1. Write an algorithm, flow-chart and Program to input n numbers in an array and find the sum and average of elements in that array.

Algorithm:-

Step 1:- Start
Step 2:- Initialize the sum=0.
Step 3:- Enter how many numbers to be stored in an array(N).
Step 4:- Enter the elements (n) in an array
Step 5:- Calculate the sum of the elements (n) of an array along with their entries in array by
Sum = sum + a[i].
Step 6:- Calculate the average of the elements of an array as
Average = (float)sum/N.
Step 7:- Display the sum of elements in an array.
Step 8:- Display the average of elements in an array.
Step 9:- End

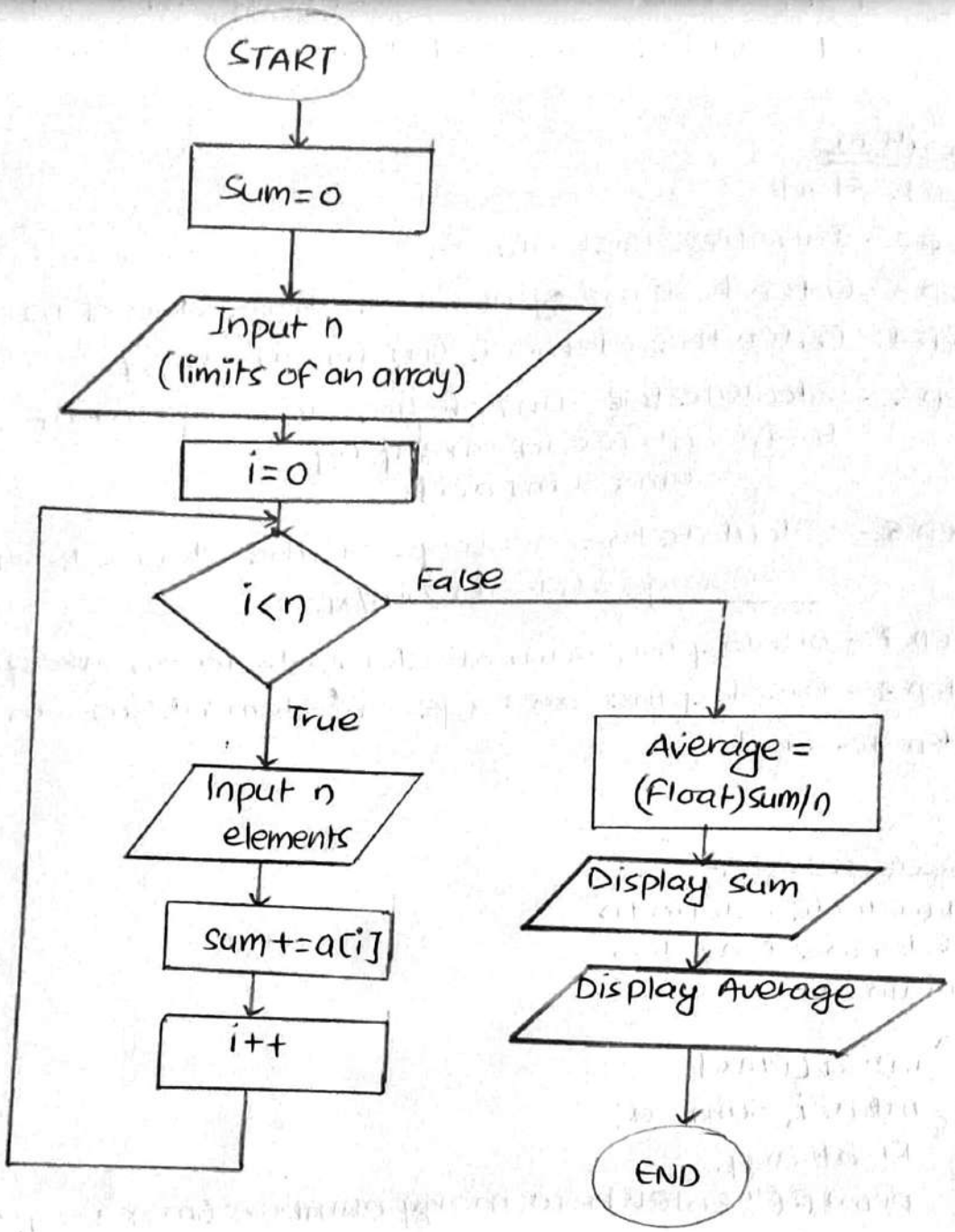
Source Code:-

```
#include <stdio.h>
#define MAX 100
main()
{
 int a[MAX];
 int n, i, sum=0;
 Float avg;
 printf("Enter how many numbers (max 100): ");
 scanf("%d", &n);
 for (i=0; i<n; i++)
  {
   printf("Enter numbers to an array:");
   scanf("%d", &a[i]);
   sum+=a[i];
  }
  avg = (float) sum/n;
  printf("The sum is %d", sum);
  printf("The average is %f", avg);
  getch();
  return 0;
}
```

