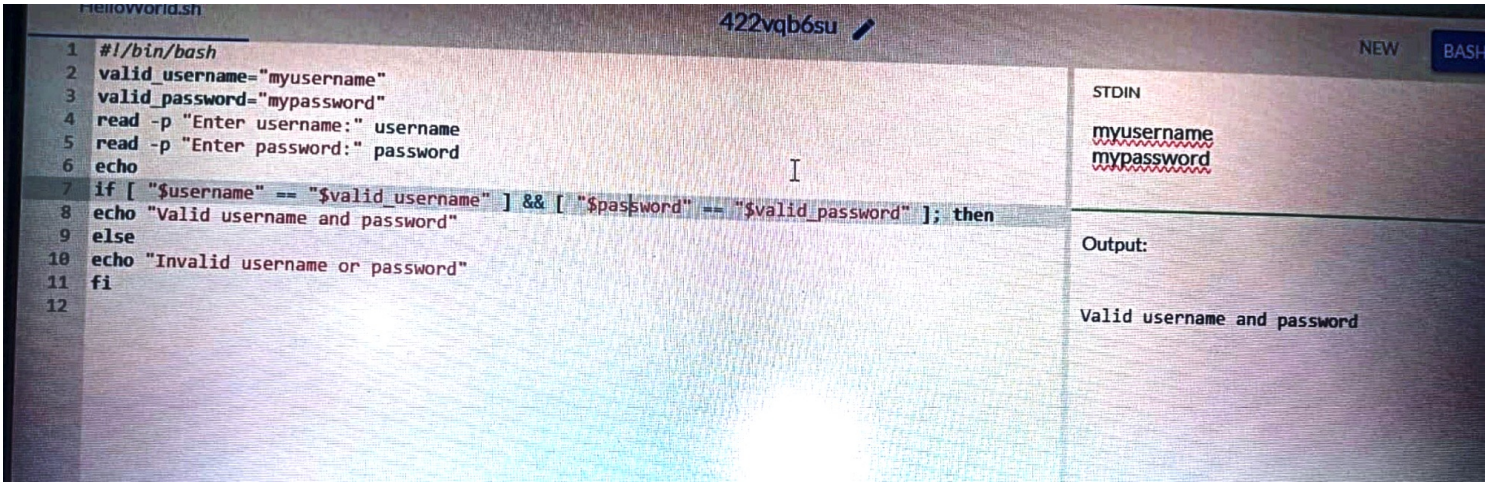
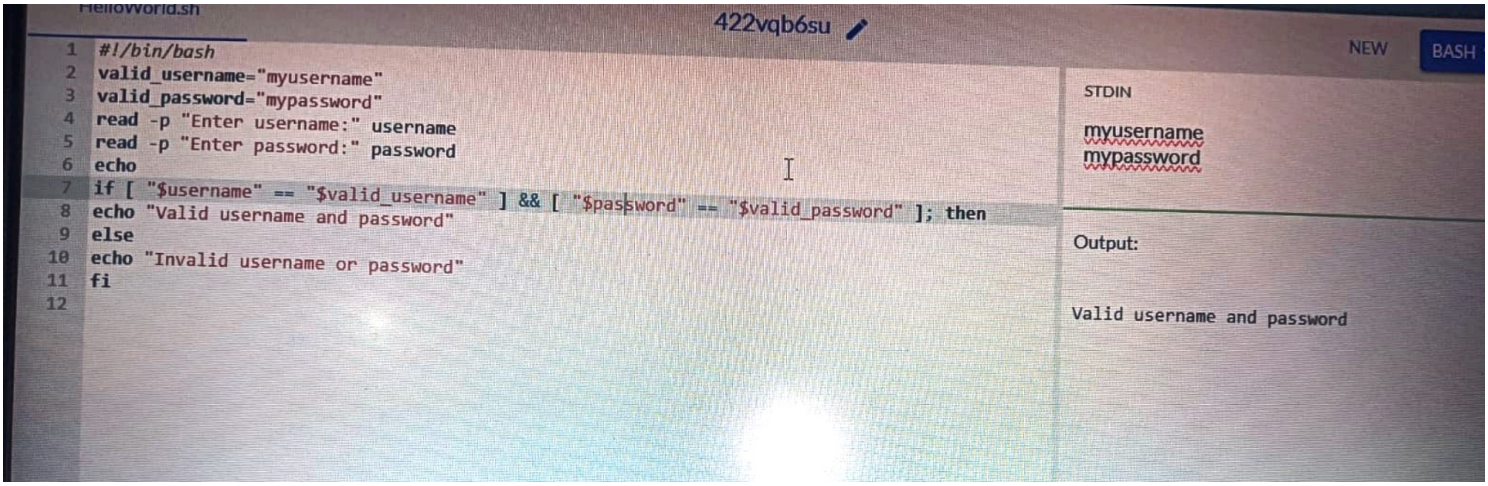

