Ramakrishna Mission
Vivekananda Centenary College, Rahara

Department of Computer Science

Registration No.: A01-1112-117-014-2021 of 2021-2022
Roll No.: 715
Semester: 3rd Semester

Paper Code: CMSA CC – VI
Topic: Programming in Bash shell

Question 1. Write a shell program to print the sum of digits of a given number.

Program 1.

```
# program to add the digits of a number

echo "Enter a number"

read n
sum=0
while [ $n -gt 0 ]
do
      d=`expr $n % 10`
      sum=`expr $sum + $d`
      n=`expr $n / 10`
done
echo "Sum of digits" $sum
```

Output:

```
Enter a number
12352
Sum of digits 13
```

Question 2. Write a shell program to check whether a given number is a prime number or not.

Program 2.

```
# program to check whether a number is prime
echo "Enter a number"
read n
if [ $n -le 1 ]
then
      echo "Not prime"
      exit
fi
for (( i=2; i*i <= n; i++ ))
do
      if [ `expr $n % $i` -eq 0 ]
      then
            echo "Not a prime number"
            exit
      fi
done
echo "Prime number"
```

Output:

```
Enter a number
17
Prime number
```

Question 3. Write a shell program to check whether a given number is a perfect number or not.

Program 3.

```
echo "Enter a number:"
read a
i=1
fact=0
while [ $i -lt $a ]
do
        if [ `expr $a % $i` -eq 0 ]
        then
                fact=`expr $fact + $i`
        fi

        i=`expr $i + 1`
done
if [ $fact -eq $a ]
then
        echo "Perfect Number"
else
        echo "Not a perfect number"
fi
```

Output:

```
Enter a number:
6
Perfect Number
```

Question 4. Write a program to check whether a number is a palindrome or not.

Program 4.

```
# program to check whether a given number is palindrome
echo "Enter a number"
read n
m=$n
rev=0
while [ $n -gt 0 ]
do
      d=`expr $n % 10`
      rev=`expr $rev \* 10 + $d`
      n=`expr $n / 10`
done
if [ $rev -eq $m ]
then
      echo "Palindrome"
else
      echo "Not palindrome"
fi
```

Output:

```
Enter a number
12321
Palindrome
```

Question 5. Write a program to check whether a number is an automorphic number or not.

Program 5.

```
echo "Enter a number: "
read num
show=`expr $num`
flag=1
square=`expr $num \* $num`
while [ $num -gt 0 ]
do
      flag1=`expr $num % 10`
      flag2=`expr $square % 10`
      if [ $flag1 -ne $flag2 ]
      then
            flag=0
            break
      fi
      num=`expr $num / 10`
```

```
	square=`expr $square / 10`
done
if [ $flag -eq 0 ]
then
	echo $show "is not an automorphic number"
else
	echo $show "is an automorphic number"
fi
```

Output:

```
Enter a number:
76
76 is an automorphic number
```

Question 6. Write a shell program to perform binary search on an array.

Program 6.

```
echo "Enter the size of the array"
read n
echo "Enter" $n "numbers"
for ((i=0; i<n; i++))
do
	read a[$i]
done
for ((i=0; i<n; i++))
do
	for ((j=i+1; j<n; j++))
	do
		if [ ${a[$i]} -gt ${a[$j]} ]
		then
			temp=${a[$i]}
			a[$i]=${a[$j]}
			a[$j]=$temp
		fi
	done
done
echo "Enter the number to be searched in the array"
read key
lo=0
hi=`expr $n - 1`
while [ $lo -le $hi ]
do
	mid=`expr $lo + $hi`
```

```
	mid=`expr $mid / 2`

	if [ ${a[$mid]} -eq $key ]
	then
		echo $key "found at index" $mid
		exit
	elif [ ${a[$mid]} -gt $key ]
	then
		hi=`expr $mid - 1`
	else
		lo=`expr $mid + 1`
	fi
done
echo $key "is not present in the array"
```

Output:

```
Enter the size of the array
5
Enter 5 numbers
12
17
58
64
76
Enter the number to be searched in the array
64
64 found at index 3
```

Question 7. Write a shell program to sort a given array using bubble sort technique.

