POP LAB Programs

1. Simulation of a Simple Calculator.

```
#include<stdio.h>
#include<conio.h>
void main()
{
        int op1,op2;
        char cop;
        clrscr();
        printf("Enter the valid arithmetic expression\n");
        scanf("%d%c%d",&op1,&cop,&op2);
        switch(cop)
        {
                case '+': printf("Result=%d",op1+op2);
                          break;
                case '-': printf("Result=%d",op1-op2);
                          break;
                case '*': printf("Result=%d",op1*op2);
                          break;
                case '/': printf("Result=%d",(op1/op2));
                  break;
                case '%': printf("Result=%d",(op1%op2));
                          break;
                default: printf("Invalid Expression\n");
        }
        getch();
}
```

Output:

Enter the valid arithmetic expression
2+3
Result=5
Enter the valid arithmetic expression
6-4
Result=2
Enter the valid arithmetic expression
*4*5*
Result=20
Enter the valid arithmetic expression
8/4
Result=2

2. **Compute the roots of a quadratic equation by accepting the coefficients. Print appropriate messages.**

```
#include<stdio.h>
#include<math.h>
#include<conio.h>
void main( )
{
        float a, b, c, d,r1,r2;
        clrscr();
        printf("Enter the 3 coefficients:\n");
        scanf("%f%f%f", &a, &b, &c);
        if((a*b*c) == 0)
        {
                printf("Roots cannot be Determined:\n");
                exit(0);
        }
        d = (b*b) - (4*a*c);
        if(d == 0)
        {
                printf("Roots are real and equal\n");
                r1=r2= -b / (2*a);
                printf("The roots are %f and %f",r1,r2);
        }
        else if(d>0)
        {
                printf("Roots are real and distinct\n");
                r1= (-b + sqrt(d)) / (2*a);
                r2= (-b - sqrt(d)) / (2*a);
                printf("The roots are %f and %f",r1,r2);
        }
        else
        {
                printf("Roots are imaginary\n");
                r1 = -b / (2*a);
                r2 = sqrt(fabs(d)) / (2*a);
                printf("The roots are %f+i%f",r1,r2);
                printf("The roots are %f+i%f",r1,r2);
        }
        getch();
}
```

Enter the 3 coefficients:

1 5 2

Roots are real and distinct

The roots are -0.438447 and -4.561553

Enter the 3 coefficients:

4 4 1

Roots are real and equal

The roots are -0.500000 and -0.500000

Enter the 3 coefficients:

5 2 4

Roots are imaginary

The roots are -0.200000+i0.871780The roots are -0.200000+i0.871780

3. **An electricity board charges the following rates for the use of electricity: for the first 200 units 80 paise per unit: for the next 100 units 90 paise per unit: beyond 300 units Rs 1 per unit. All users are charged a minimum of Rs. 100 as meter charge. If the total amount is more than Rs 400, then an additional surcharge of 15% of total amount is charged. Write a program to read the name of the user, number of units consumed and print out the charges.**

```
#include <stdio.h>
#include<conio.h>
void main()
{
        char name[10];
        float unit, charge=100;
        clrscr();
        printf("Enter your name and unit Consumed:");
        scanf("%s %f",name,&unit);
        if(unit<=200)
                charge=charge+unit*0.80;
        else if(unit<=300)
                charge=charge+(unit-200)*0.90+160;
        else
                charge=charge+(unit-300)*1+1250;
        if(charge>400)
                charge=charge+charge*0.15;
        printf("Name:%s\ncharge:%f\n ",name,charge);
        getch();
}
```

Output:

Enter your name: Devansh	*Enter your name: Seena*
Enter No. of unit Consumed:175	*Enter No. of unit Consumed:445*
Name:Devansh	**Name:Seena**
charge:240.000000	**charge:1719.250000**
Enter your name: Manasa	*Enter your name: Devu*
Enter No. of unit Consumed:238	*Enter No. of unit Consumed:337*
Name:Manasa	**Name:Devu**
charge:294.200012	**charge:1595.050049**