HelloWorld.sh
422vqb6su
NEW
BASH
#!/bin/bash
valid_username="myusername"
valid_password="mypassword"
read -p "Enter username:" username
read -p "Enter password:" password
echo
if ["$username" == "$valid_username"] && ["$password" == "$valid_password"]; then
echo "Valid username and password"
else
echo "Invalid username or password"
fi
STDIN
myusername
mypassword
Output:
Valid username and password


HelloWorld.sh
422vqb6su
NEW
BASH
#!/bin/bash
valid_username="myusername"
valid_password="mypassword"
read -p "Enter username:" username
read -p "Enter password:" password
echo
if ["$username" == "$valid_username"] && ["$password" == "$valid_password"]; then
echo "Valid username and password"
else
echo "Invalid username or password"
fi
STDIN
myusername
mypassword
Output:
Valid username and password

HelloWorld.sh

422vqzkjf

NEW

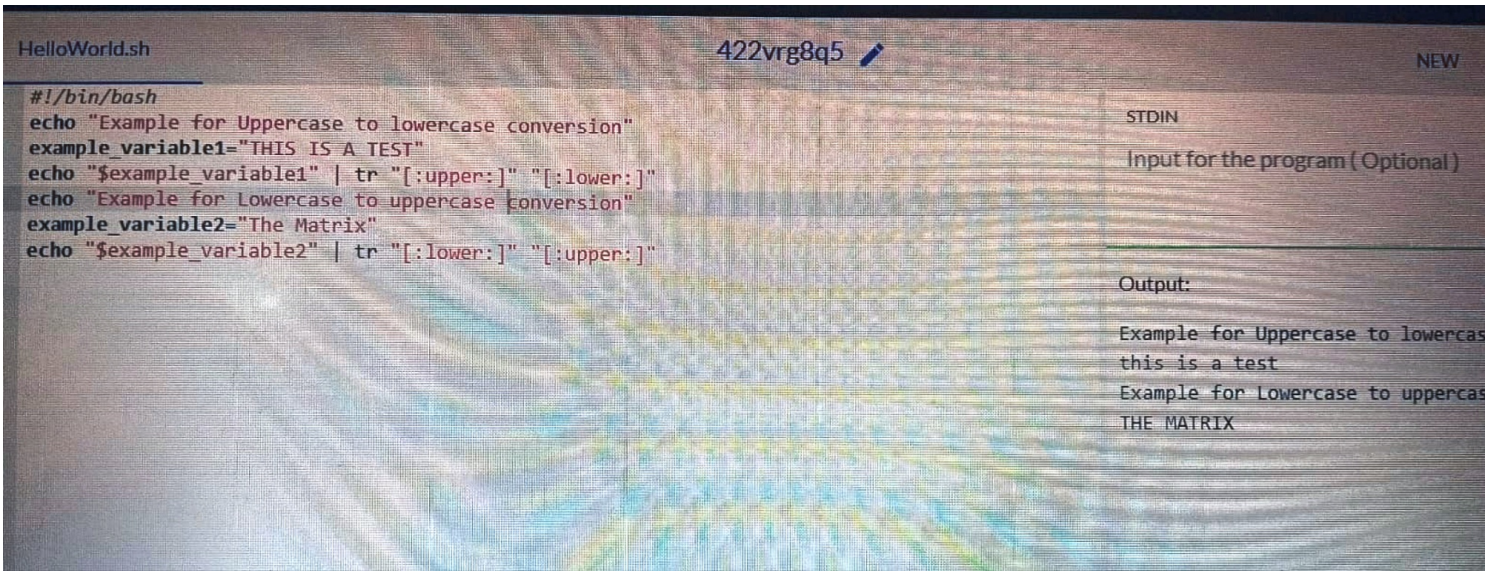
```
#!/bin/bash
a=10
b=20
x="hello"
y="world"
echo "a+b:$(($a+$b))"
echo "a-b:$(($a-b))"
echo "a%b:$(($a%$b))"
echo "a/b:$(($a/b))"
c=$x$y
echo $c
if [ $a == $b ]
then
  echo "a is equal to b"
else
  echo "a is not equal to b"
fi
```