Sample output:-

```
Enter how many numbers (max 100): 3
Enter numbers to an array: 10
Enter numbers to an array: 20
Enter numbers to an array: 30
The sum is 60
The average is 20.000000
```

Flowchart :-



2. WAP that takes input of two numbers and an operator in (+, -, *, /) as input and pass those numbers and an operator to the function. The function should calculate the result of two numbers as indicated by operator and return the result. Display the result of computation in your program.

Source code:-

```
#include <stdlib.h>
#include <stdio.h>
#include <conio.h>
Float compute (int, int, char);
main()
{
  int a, b;
  Float r;
  char op;
  printf("Input first number: ");
  scanf("%od", &a);
  printf("Input second number:");
  scanf("%od", &b);
  printf("Input operator: ");
  scanf("%c", &op);
   r= compute (a, b, op);
   printf("The result is %of", r);
   getch();
   return 0;
}
Float compute (int x, int y, char o)
 {
    Float result;
    if (o=='+')
    {
      result = x+y;
    }
    else if (o== '-')
      {
        result = x-y;
      }
    else if (o=='*')
      {
        result = x*y;
       }
    else if (o=='/')
      {
        result =(float)x/y;
      }
    else
       {
```

```
    printf("Invalid operator");
    getch();
    exit(0);
  }
  return result;
}
```

Sample output:-

```
Input first number: 19
Input second number: 11
Input operator: -
The result is 8
```

3. Write algorithm and program to compute the followings using recursion.

a. factorial of an integer N.

Algorithm:-

Step 1:- Start

Step 2:- Enter an integer N.

Step 3:- call the function by passing integer N.

Step 4:- In function definition

```
if (n==0)
 return 1;
else
 return(n* fact (n-1));
```

Step 5:- Display the factorial of integer 'N' returned by function

Step 6:- End

Source code:-

```
#include <stdio.h>
#include <conio.h>
long int fact (int);
main()
{
 int n;
 long int f;
 printf("Enter an integer:");
 scanf("%od", &n);
 f= fact(n);
 printf("The factorial of %od is %old", n,f);
 getch();
 return 0;
}
long int fact (int n)
{
 if (n==0)
   return 1;
  else
    return (n* fact (n-1));
}
```

Sample output:-

Enter an integer: 9

The factorial of 9 is 362880

b. multiplication of two integer a and b. i.e. a*b.

Alegorithm:-

Step 1:- Start

Step 2:- Enter two integers a & b.

Step 3:- Call the function by passing two integers a & b.

Step 4:- In function definition

```
if (b==0)
 return 0;
else
  return (a+a*(b-1));
```

Step 5:- Display the multiplication of two integers returned by the function.

Step 6:- End

Source Code:-

```
#include<stdio.h>
#include<conio.h>
long int multiple(int, int);
main()
{
  int a, b;
  long int d;
  printf("Enter the two numbers:");
  scanf("%d %d", &a, &b);
  d= multiple(a, b);
  printf("The multiplication of %d * %d = %ld", a, b, d);
  getch();
  return 0;
}
long int multiple(int a, int b)
  { if (b==0)
      return 0;
      else
       return (a+a*(b-1));
  }
```

Sample output:-

```
Enter the two numbers: 15 18
The multiplication of 15*18 = 270
```

c. computation of a^b (a raised to power b).

Algorithm:-

Step 1:- Start
Step 2:- Enter the base 'a'.
Step 3:- Enter the power 'b'.
Step 4:- Call the function by passing base and power.
Step 5:- In function definition
```
if (b==1)
 return a;
else
 return (a*power(a,b-1));
```
Step 6:- Display the value of a^b returned by the function
Step 7:- End

Source code:-

```
#include<stdio.h>
#include<conio.h>
long int power(int,int);
main()
{
  int a,b;
  long int d;
  printf("Enter the base:");
  scanf("%d",&a);
  printf("Enter the power:");
  scanf("%d",&b);
  d=power(a,b);
  printf("The value of %d^%d=%ld",a,b,d);
  getch();
  return 0;
}
long int power(int a, int b)
{ if (b==1)
    return a;
  else
    return (a* power(a,b-1));
}
```

Sample output:-

Enter the base: 2
Enter the power: 6
The value of 2^6 = 32.