4. **To display the following by reading the number of rows as input.**

```
#include<stdio.h>
#include<conio.h>
void main()
{
        int i,j,n;
        printf("Input number of rows:");
        scanf("%d",&n);
        for(i=0;i<=n;i++)
        {
                /*print blank space*/
                for(j=1;j<=n-i;j++)
                printf(" ");
                /*Numbers in ascending order*/
                for(j=1;j<=i;j++)
                printf("%d",j);
                /*Reverse order*/
                for(j=i-1;j>=1;j--)
                printf("%d",j);
                printf("\n");
        }
        getch();
}
```

Output:

```
Input number of rows:5
    1
   121
  12321
 1234321
123454321
```

5. Implement Binary Search on Integers.

```
#include<stdio.h>
#include<conio.h>
 void main()
{
        int n, a[100], i, key, high, low, mid, loc=-1;
        printf("Enter the size of the array\n");
        scanf("%d",&n);
        printf("Enter the elements of array in sorted order\n");
        for(i=0;i<n;i++)
        scanf("%d",&a[i]);
        printf("Enter the key element to be searched\n");
         scanf("%d",&key);
        low=0;
        high=n-1;
        while(low<=high)
        {
                mid=(low+high)/2;
                if(key==a[mid])
                {
                        loc = mid+1;
                        break;
                }
                else
                {
                        if(key<a[mid])
                                high=mid-1;
                        else
                                  low=mid+1;
                }
        }
         if(loc>0)
                printf("\n The element %d is found at %d ",key,loc);
         else
                printf("\nThe search is unsuccessful");
        getch();
}
```

Output:

```
Enter the size of the array
5
Enter the elements of array in sorted order
10 20 30 40 50
Enter the key element to be searched
40
 The element 40 is found at 4
```

6. Implement Matrix multiplication and validate the rules of multiplication.

```
#include<stdio.h>
#include<stdlib.h>
#include<conio.h>
void main( )
{
        int m,n,p,q,row,col,k,a[3][3],b[3][3],c[3][3];
        clrscr();
        printf("Enter the order of matrix A\n");
        scanf("%d%d",&m,&n);
        printf("enter order of matrix B\n");
        scanf("%d%d",&p,&q);
        if(n!=p)
        {
                printf("Matrix Multiplication is not possible\n");
                exit(0);
        }
        printf("Enter the elements into matrix A\n");
        for(row=0;row<m;row++)
        {
                for(col=0;col<n;col++)
                {
                        scanf("%d",&a[row][col]);
                }
        }
        printf("Enter the elements into matrix B\n");
        for(row=0;row<p;row++)
        {
                for(col=0;col<q ;col++)
                {
                        scanf("%d",&b[row][col]);
                }
        }
        for(row=0;row<m;row++)
        {
                for(col=0;col<q;col++)
                {
                        c[row][col]=0;
                        for(k=0;k<n;k++)
                        {
                                c[row][col]= c[row][col]+a[row][k]*b[k][col];
                        }
                }
        }
        printf("The elements of matrix A are\n");
        for(row=0;row<m;row++)
        {
```

```
            for(col=0;col<n;col++)
            {
                printf("%d",a[row][col]);
            }
                printf("\n");
        }
        printf("The elements of matrix B are\n");
        for(row=0;row<p;row++)
        {
            for(col=0;col<q;col++)
            {
                printf("%d",b[row][col]);
            }
            printf("\n");
        }
        printf("Product of Matrix A and B is\n");
        for(row=0;row<m;row++)
        {
            for(col=0;col<q;col++)
            {
                printf("%d",c[row][col]);
            }

            printf("\n");
        }
        getch();
}
```

Output:

```
Enter the order of matrix A
3 3
enter order of matrix B
3 3
Enter the elements into matrix A
1 2 3
1 2 3
1 2 3
Enter the elements into matrix B
3 2 1
3 2 1
3 2 1
```

```
The elements of matrix A are
1  2  3
1  2  3
1  2  3
The elements of matrix B are
3  2  1
3  2  1
3  2  1
Product of Matrix A and B is
18  12  6
18  12  6
18  12  6
```