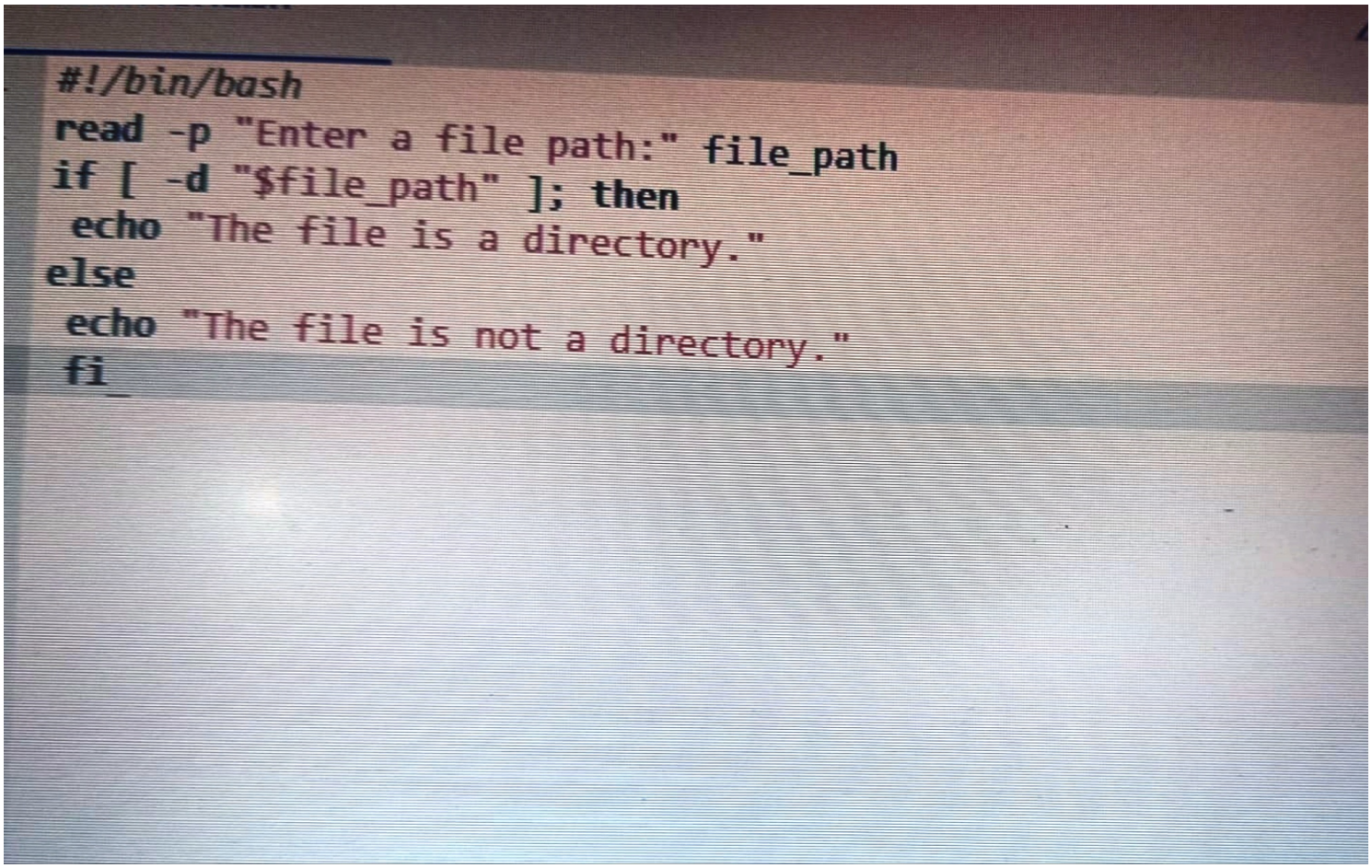
STDIN

Input for the program (Optional)

