PRACTICAL – 5

1. **Write a Shell script to accept a string as command line argument and reverse the same.**

```
if [ $# -ne 1 ]; then
echo "Usage: $0 <string>"
exit 1
fi
string=$1
reverse=""
for (( i=${#string}-1; i>=0; i-- )); do
reverse="$reverse${string:$i:1}"
done
echo $reverse
```

```
21012021003@telnetserver:~$ bash pr5.1.sh Amit
timA
```

2. **Write a shell script to calculate the loss percentage of an article, given the cost price and the selling price as command line arguments.**

```
if [ $# -ne 2 ]; then

  echo "Usage: $0 <cost price> <selling price>"

  exit 1

fi

cost_price=$1

selling_price=$2

if [ $selling_price -lt $cost_price ]; then

  loss=$(($cost_price - $selling_price))

  loss_percent=$(echo "scale=2; ($loss / $cost_price) * 100" | bc)

  echo "Loss Percentage: $loss_percent%"
```

Name: AMIT GOSWAMI
Enroll.No: 21012021003
Class : IT-B(AB5)

```
elif [ $selling_price -gt $cost_price ]; then
  profit=$(($selling_price - $cost_price))
  profit_percent=$(echo "scale=2; ($profit / $cost_price) * 100" | bc)
  echo "Profit Percentage: $profit_percent%"
else
  echo "No loss, no profit."
fi
```

```
21012021003@telnetserver:~$ bash pr5.2.sh 20 10
Loss Percentage: 50.00%
```

3. **Write a shell script to accept the name of the user and check out if the same has logged in or not.**

```
If          [ $# -ne 1 ]; then
  echo "Usage: $0 <username>"
  exit 1
fi
username=$1
last | grep $username > /dev/null
if [ $? -eq 0 ]; then
  echo "$username has logged in."
else
  echo "$username has not logged in."
fi
```

```
21012021003@telnetserver:~$ bash pr5.3.sh Amit
Amit has not logged in.
```

Name: AMIT GOSWAMI
Enroll.No: 21012021003
Class : IT-B(AB5)

4. **Write a shell script to check whether the file whose name is scanned exists and readable.**

```
if [ $# -ne 1 ]; then
  echo "Usage: $0 <filename>"
exit 1
fi
filename=$1
if [ -r $filename ]; then
  echo "$filename exists and is readable."
else
  echo "$filename does not exist or is not readable."
fi
```

```
21012021003@telnetserver:~$ bash pr5.4.sh abc.txt
abc.txt exists and is readable.
21012021003@telnetserver:~$ bash pr5.4.sh abcd.txt
abcd.txt does not exist or is not readable.
```

5. **Write a shell script to check if the input string is a palindrome.**

```
if [ $# -ne 1 ]; then
  echo "Usage: $0 <string>"
  exit 1
fi
string=$1
reverse=""
for (( i=${#string}-1; i>=0; i-- )); do
  reverse="$reverse${string:$i:1}"
done
```

Name: AMIT GOSWAMI
Enroll.No: 21012021003
Class : IT-B(AB5)

```
if [ "$string" = "$reverse" ]; then
  echo "$string is a palindrome."
else
  echo "$string is not a palindrome."
fi
```

```
21012021003@telnetserver:~$ bash pr5.5.sh malayalam
malayalam is a palindrome.
```

6. **Write a shell script to accept a number and a word as command line arguments and print the word the given number of times on each line.**

```
if [ $# -ne 2 ]; then
  echo "Usage: $0 <number> <word>"
  exit 1
fi

number=$1
word=$2

for (( i=1; i<=number; i++ )); do
  echo $word
done
```

```
21012021003@telnetserver:~$ bash pr5.6.sh 5 Amit
Amit
Amit
Amit
Amit
Amit
```

7. **Write a shell script to find the file or directory with the maximum size in the current directory.**

Name: AMIT GOSWAMI
Enroll.No: 21012021003
Class : IT-B(AB5)

```
max_size=0

for file in *; do

 if [ -d $file ]; then

   size=$(du -s $file | cut -f1)

 else

   size=$(wc -c < $file)

 fi

 if [ $size -gt $max_size ]; then

   max_size=$size

   max_file=$file

 fi

done



echo "The file or directory with the maximum size is $max_file"

echo "Size: $max_size bytes"
```

```
21012021003@telnetserver:~$ bash pr5.7.sh
The file or directory with the maximum size is pr5.2.sh
Size: 552 bytes
```

8. **Write a shell script to accept two filenames and check if both exist. If the second filename exists, then the contents of the first filename should be appended to it. If the second filename does not exist then create a newfile with the contents of the first file.**

```
if [ $# -ne 2 ]; then
   echo "Usage: $0 file1 file2"
   exit 1
fi
if [ ! -f $1 ]; then
   echo "File '$1' does not exist"
```

```
    exit 1
fi
if [ -f $2 ]; then
    cat $1 >> $2
    echo "Contents of '$1' appended to '$2'"
else
    cp $1 $2
    echo "File '$2' created with contents of '$1'"
fi
```

```
210120210003@telnetserver:~$ vi pr5.8.sh
210120210003@telnetserver:~$ chmod +x pr5.8.sh
210120210003@telnetserver:~$ bash pr5.8.sh abc.txt pr2
Contents of 'abc.txt' appended to 'pr2'
```

9. **Write a shell script to accept a number in the command line and displays the sum up to that number. By default, the sum up to 50 should be displayed.**

```
if [ $# -eq 0 ]

then

  n=50

else

  n=$1

fi

sum=0

for (( i=1; i<=$n; i++ ))

do

  sum=$((sum+i))

done



echo "The sum up to $n is $sum."
```

```
21012021003@telnetserver:~$ bash pr5.9.sh 50
The sum up to 50 is 1275.
```

10. **Write a shell script to find the number of ordinary files and directory files in the current directory.**

```
num_files=0

num_directories=0

for item in $(ls -1)

do

 if [ -f $item ]

 then

  num_files=$((num_files+1))

 elif [ -d $item ]

 then

  num_directories=$((num_directories+1))

 fi

done

echo "Number of files: $num_files"

echo "Number of directories: $num_directories"
```

```
21012021003@telnetserver:~$ vi pr5.10.sh
21012021003@telnetserver:~$ chmod +x pr5.10.sh
21012021003@telnetserver:~$ bash pr5.10.sh
Number of files: 23
Number of directories: 1
```

11. **Write a shell script to accept an alphabet from the user and list all the files/directory starting with that alphabet in the current directory.**

```
echo "Enter an alphabet:"
read alphabet
for item in $alphabet*
do
  echo $item
done
```

```
21012021003@telnetserver:~$ vi pr5.11.sh
21012021003@telnetserver:~$ chmod +x pr5.11.sh
21012021003@telnetserver:~$ bash pr5.11.sh
Enter an alphabet:
a
abc.txt
```

Name: AMIT GOSWAMI
Enroll.No: 21012021003
Class : IT-B(AB5)