Question 1

Simple Calculator

Simulation of a Simple Calculator.

C Code

```
#include<stdio.h>
#include<stdlib.h>
#include<math.h>

/*************************************************************************
*Function                       :       main
*Input parameters       :       no parameters
*RETURNS                        :       0 on success
*************************************************************************/

int main()
{
        int iChoice, iOperand1, iOperand2;
        char cOperator;

        for(;;)
        {
                printf("\nEnter the arithmetic expression(Do not add spaces in
the expression)\n");
                scanf("%d%c%d", &iOperand1, &cOperator, &iOperand2);
                switch(cOperator)
                {
                        case '+':       printf("\nResult = %d", iOperand1 +
iOperand2);
                                        break;

                        case '-':       printf("\nResult = %d", iOperand1 -
iOperand2);
                                        break;

                        case '*':       printf("\nResult = %d", iOperand1 *
iOperand2);
                                        break;

                        case '/':       printf("\nResult = %g",
(float)iOperand1 / iOperand2);
                                        break;
                        case '%':       printf("\nResult = %d", iOperand1 %
iOperand2);
                                        break;

                }
                printf("\nPress 1 to continue and 0 to quit : ");
```

```
                scanf("%d", &iChoice);
                if(0==iChoice)
                {
                        break;
                }
        }
        return 0;
}
```

Output

```
/*************************************
Enter the arithmetic expression
4+6

Result = 10
Press 1 to continue and 0 to quit : 1

Enter the arithmetic expression
2-9

Result = -7
Press 1 to continue and 0 to quit : 1

Enter the arithmetic expression
5*2

Result = 10
Press 1 to continue and 0 to quit : 1

Enter the arithmetic expression
4/5

Result = 0.8
Press 1 to continue and 0 to quit : 1

Enter the arithmetic expression
8/4

Result = 2
Press 1 to continue and 0 to quit : 1

Enter the arithmetic expression
15%4

Result = 3
Press 1 to continue and 0 to quit : 0
*************************************/
```

Question 2

Quadratic Equation

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

C Code

```
/*****************************************************************************
*File                   : A02Quadratic.c
*Description    : Program to find the roots of a Quadratic Equation
*Author         : Prabodh C P
*Compiler               : gcc compiler, Ubuntu 22.04
*Date                   : 10 August 2022
*****************************************************************************/

#include<stdio.h>
#include<stdlib.h>
#include<math.h>
/*****************************************************************************
*Function                       :       main
*Input parameters       :       no parameters
*RETURNS                        :       0 on success
*****************************************************************************/
int main(void)
{
        float fA,fB,fC,fDesc,fX1,fX2,fRealp,fImagp;
        int iState;

        printf("\n*************************************************************
*");
        printf("\n*\tPROGRAM TO FIND ROOTS OF A QUADRATIC EQUATION\t    *\n");
        printf("*************************************************************"
);

        printf("\nEnter the coefficients of a,b,c \n");
        scanf("%f%f%f",&fA,&fB,&fC);
        if(0 == fA)
```

```
        {
                printf("\nInvalid input, not a quadratic equation - try
again\n");
                exit(0);
        }

        /*COMPUTE THE DESCRIMINANT*/
        fDesc = fB * fB - 4 * fA * fC;
        ((0 == fDesc) ? (iState = 1):((fDesc > 0) ? (iState = 2) : (iState =
3)));
        switch(iState)
        {
                case 1:
                        fX1 = fX2 = -fB/(2*fA);
                        printf("\nRoots are equal and the Roots are \n");
                        printf("\nRoot1 = %g and Root2 = %g\n",fX1,fX2);
                        break;
                case 2:
                        fX1 = (-fB+sqrt(fDesc))/(2*fA);
                        fX2 = (-fB-sqrt(fDesc))/(2*fA);
                        printf("\nThe Roots are Real and distinct, they are
\n");
                        printf("\nRoot1 = %g and Root2 = %g\n",fX1,fX2);
                        break;
                case 3:
                        fRealp = -fB / (2*fA);
                        fImagp = sqrt(fabs(fDesc))/(2*fA);
                        printf("\nThe Roots are imaginary and they are\n");
                        printf("\nRoot1 = %g+i%g\n",fRealp,fImagp);
                        printf("\nRoot2 = %g-i%g\n",fRealp,fImagp);
        }
        return 0;
}
```

Output

```
/*************************************
************************************************************
*       PROGRAM TO FIND ROOTS OF A QUADRATIC EQUATION      *
************************************************************
Enter the coefficients of a,b,c
1 -5 6

The Roots are Real and distinct, they are

Root1 = 3 and Root2 = 2
************************************************************
*       PROGRAM TO FIND ROOTS OF A QUADRATIC EQUATION      *
************************************************************
Enter the coefficients of a,b,c
1 4 4

Roots are equal and the Roots are

Root1 = -2 and Root2 = -2
************************************************************
```

```
*       PROGRAM TO FIND ROOTS OF A QUADRATIC EQUATION     *
************************************************************
Enter the coefficients of a,b,c
1 3 3

The Roots are imaginary and they are

Root1 = -1.5+i0.866025        Root2 = -1.5-i0.866025
************************************************************
*       PROGRAM TO FIND ROOTS OF A QUADRATIC EQUATION     *
************************************************************
Enter the coefficients of a,b,c
0 1 2

Invalid input, not a quadratic equation - try again
************************************/
```

Question 3

Electricity Bill

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.