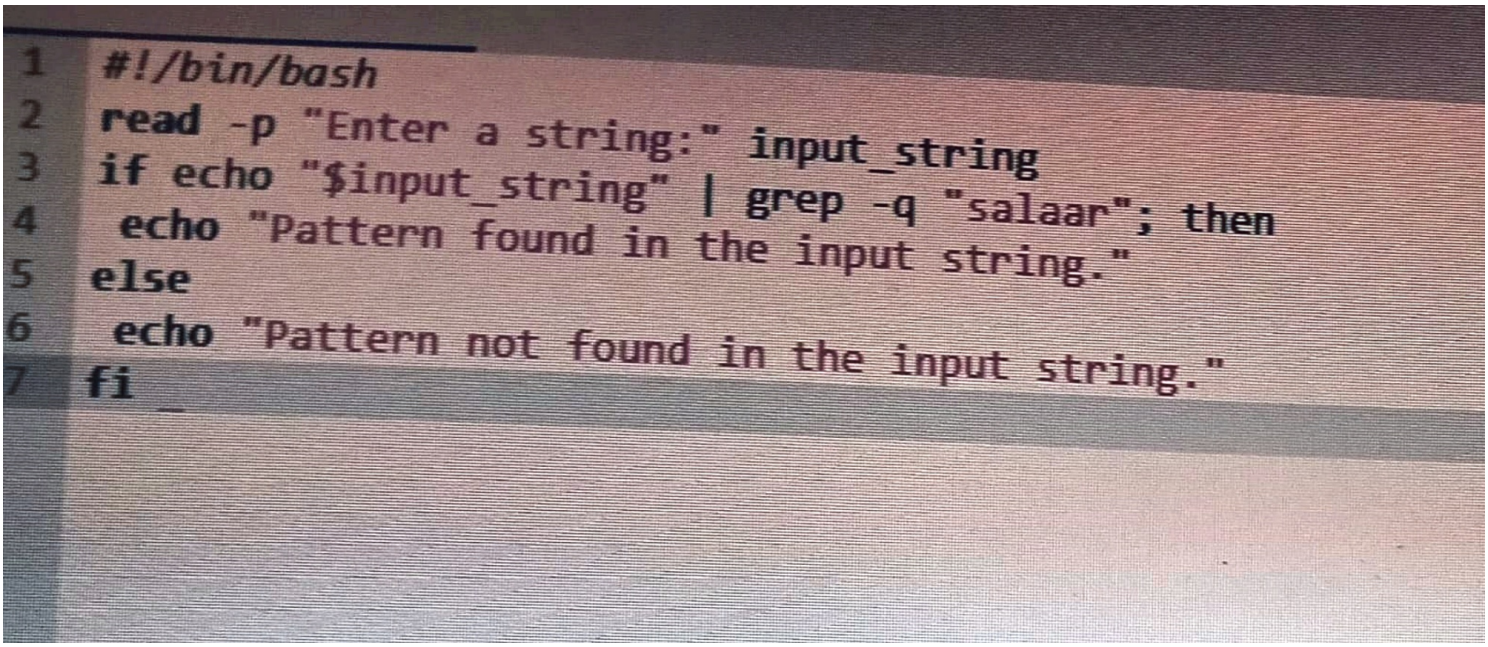
Output:

```
a+b:30
a-b: -10
a%b:10
a/b:0
helloworld
a is not equal to b
```

