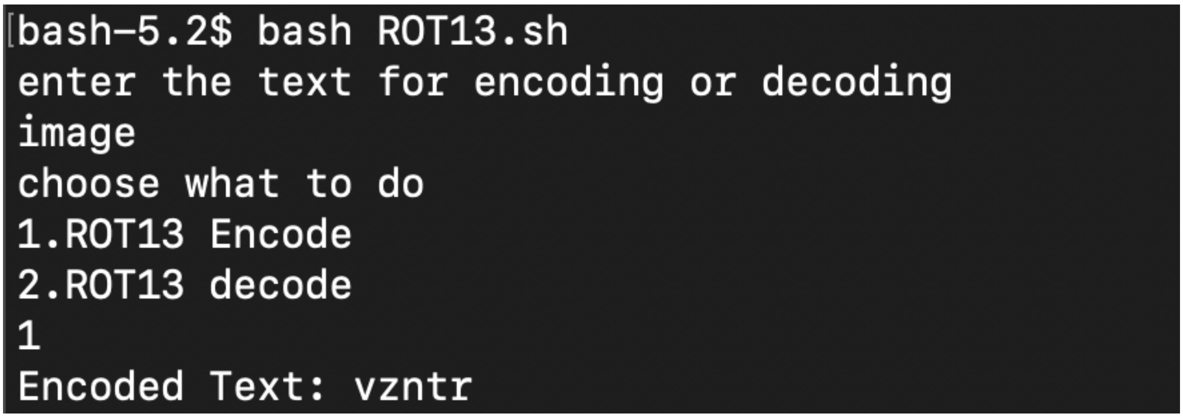
-

- 3) Write a script to run the following operations by reading an input and a choice from the user:
  - ROT13 Encode
  - ROT13 Decode



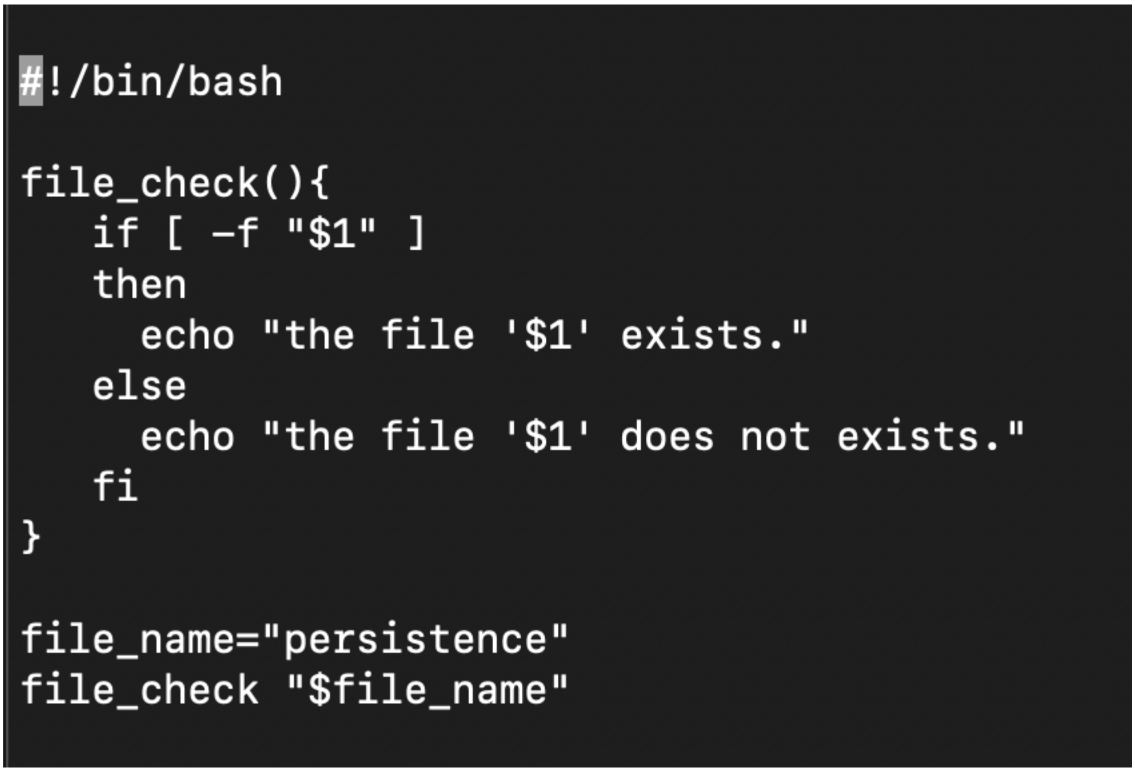
```
#!/bin/bash

rot13(){
   echo "$1" | tr 'A-Za-z' 'N-ZA-Mn-z'
}
 echo "enter the text for encoding or decoding"
read text

echo "choose what to do"
echo "1.ROT13 Encode"
echo "2.ROT13 decode"
read choice

if [ "$choice" -eq 1 ]
then
    result=$(rot13 "$text")
    echo "Encoded Text: $result";
elif [ "$choice" -eq 2 ]
then
    result=$(rot13 "$text")
    echo "Decoded Text: $result";
fi
```

- OUTPUT



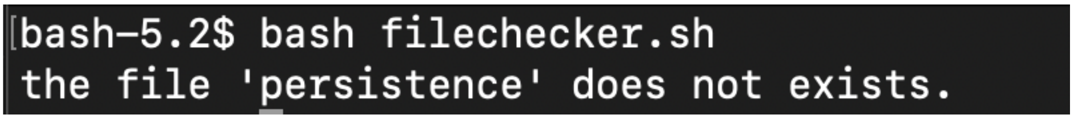
```
bash-5.2$ bash ROT13.sh
enter the text for encoding or decoding
image
choose what to do
1.ROT13 Encode
2.ROT13 decode
1
Encoded Text: vzntr
```

- 4) Using functions, write a shell script program to check whether a file named “persistence” is present in your system.
- CODE



```
#!/bin/bash

file_check(){
   if [ -f "$1" ]
   then
     echo "the file '$1' exists."
   else
     echo "the file '$1' does not exists."
   fi
}

file_name="persistence"
file_check "$file_name"
```

- OUTPUT



```
bash-5.2$ bash filechecker.sh
the file 'persistence' does not exists.
```