7. **Compute sin(x)/cos(x) using Taylor series approximation. Compare your result with the built-in library function. Print both the results with appropriate inferences.**

```
#include<stdio.h>
#include<math.h>
#include<conio.h>
int main( )
{
        int i;
        float x,t,sum,sum1,y;
        clrscr();
        printf("Enter the angle\n");
        scanf("%f",&x);
        y=x;
        x=3.1428*(x/180.0);
        sum=x;
        t=x;
        i=1;
        do
        {
                i=i+2;
                t=(-t*x*x)/((i-1)*i);
                sum=sum+t;
        }while(fabs(t)>0.00005);
        printf("sin(%f) using taylor series=%f\n",y,sum);
        sum1=sin(x);
        printf("Using inbuilt function sin(%f)=%f",y,sum1);
        getch();
}
```

Output:

Enter the angle
45
Sin (45.000000) using taylor series=0.707320
Using inbuilt function sin(45.000000)=0.707320

8. Sort the given set of N numbers using Bubble sort.

```
#include<stdio.h>
#include<conio.h>
void main()
{
        int i,j,n,temp;
        int a[20];
        clrscr();
        printf("enter the value of n");
        scanf("%d",&n);
        printf("Enter the numbers in unsorted order:\n");
        for(i=0;i<n;i++)
        scanf("%d", &a[i]);
        // bubble sort logic
        for(i=0;i<n;i++)
        {
                for(j=0;j<(n-i)-1;j++)
                {
                        if( a[j]>a[j+1])
                        {
                                temp=a[j];
                                a[j]=a[j+1];
                                a[j+1]=temp;
                        }
                }
        }
        printf("The sorted array is\n");
        for(i=0;i<n;i++)
        {
                printf("%d\n",a[i]);
        }
        getch();
}
```

Output:

```
enter the value of n
6
Enter the numbers in unsorted order:
4 5 3 2 1 6
The sorted array is
1
2
3
4
5
6
```

9. **Write functions to implement string operations such as compare, concatenate, and find string length. Use the parameter passing techniques.**

```
#include<stdio.h>
#include<string.h>
#include<conio.h>
void compare(char [ ],char [ ]);
void concat(char [ ],char [ ]);
void length(char *[ ]);
void main( )
{
        int n,digit;
        clrscr();
        char str1[10],str2[10];
        do
        {
                printf("press 1-compare 2-concatenate 3-length of string");
                printf("\n enter your choice= ");
                scanf("%d",&n);
                switch(n)
                {
                        case 1:printf("enter first string=");
                                scanf("%s",str1);
                                printf("enter second string=");
                                scanf("%s",str2);
                                compare(str1,str2);
                                break;
                        case 2: printf("enter first string=");
                                scanf("%s",str1);
                                printf("enter second string=");
                                scanf("%s",str2);
                                concat(str1,str2);
                                break;
                        case 3:printf("enter string=");
                                scanf("%s",str1);
                                length(&str1);
                                break;
                        default: printf("wrong choice");
                        break;
                }
                printf("\n Do you want to continue(1/0)? ");
                scanf("%d", &digit);
        }while(digit==1);
        getch();
}
void compare(char str1[ ],char str2[ ])
{
        int i;
```

```
        i=strcmp(str1,str2);
        if(i==0)
                printf("strings are equal\n ");
        else
                printf("string are not equal\n");
}
void concat(char str1[ ],char str2[ ])
{
        strcat(str1,str2);
        printf("concatenate string=%s",str1);
}
void length(char *str1[ ])
{
        int len;
        len=strlen(str1);
        printf("the length of string=%d",len);
}
```

Output:

```
press 1-compare 2-concatenate 3-length of string
 enter your choice= 1
enter first string=Seena
enter second string=Seena
strings are equal

 Do you want to continue(1/0)? 1
press 1-compare 2-concatenate 3-length of string
 enter your choice= 2
enter first string=Devansh
enter second string=Manasa
concatenate string=DevanshManasa

 Do you want to continue(1/0)? 1
press 1-compare 2-concatenate 3-length of string
 enter your choice= 3
enter string=Dhanalakshmi
the length of string=12

 Do you want to continue(1/0)? 0
```