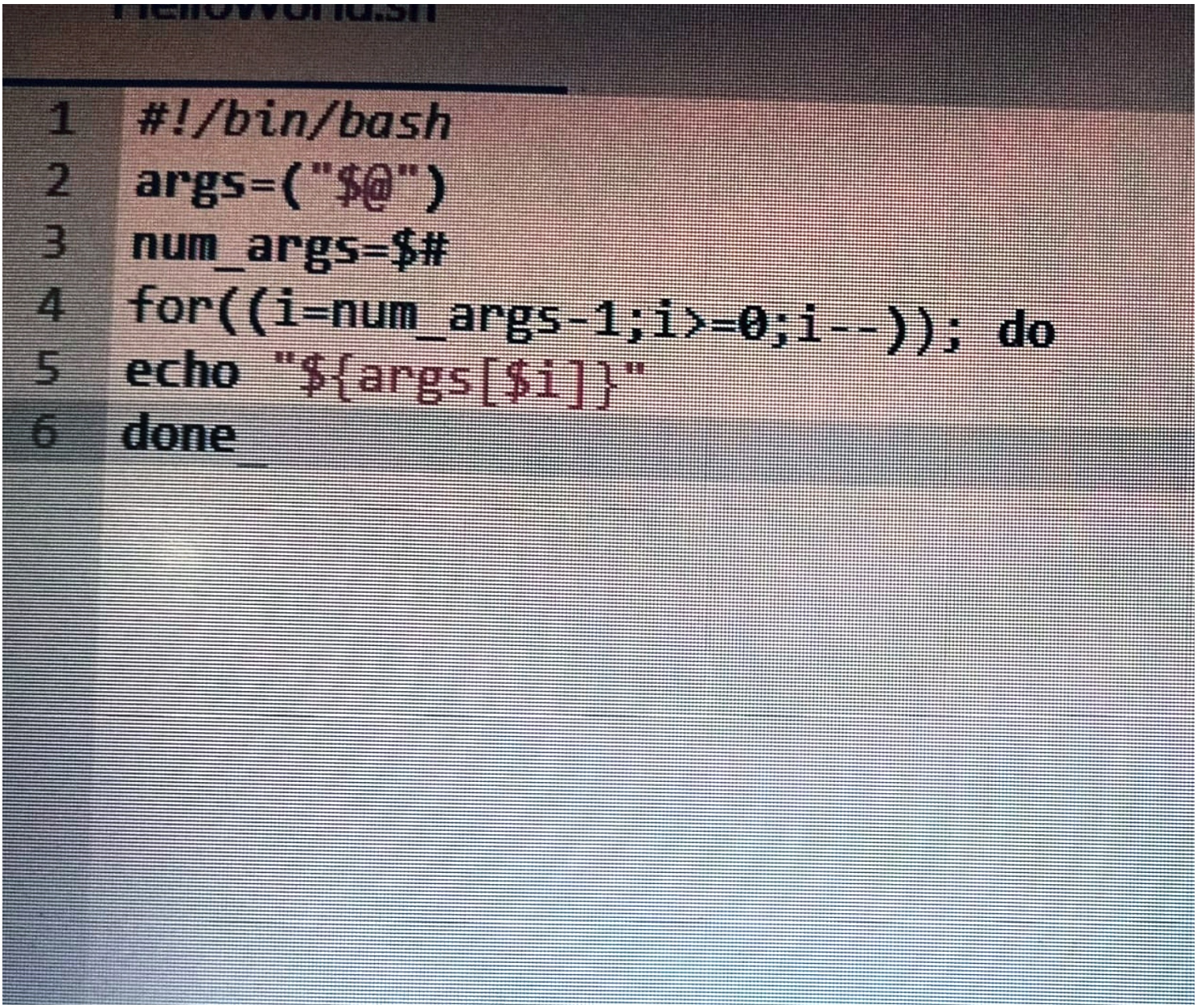

HelloWorld.sh
422vrg8q5
NEW
#!/bin/bash
echo "Example for Uppercase to lowercase conversion"
example_variable1="THIS IS A TEST"
echo "$example_variable1" | tr "[:upper:]" "[:lower:]"
echo "Example for Lowercase to uppercase conversion"
example_variable2="The Matrix"
echo "$example_variable2" | tr "[:lower:]" "[:upper:]"
STDIN
Input for the program (Optional)
Output:
Example for Uppercase to lowercas
this is a test
Example for Lowercase to uppercas
THE MATRIX



```
#!/bin/bash
read -p "Enter a file path:" file_path
if [ -d "$file_path" ]; then
 echo "The file is a directory."
else
 echo "The file is not a directory."
 fi
```