Program 7.

```
echo "Enter the size of the array"
read n
echo "Enter" n "numbers"
for ((i=0; i<n; i++))
do
	read a[$i]
done
for ((i=0; i<n; i++))
do
	for ((j=i+1; j<n; j++))
	do
		if [ ${a[$i]} -gt ${a[$j]} ]
		then
```

```
                temp=${a[$i]}
                a[$i]=${a[$j]}
                a[$j]=$temp
        fi
    done
done
echo "Sorted array is"
for ((i=0; i<n; i++))
do
    echo ${a[$i]}
done
```

Output:

```
Enter the size of the array
5
Enter 5 numbers
65
2
122
58
36
Sorted array is
2
36
58
65
122
```

Question 8. Write a program to sort an array using selection sort technique.

Program 8.

```
echo "Enter the size of the array"
read n
echo "Enter" $n "numbers"
for ((i=0; i<n; i++))
do
    read a[$i]
done
for ((i=0; i<n; i++))
do
    minidx=$i
    for ((j=i+1; j<n; j++))
    do
        if [ ${a[$minidx]} -gt ${a[$j]} ]
```

```
            then
                  minidx=$j
            fi
      done
      temp=${a[$minidx]}
      a[$minidx]=${a[$i]}
      a[$i]=$temp
done
echo "Sorted array = "
for ((i=0; i<n; i++))
do
      echo ${a[$i]}
done
```

Output:

```
Enter the size of the array
5
Enter 5 numbers
3
1
55
20
99
Sorted array =
1
3
20
55
99
```

Question 9. Write a program to check whether a given string is palindrome or not.

Program 9.

```
echo "Enter a string"
read str
len=`echo $str | wc -c`
for ((i=1; i<=len/2; i++))
do
      c1=`echo $str | cut -c $i`
      other_char=`expr $len - $i`
      c2=`echo $str | cut -c $other_char`
      if [ $c1 != $c2 ]
      then
            echo "Not palindrome"
            exit
      fi
done
echo "Palindrome"
```

Output:

```
Enter a string
madam
Palindrome
```

Question 10. Write a program to count the number of words, characters and lines in a file.

Program 10.

```
echo "Enter a filename"
read myfile
ch=`cat $myfile | wc -c`
wd=`cat $myfile | wc -w`
l=`grep -c "." $myfile`
echo $ch
echo $wd
echo $l
```

Output:

```
Enter a filename
test.txt
132
25
3
```

Question 11. Write a program to print the lines containing a particular word in a file.

Program 11.

```
echo "Enter a filename"
read filename
echo "Enter the word"
read wd
grep -i $wd $filename
```

Output:

```
Enter a filename
test.txt
Enter the word
is
This is a file containing some text.
```

Question 12. Write a program to print the lines not containing a particular word in a file.

Program 12.

```
echo "Enter the filename"
read filename
echo "Enter word"
read word
grep -v $word $filename
```

Output:

```
Enter a filename
test.txt
Enter word
is
Some text here.
```

Question 13. Write a program to reverse a string taken from command line argument.

Program 13.

```
str=$1
len=`echo $str | wc -c`
for ((i=len; i>0; i--))
do
      ch=`echo $str | cut -c $i`
      echo -n $ch
done
echo ""
```

Output:

```
./reverse_string.sh hello
olleh
```

Question 14. Write a program to count the number of vowels and consonants from a string taken as command line argument.

Program 14.

```
str=$1
len=`echo $str | wc -c`
v=0
c=0
for (( i=1; i<len; i++ ))
do
    ch=`echo $str | cut -c $i`
    case $ch in
        [aeiouAEIOU]) v=`expr $v + 1`
    esac
done
c=`expr $len - $v`
c=`expr $c - 1`
echo "The number of vowels is $v"
echo "The number of consonants is $c"
```

Output:

```
./vowel_consonant.sh hello
The number of vowels is 2
The number of consonants is 3
```