4. Write an algorithm / program to print the prime numbers up to 100.

Algorithm :-

Step 1: Start
Step 2 : Intialize the num to 1
Step 3: Start the process from i=2
Step 4 : Check until i<=num if num%i==0, then break
Step 5 : Increase the value of i until i<=num.
Step 6: if (i==num). display the number and increase the value of number
Step 7: Repeat the process from 3 to 6, until num <=100.
Step 8: End

Source Code :-

```
#include <stdio.h>
#include <conio.h>
main()
{
  int i, num=1;
  do
   {
     i=2;
     while (i<=num)
     {
       if (num%i==0)
         {
           break;
         }
         i++;
     }
     if (i==num)
     printf("%d\t", num);
     num++;
    }while (num<=100);
   getch();
   return 0;
}
```

Sample output :-

```
2    3    5    7    11   13   17   19   23   29
31   37   41   43   47   53   59   61   67   71
73   79   83   89   97
```

5. Write a program to transpose mxn matrix and show the input matrix and it's transpose.

Source code:-

```
#include <stdio.h>
#include <conio.h>
main()
{
  int m[3][3], n[3][3], i, j;
  printf("Enter the elements of the matrix:");
  for (i=0; i<3; i++)
    {
      for (j=0; j<3; j++)
        {
          printf("m[%od][%od]", i, j);
          scanf("%od", &m[i][j]);
        }
    }
    for (i=0; i<3; i++)
      {
        for (j=0; j<3; j++)
          n[j][i] = m[i][j];
      }
  printf("The input matrix is \n");
  for (i=0; i<3; i++)
    {
      for (j=0; j<3; j++)
        {
          printf("%od\t", m[i][j]);
        }
        printf("\n");
    }
  printf("\n The transpose matrix is \n");
    for (i=0; i<3; i++)
      {
        for (j=0; j<3; j++)
          {
            printf("%od\t", n[i][j]);
          }
          printf("\n");
      }
    getch();
    return 0;

}
```

6. Write a program to perform the following regarding the string.
   a. input store a string and show the string.
   b. compute the length of string and display the length
   c. copy the string into another string and show both strings
   d. input another string into first string.
   e. concat the two string into third string and show all three strings.

Do all these operations in a single program without using the library functions for strings.

Source Code:

```
#include<stdio.h>
#include<conio.h>
#include<stdlib.h>
main()
{
   char *s1, *s2, *s3;
   int l=0, i, j;
   printf("Enter the first string:");
   s1= (char *) malloc (sizeof (char) * 100);
   scanf ("%os", s1);
   printf("Input string is: %os", s1);
   for (i= 0; s1[i]= '\0'; i++)      // finding length
     {
       i++;
     }
    printf ("Length of string is %od", l);
    s2 = (char *) malloc (sizeof (char)* 100);
    i=0;
    j=0;
    while (s2 [j] != '\0')
     {
        s2[j] = s1[i];
        i++;
        j++;
     }
    printf ("Original string is : %os", s1);
    printf ("Copied string is : %os", s2);
    printf ("Input another string in first string:");
    scanf ("%os", s1);
    printf ("New string is : %os", s1);
    s3 = (char *) malloc (sizeof(char) * 200);
    for (i=0; s1[i] = '\0'; i++)   //- concating
     {
        s3[i] = s1[i];
     }
```

```
for (j=0; s2[j] != '\0'; j++)
 {
    s3[i] = s2[j];
    i++;
 }
 s3[i] = '\0';
 printf ("First string is: %os", s1);
 printf ("Second string is: %os", s2);
 printf ("Concated string is: %os", s3);
 getch();
 return 0;
}
```

Sample output:-

```
Enter the first string: Dipendra
Input string is: Dipendra
Length of string is 8
ORiginal string is: Dipendra
copied string is: Dipendra
Input another string in the first string: Chand
New string is: chand
First string is: chand
Second string is: Dipendra
Concated string is: ChandDipendra
```

7. Write a program by using your own function strinput(), strshow(), strslength(), strcopy() and strconcat() to perform all operations as in Q. No. 6.