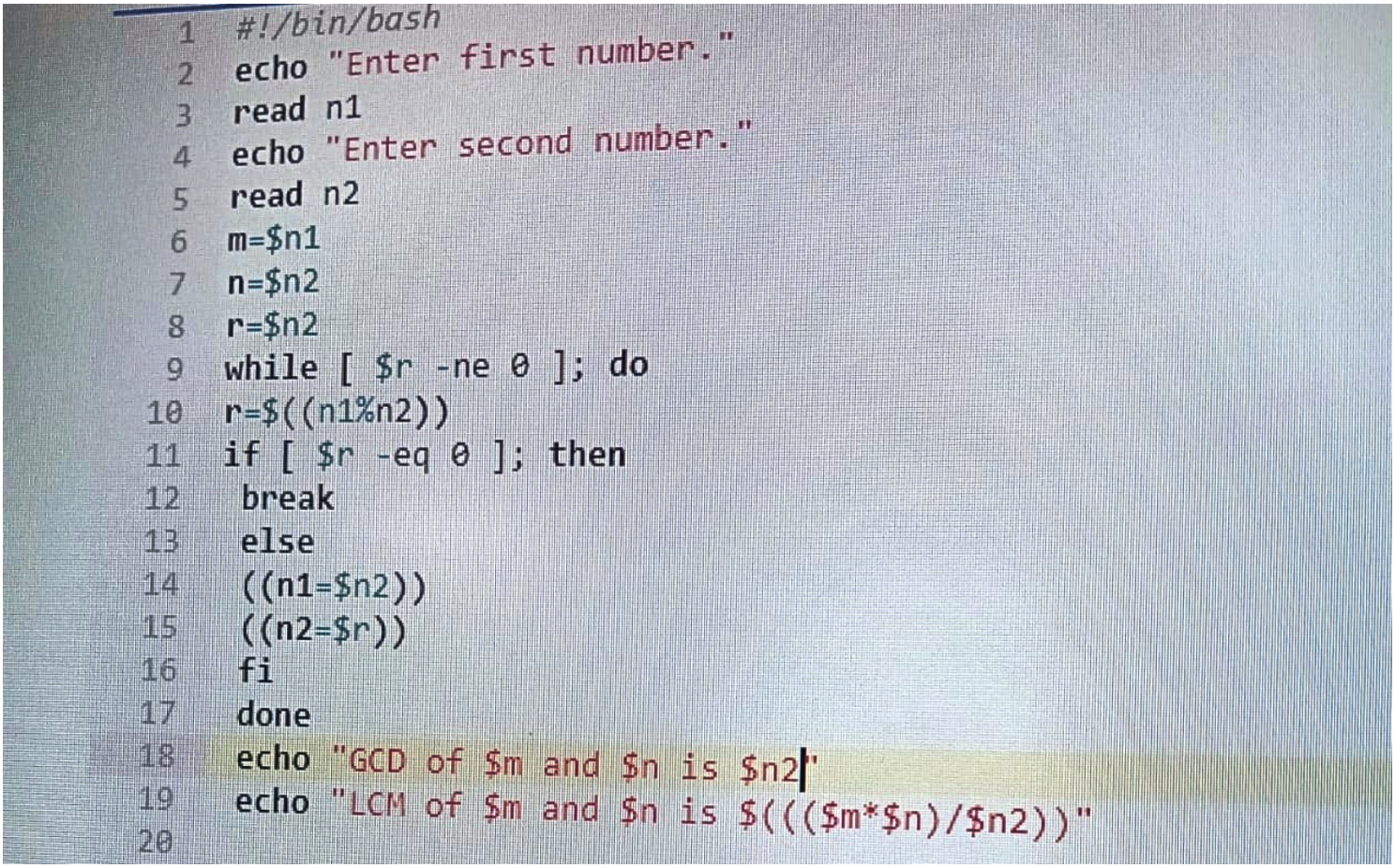


```
1  #!/bin/bash
2  read -p "Enter a string:" input_string
3  if echo "$input_string" | grep -q "salaar"; then
4   echo "Pattern found in the input string."
5  else
6   echo "Pattern not found in the input string."
7  fi
```