10. Implement structures to read, write and compute average- marks of the students, list the students scoring above and below the average marks for a class of N students.

```
#include<stdio.h>
#include<conio.h>
struct student
{
        char usn[10];
        char name[10];
        float m1,m2,m3;
        float avg,total;
};
void main()
{
        struct student s[20];
        int n,i;
        float avg,sum=0.0;
        clrscr();
        printf("Enter the number of student=");
        scanf("%d",&n);
        for(i=0;i<n;i++)
        {
                printf("Enter the detail of %d students\n",i+1);
                printf("\n Enter USN=");
                scanf("%s",s[i].usn);
                printf("\n Enter Name=");
                scanf("%s",s[i].name);
                printf("Enter the three subject score\n");
                scanf("%f%f%f",&s[i].m1,&s[i].m2,&s[i].m3);
                s[i].total=s[i].m1+s[i].m2+s[i].m3;
                s[i].avg=s[i].total/3;
        }
        for(i=0;i<n;i++)
        {
                if(s[i].avg>=35)
                        printf("\n %s has scored above the average marks",s[i].name);
                 else
                        printf("\n %s has scored below the average marks",s[i].name);
        }
         getch();
}
```

Output:

```
Enter the number of student=3
Enter the detail of 1 students

 Enter USN=3BR22DS300

 Enter Name=BITM
Enter the three subject score
87 67 75
Enter the detail of 2 students

 Enter USN=3BR22DS301

 Enter Name=DATASCIENCE
Enter the three subject score
67 98 58
Enter the detail of 3 students

 Enter USN=3BR22DS303

 Enter Name=DUMMY
Enter the three subject score
35 25 29

BITM has scored above the average marks
DATASCIENC has scored above the average marks
DUMMY has scored below the average marks
```

11. Develop a program using pointers to compute the sum,mean and standard deviation of all elements stored in an array of N real numbers.

```
#include<stdio.h>
#include<math.h>
#include<conio.h>
int main()
{
        int n,i;
        float x[20],sum,mean;
        float variance , deviation;
        clrscr();
        printf("Enter the value of n \n");
        scanf("%d",&n);
        printf("enter %d real values \n",n);
        for (i=0;i<n;i++)
        {
                scanf("%f",(x+i));
        }
        sum=0;
        for(i=0;i<n;i++)
        sum= sum+*(x+i);
        printf("sum=%f\n",sum);
        mean=sum/n;
        sum=0;
        for(i=0;i<n;i++)
        {
                sum=sum+(*(x+i)-mean)*(*(x+i)-mean);
        }
        variance = sum/n;
        deviation=sqrt(variance);
        printf("mean(Average)=%f\n",mean);
        printf("variance=%f\n",variance);
        printf("standard deviation=%f\n",deviation);
        retrun 0;
}
```

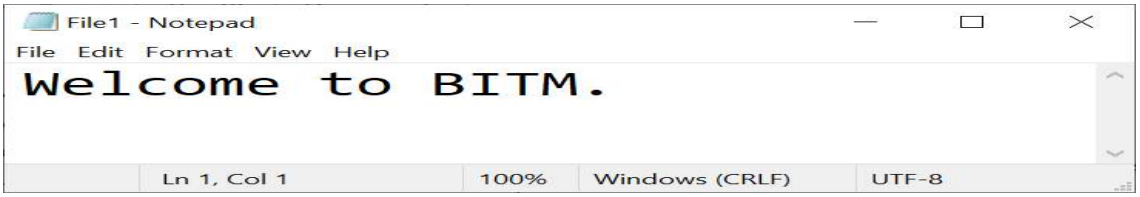
Output:

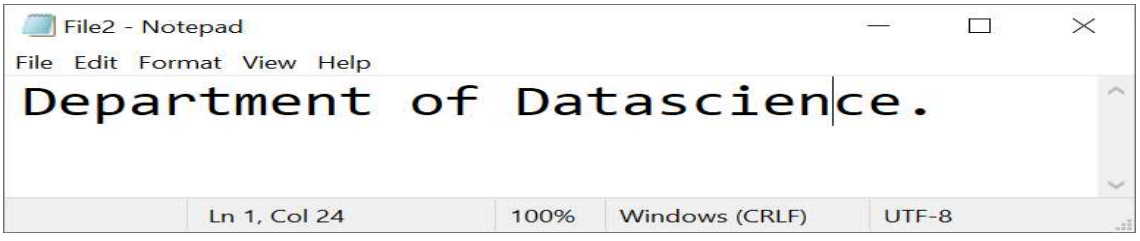
```
Enter the value of n
5
enter 5 real values
4.6     5.7     2.2     3.4     6.1
sum=22.000000
mean(Average)=4.400000
variance=2.092000
standard deviation=1.446375
```

12.Write a C program to copy a text file to another, read both the input file name and target file name.