Source Code:-

```
#include <stdio.h>
#include <conio.h>
#include <stdlib.h>
void strinput (char *);
void strshow (char *);
int strslength (char *);
void strcopy (char *, char *);
void strconcat (char *, char *);
main()
{
  char *s1, *s2, *s3;
  int len;
  s1= (char *) malloc (sizeof (char) *100);
  strinput (s1);
  strshow (s1);
  len = strslength (s1);
  printf ("Length of string is %d", len);
  s2 = (char *) malloc (sizeof (char) *100);
  strcopy (s2, s1);
  printf ("Original string is : %s", s1);
  printf ("Copied string is : %s", s2);
  printf ("Input another string in first string: ");
  strinput (s1);
  strconcat (s1, s2);
  s3 = (char *) malloc (sizeof (char) * 200);
  printf ("First string is: %s", s1);
  printf ("Second string is : %s", s2);
  printf ("Concated string is : %s", s3);
  getch();
  return 0;
}
void strinput (char *s1)
{
  printf ("Enter the string:");
  scanf ("%s", s1);
}
void strshow (char *s)
{
  printf ("Input string is:%s", s);
}
```

```
int strslength (char *str)
{
  int l=0, i;
  for (i=0; str[i]!='\0'; i++)
   {
     l++;
   }
  return l++;
}
void strcopy (char *s2, char *s1)
{
  int i=0, j=0;
  while (s2[j]!='\0')
   {
     s2[j] = s1[i];
     i++;
     j++;
   }
}
void strconcat (char *s1, char *s2)
{
   char *s3;
   s3= (char*) malloc (sizeof(char)*200);
   int i, j;
   for (i=0; s1[i]!='\0'; i++)
   {
     s3[i] = s1[i];
   }
   for (j=0; s2[j]!='\0'; j++)
    {
      s3[i]= s2[j];
      i++;
    }
   s3[i]='\0';
}
```

Sample output:-

```
Enter the string: Dipen
Input string is : Dipen
Length of string is 5
original string is : Dipen
copied string is : Dipen
Input Another string in first string
Enter the string: Chand
First string is : Dipen
Second string is : Chand
concated string is : DipenChand
```

8. WAP defining a structure to store the record of a student that includes first Name, last Name, Address, Roll No Age etc. Input the records of N students and show all the records.

Source Code:-

```
#include <stdio.h>
#include <conio.h>
#include <stdlib.h>
#define N 36
struct name
  {
   char fname[10];
   char lname[10];
  }n;
main()
{
  struct student
    {
      struct name n;
      char address[40];
       int rollno;
       int age;
    };
  struct student s[36];
  int i;
  printf("Enter the records of the student: \n");
  for(i=0; i<N; i++)
  {
    printf("student %od", i+1)
    printf(" First Name:");
    scanf("%os", s[i].n.fname);
    printf("\n Last Name:");
    scanf("%os", s[i].n.lname);
    printf("\n Address:");
    scanf("%os", s[i].address);
    printf("\n Roll No:");
    scanf("%od", &s[i].rollno);
    printf("\n Age:");
    scanf("%od", &s[i].age);
  }
  printf("S.No\t Name\t\t Address \t\t Roll No\t Age\n");
  for(i=0; i<N; i++)
  {
   printf("%od\t %os %os\t\t %os \t\t %od\t %od\n", i+1, s[i].n.fname,
                    s[i].n.lname, s[i].address, s[i].rollno, s[i].age);
  }
```

```
getch();
return 0;

}
```

9. WAP to open a new file, read rollno, name, address and phone_no until the user says "no". After reading all the data, write it to the file. Display the records from file in alphabetical order of student name.

Source Code:-

```
#include<stdio.h>
#include<conio.h>
#include<stdlib.h>
#include<string.h>
#define MAX 100
typedef struct student
  {
    int rollno;
    char name[40];
    char address[50];
    char phoneno[20];
  }student;
main()
{
  int i=0, j, n;
  char choice[10];
  student stu[MAX], tmp, s[MAX];
  FILE *fp;
  fp= fopen("record.dat", "w+");
  if (fp==NULL)
  {
    printf("Error on creating file");
    getch();
    exit(0);
  }
do
  {
    printf("Enter the record of the student:%od:\n", i+1);
    printf("Roll No:");
    scanf("%od", &stu[i].rollno);
    printf("Name:");
    scanf("%os", stu[i].name);
    printf("Address:");
    scanf("%os", stu[i].address);
    printf("Phone No:");
    scanf("%os", stu[i].phoneno);
    i++;
    printf("Do you want to continue? Yes|No");
    scanf("%os", choice);
  }while(strcmp(choice, "no")!=0);
```

```
fwrite(&stu[0], sizeof(student), i, fp);
rewind(fp);
n=0;
while(1)
{
  if(fread(&s[n], sizeof(student), 1, fp)==0)
    break;
    n++;
}
for(i=0; i<n-1; i++)
 {
  for(j=i+1; j<n; j++)
  {
    if(strcmp(s[i].name, s[j].name)>0)
      {
        tmp=s[i];
        s[i]=s[j];
        s[j]=tmp;
      }
  }
 }
printf("\n Record of students after sorting alphabetically:");
printf("S.No\t Name\t Roll No\t Address\t Phone No:\n");
for(i=0; i<n; i++)
{
printf("%d\t %s\t %d\t %s\t %s\n", i+1, s[i].name, s[i].rollno,
                                    s[i].address, s[i].phoneno);

}
fclose(fp);
getch();

}
```