

به نام خدای رنگین کمان

گزارش کار آزمایش ۳

تهیه شده توسط
امیر فاضل کوزه گر کالجی
علیرضا زارع زین آبادی

۹۹۳۱.۹۹-۹۹۳۱.۲۲

تمرین اول:

کد این تمرین به صورت زیر است:

```
1  #!/bin/bash
2  numReg='^[0-9]+$'
3
4  function check_is_num(){
5      if ! [[ $1 =~ $numReg ]] ; then
6          echo "error: $1 is Not a number"
7          return 0
8      fi
9      return 1
10 }
11
12 first=$1
13 second=$2
14 check_is_num $first
15 is_first_num=$?
16
17 check_is_num $second
18 is_second_num=$?
19
20 if [ $is_first_num -gt 0 ] && [ $is_second_num -gt 0 ]
21 then
22     sum=$(expr $first + $second)
23     echo "sum: $sum"
24
25     if [ $first -gt $second ]
26     then
27         echo "greater number is: $first"
28     elif [ $first -lt $second ]
29     then
30         echo "greater number is: $second"
31     else
32         echo "numbers are equal!"
33     fi
34 fi
35
```

با استفاده از تابع `check_is_num` چک میکنیم که ورودی ها طبق خواسته ما هستند یا نه (البته فقط اعداد صحیح میپذیریم) اگر ورودی ها صحیح بودند جمع آنها حساب میشود و در نهایت عدد بزرگتر مشخص میشود (اگر برابر بودند، برابری شان اعلام میشود :))

```
ab/lab-3$ bash Q1.sh 2 10
sum: 12
greater number is: 10
mr_amirfazel@Tahmoores:/medi
ab/lab-3$ bash Q1.sh 2 a
error: a is Not a number
mr_amirfazel@Tahmoores:/medi
ab/lab-3$ bash Q1.sh a 12
error: a is Not a number
mr_amirfazel@Tahmoores:/medi
ab/lab-3$ bash Q1.sh a b
error: a is Not a number
error: b is Not a number
```

شکل زیر برنامه را با ورودی های مختلف نشان میدهد:

تمرین دوم:
کد :

```
# !/bin/bash
echo "Enter Two numbers : "
read a
read b

echo "Enter operator :"
echo "1. Addition"
echo "2. Subtraction"
echo "3. Multiplication"
echo "4. Division"
read op

case $op in
1)res=`echo $a + $b | bc`
;;
2)res=`echo $a - $b | bc`
;;
3)res=`echo $a \* $b | bc`
;;
4)res=`echo "scale=2; $a / $b" | bc`
;;
esac
echo "Result : $res"
```

خروجی:

```
azare242@ahora-pc ~/Desktop
$ ./calc.sh
Enter Two numbers :
1
2
Enter Choice :
1. Addition
2. Subtraction
3. Multiplication
4. Division
2
Result : -1
```

تمرین
سوم:

```
#!/bin/bash
echo "Enter Number: "
read number
temp1=$number
while [ $temp1 -ne 0 ]
do
    rev=$rev$((temp1%10))
    temp1=$((temp1/10))
done

temp2=$number
sum=0
while [ $temp2 -gt 0 ]
do
    k=$((temp2%10))
    temp2=$((temp2/10))
    sum=$((sum+k))
done
echo "Reverse of $number = $rev, sum of digits of $number = $sum"
```

خروجی:

```
azare242@ahora-pc ~/Desktop
$ ./rev_sum.sh
Enter Number:
131884129
Reverse of 131884129 = 921488131, sum of digits of 131884129 = 37
```

```
#!/bin/bash

echo "enter x, y, filename"
read x
read y
read filename
lines=$(cat $filename)
i=0
for line in $lines
do
    if [ $i -eq $x ] || [ $i -eq $y ]; then
        echo "$line"
    elif [ $i -gt $x ] && [ $i -lt $y ]; then
        echo "$line"
    fi
    i=$((i+1))
done
```

```
A
BC
DEF
GHIJ
KLMNO
PQRSTU
VWXUZ
```

تمرین چهارم:

محتوای فایل تست:

خروجی:

```
azare242@ahora-pc ~/Desktop
$ ./read*.sh
enter x, y, filename
3
5
test.txt
GHIJ
KLMNO
PQRSTU
```

تمرین پنجم:
کد این قسمت به شکل زیر میباشد:

```

1 #!/bin/bash
2
3 function pattern1(){
4     for ((i=1;i<=5;i++))
5     do
6         for j in $(seq 1 $i)
7         do
8             echo -n "$i "
9         done
10        echo
11    done
12 }
13
14 function pattern2(){
15     for ((i=1;i<=6;i++))
16     do
17         for ((j=5;j>=$i;j--))
18         do
19             echo -n " "
20         done
21         for j in $(seq 1 $i)
22         do
23             echo -n "*"
24         done
25         echo
26     done
27     for ((i=1;i<=6;i++))
28     do
29         for ((j=1;j<$i;j++))
30         do
31             echo -n " "
32         done
33     done
34
35     function pattern3(){
36         for ((i=1;i<=5;i++))
37         do
38             for ((j=1;j<$i;j++))
39             do
40                 echo -n "| "
41             done
42             echo "|_"
43         done
44     }
45
46     case $1 in
47         "1")
48             pattern1
49             ;;
50         "2")
51             pattern2
52             ;;
53         "3")
54             pattern3
55             ;;
56     esac

```

خروجی آن برای ورودی های
مختلف به شکل زیر خواهد بود:

```
mr_amirfazel@Tahmoores:/media/
1
2 2
3 3 3
4 4 4 4
5 5 5 5 5
mr_amirfazel@Tahmoores:/media/
      *
     * *
    * * *
   * * * *
  * * * * *
 * * * * * *
* * * * * *
* * * * * *
 * * * * *
  * * * *
   * * *
    * *
     *
mr_amirfazel@Tahmoores:/media/
3: command not found
mr_amirfazel@Tahmoores:/media/
|_
| |
| | | |
| | | |
| | | |
mr_amirfazel@Tahmoores:/media/
```