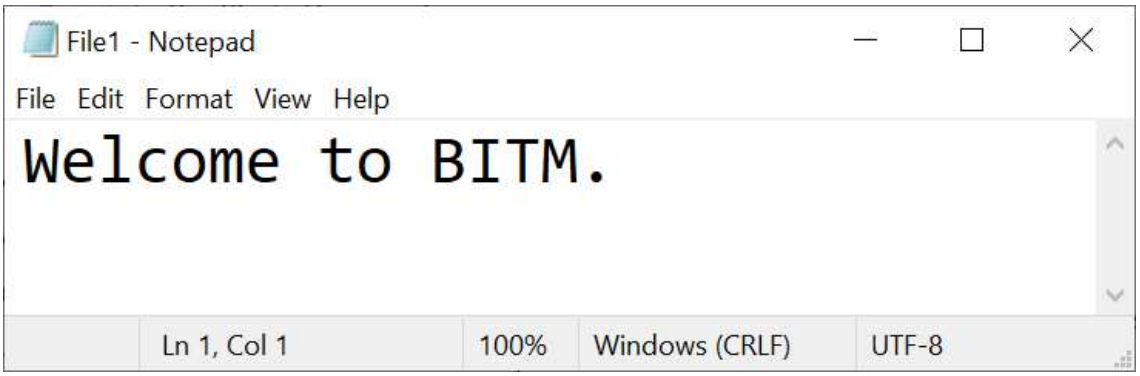
```
#include<stdio.h>
#include<conio.h>
void main()
{
        FILE *fp1,*fp2;
        char ch, sfile[10],dfile[10];
        clrscr();
        printf("Enter the source file\n");
        gets(sfile);
        printf("Enter teh destination file\n");
        gets(dfile);
        fp1=fopen(sfile,"r");
        fp2=fopen(dfile,"w");
        if(fp1==NULL||fp2==NULL)
                printf("File doesnot exists");
        else
        {
                while((ch=fgetc(fp1))!=EOF)
                        fputc(ch,fp2);
                printf("File copied\n");
        }
        getch();
}
```

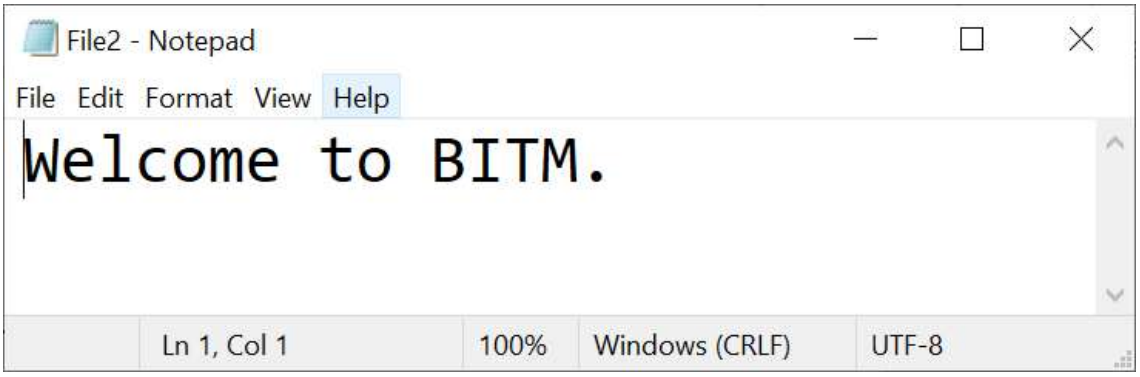
Output:





Enter the source file
File1.txt
Enter teh destination file
File2.txt
File copied





**********End of Lab Program**********