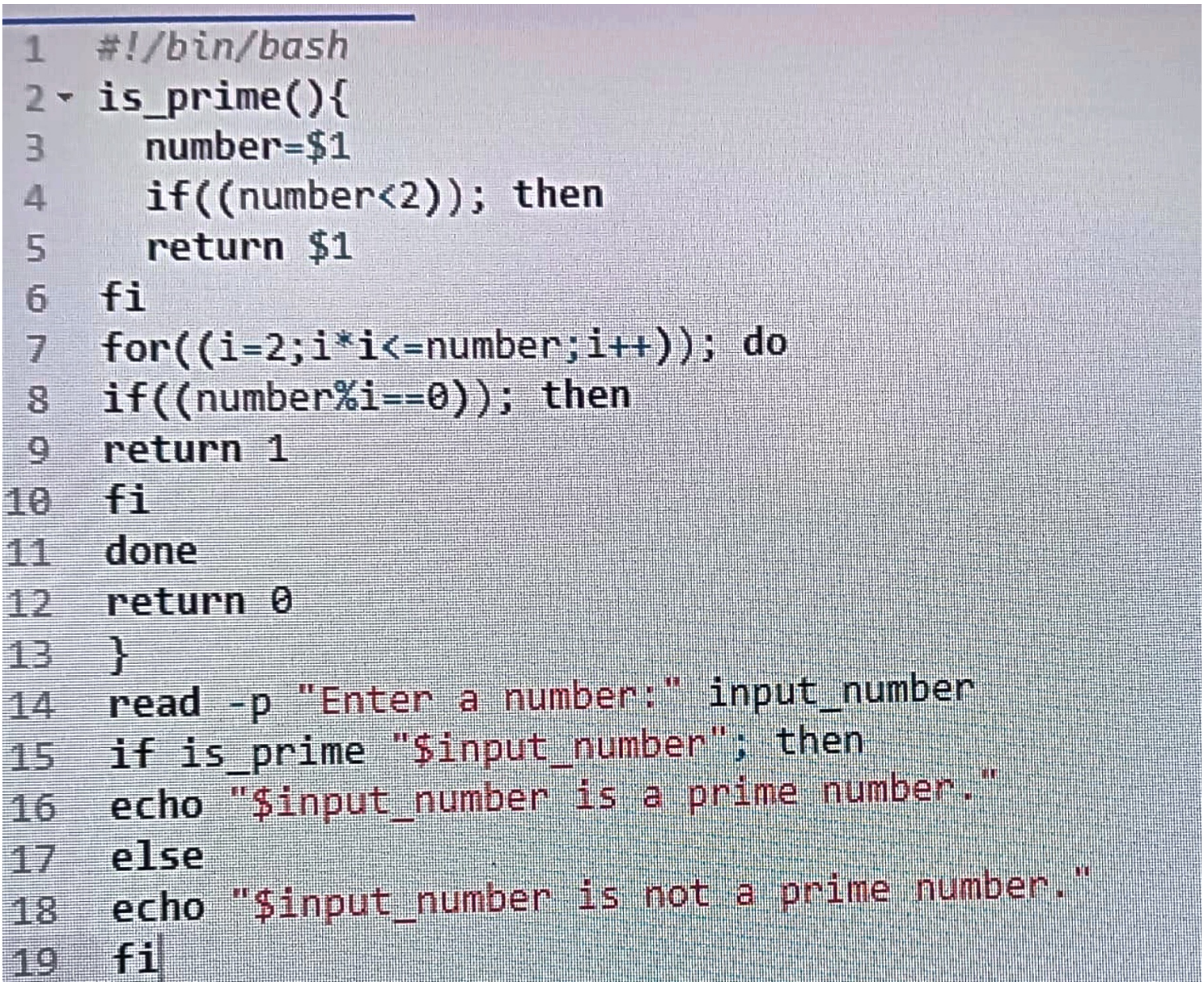

1 #!/bin/bash
2 args=("$@")
3 num_args=$#
4 for((i=num_args-1;i>=0;i--)); do
5 echo "${args[$i]}"
6 done

HelloWorld.sh

```
1  #!/bin/bash
2  read -p "Enter a year:" year
3  if((year%4==0 && year%100!=0))||((year%400==0)); then
4  echo "The year $year is a leap year."
5  else
6  echo "The year $year is not a leap year."
7  fi
```



```
#!/bin/bash
echo "Enter first number."
read n1
echo "Enter second number."
read n2
m=$n1
n=$n2
r=$n2
while [ $r -ne 0 ]; do
r=$((n1%n2))
if [ $r -eq 0 ]; then
 break
 else
 ((n1=$n2))
 ((n2=$r))
 fi
 done
 echo "GCD of $m and $n is $n2"
 echo "LCM of $m and $n is $(((($m*$n)/$n2))"
```



```
#!/bin/bash
is_prime(){
    number=$1
    if((number<2)); then
    return $1
fi
for((i=2;i*i<=number;i++)); do
if((number%i==0)); then
return 1
fi
done
return 0
}
read -p "Enter a number:" input_number
if is_prime "$input_number"; then
echo "$input_number is a prime number."
else
echo "$input_number is not a prime number."
fi
```

```
#!/bin/bash
read -p "Enter a string =": input_string
clean_string=$(echo "$input_string" | tr -dc "[:alnum:]" | tr "[:upper:]" "[:lower:]
reverse_string=$(echo "$clean_string" | rev)
if [ "$clean_string" == "$reverse_string" ]; then
echo "The string $input_string is a palindrome."
else
echo "The string $input_string is not a palindrome."
fi
```