Name:- A.F.M. Rafiul Hassan
ID:- 22-47048-1

Final Assignment

03

```
① #!/bin/bash
   read -a num
   sum=0
   for i in "${num[@]}"; do
      sum='expr $sum+$i'
   done
   echo $sum
```

```
② #!/bin/bash
   file ="notes.txt"
  if [-f "$file"]; then
  count ='wc -w < $file'
  echo $count
  else
   echo file doesn't exists.
  fi
```

```
③ #!/bin/bash
  file= "records.txt".
 If [-f "$file" ]; then
  count ='wc -l <$file'
  echo $count
 else
  echw file doesn't exists.

 fi
```

```
④ #!/bin/bash
  if [-f log.txt] ; then
    sed -i 's/error/issue/g' log.txt
    cat log.txt
   else
     echo " file Not found!"
  fi
```

```
⑤ #!/bin/bash
  wcount =0
  while read -r line; do
  wcount =$((wcount+$(echo $line | wc -w)))
   done < $file.txt
```

```
⑥ #!/bin/bash
  a=0
  b=1
  sum=0
  for((i=0; i<=10; i++)); do
       sum=$((sum + a))
       next=$((a+b))
    a=$b
    b=$next
   done
   echw    $ sum
```

P.T.O

7
```
#!/bin/bash
read num
factorial=1
if [ $num -lt 0 ]; then
echo "Invalid"
else
  for ((i=1; i<=num ; i++)); do
     factorial = $((factorial * i))
  done
  echo "The factorial of $num is:
         $ factorial"
fi;
```

8
```
#!/bin/bash
If [-f info.txt]; then
    head -n 10 info.txt
elif
  echo " file not found!"
fi
```

9
```
#!/bin/bash
array = (10 50 30, 40 100 60)

echo "${array[@]}" | tr ' ' '\n' |
                    sort | uniq -c "
```

10
```
#!/bin/bash
If [-f data.txt ]; then
        tail -n 5 data.txt
else echo " file not found!"
fi
```

11
```
#!/bin/bash
  if [-f output.txt]; then
     count = $(grep -o "Info" output.txt |
                               wc -l)
      echo $count
    else
       echo " file Not found"!"
   fi
```

12
```
#!/bin/bash
  if [-f log.txt ]; then
    tae log.txt
  else
   echo "file Not found!"

  fi
```

13
```
#!/bin/bash
 If [ -f words.txt ]; then
    grep -i "Priority" words.txt
  else
    echo" file not found! "
  fi
```

14
```
#!/bin/bash
read N
Product = 1
term =1
if [$N -lt 1]; then
  echo "Invalid"
else
  for ((i=1; i<=N ; i++)); do
   Product =$((Product * term))
   term =$((term * 10 +1))
done
fi echo $ Product
```

15

```
#!/bin/bash
  read N.
  Product =1
  term =10
  if [ $N -lt 1 ]; then
    echo "Invalid!"
  else
   for ((i=1; i<=n; i++)); do
     Product = $((Product * term))
     term =$((term * 10))
   done
  echo $Product
fi
```

16

```
#!/bin/bash
array=( 10 20 50 70 100 60 )
largest ="${array[@]}"; do
  if [$i -gt $largest ]; then
    largest =$i
  fi
done
echo $largest
```

17

```
#!/bin/bash
array= (1 2 3 4 5)
total=0
for i in "${array[@]}"; do
    total = $((total + i * i))
done
 echo $total
```

18

```
#!/bin/bash
if grep -q "Completed" task.txt; then
  echo "Exists"
 else
  echo "Not Exists"
 fi
```

19

```
#! /bin/bash
while IFS=',' read -r line
do
  name=($line)
  If [ ${#name[@]} -ge 3 ]; then
    echo "${name[2]}, ${name[1]}, ${name[0]}
   else
     echo "Error!"
   fi
 done < names.txt
```

20

```
#! /bin/bash
if [ -f notes.txt ]; then
  count= $(grep -o -w "$word" notes.txt |
                     wc -l)
   echo "$word appears $count times"
else
   echo file doesn't exists
fi
```

21)
```
#!/bin/bash

if [ -f log.txt ]; then
  sed -i '/^$/d' log.txt
  cat log.txt
else
    echo "file not found!"
fi
```

22)
```
#!/bin/bash
  read -a fname
for file in "${fname[@]}"; do
  sort $file
done
```

23)
```
#!/bin/bash

if [ -f data.txt ]; then
    stat --format=%s data.txt
else
    echo "file not found"
fi
```

24)
```
#!/bin/bash
read -a number
sum=0
for num in "${number[@]}"; do
if [ $num -gt 0 ]; then
  sum = $((sum + i))
fi
done
echo $sum
```

25)
```
#!/bin/bash
read name
fname=$(echo $name |
         cut -d ' ' -f 1)

lname=$(echo $name |
         cut -d ' ' -f 2)

echo $fname
echo $lname
```

26)
```
#!/bin/bash

read old_word
read new_word

if [ -f document.txt ]; then
   sed -i "s/$old_word/$new_word/g"
            document.txt
   cat document.txt
else
   echo "file not found!"
fi
```

[P.T.O]

27
```
#!/bin/bash
if [ -f logfile.txt ]; then
        date >> logfile.txt
else
      echo "File Not found!"
fi
```

28
```
#!/bin/bash
array=(10 30 50 70 100)
sum=0
arrayLen="${#array[@]}"
for i in "${array[@]}"; do
      sum=$((sum + i))
done
avg=$(echo "scale=2; $sum/$arrayLen" | bc)
echo "Avg of array elements: $avg"
```

29
```
#!/bin/bash
files=$(find . -type f | wc -l)
dirs=$(find . -type d | wc -l)

echo $files
echo $((dirs-1))
```

30
```
#!/bin/bash
if [ -f data.txt ]; then
   sort -n data.txt
else
   echo "file not found!"
fi
```