C Code

```c
/*****************************************************************************
*File                   : A03ElectricBill.c
*Description    : Program to perform a binary search on 1D Array
*Author         : Prabodh C P
*Compiler               : gcc compiler, Ubuntu 22.04
*Date                   : 10 August 2022
*****************************************************************************/
#include<stdio.h>
#include<stdlib.h>
/*****************************************************************************
*Function               :       main
*Input parameters       :       no parameters
*RETURNS                :       0 on success
*****************************************************************************/
int main(void)
{
        char cName[30];
        int iUnits;
        const int iMinCharge = 100;
        const double dSlab1 = 0.8;
        const double dSlab2 = 0.9;
        const double dSlab3 = 1.0;
        const double dSurcharge = 0.15;
```

```
        double dBillAmount = 0.0;

        printf("\nEnter the name of the customer : ");
        scanf("%s", cName);
        printf("\nEnter the number of units consumed : ");    scanf("%d",
&iUnits);
        dBillAmount += iMinCharge;
        if(iUnits <= 200)
        {
                dBillAmount += iUnits*dSlab1;
        }
        else if(iUnits > 200 && iUnits <= 300)
        {
                dBillAmount += (200*dSlab1)+((iUnits-200)*dSlab2);
        }
        else
        {
                dBillAmount += (200*dSlab1)+(100*dSlab2)+((iUnits-
300)*dSlab3);
        }
        if(dBillAmount > 400)
        {
                dBillAmount += dBillAmount * dSurcharge;
        }
        printf("\nElectricity Bill\n=====================================");
        printf("\nCustomer Name\t: %s", cName);
        printf("\nUnits Consumed\t: %d", iUnits);
        printf("\nBill Amount\t: %0.2lf Rupees\n\n", dBillAmount);
        return 0;
}
```

Output

```
/*************************************
Enter the name of the customer : Ramesh
Enter the number of units consumed : 457

Electricity Bill
=================================
Customer Name   : Ramesh
Units Consumed  : 457
Bill Amount     : 583.05 Rupees

Enter the name of the customer : Sayeed
Enter the number of units consumed : 150

Electricity Bill
=================================
Customer Name   : Sayeed
Units Consumed  : 150
Bill Amount     : 220.00 Rupees

Enter the name of the customer : Jaswinder
Enter the number of units consumed : 300

Electricity Bill
```

```
==================================
Customer Name   : Jaswinder
Units Consumed  : 300
Bill Amount     : 350.00 Rupees

Enter the name of the customer : Shyam
Enter the number of units consumed : 182

Electricity Bill
==================================
Customer Name   : Shyam
Units Consumed  : 182
Bill Amount     : 245.60 Rupees
************************************/
```

Output

```
/************************************
Enter Customer name:SHAM

Enter number of units consumed: 150
ELECTRIC BILL
Name :                    SHAM
Units Consumed :          150 units
Bill Minimum             =  100.00 Rs
        150 *  80 paise     =  120.00
Total Bill Charges       =  220.00 Rs
==================================

Enter Customer name:BABU

Enter number of units consumed: 260
ELECTRIC BILL
Name :                    BABU
Units Consumed :          260 units
Bill Minimum             =  100.00 Rs
        200 *  80 paise     =  160.00
         60 *  90 paise     =   54.00
Total Bill Charges       =  314.00 Rs
==================================

Enter Customer name:RAMU

Enter number of units consumed: 555
ELECTRIC BILL
Name :                    RAMU
Units Consumed :          555 units
Bill Minimum             =  100.00 Rs
        200 *  80 paise     =  160.00 Rs
        100 *  90 paise     =   90.00 Rs
        255 * 100 paise     =  255.00 Rs
Surcharge Amount         =  104.36 Rs
Total Bill Charges       =  695.75 Rs
==================================

************************************/
```

Question 4

Printing Pattern

Write a C Program to display the following by reading the number of rows as input

C Code

```
/*************************************************************************
*File                   : A04Pattern.c
*Description    : Program to print a Pattern
*Author                 : Prabodh C P
*Compiler               : gcc compiler, Ubuntu 22.04
*Date                   : 26 December 2022
*************************************************************************/
#include<stdio.h>
#include<stdlib.h>
/*************************************************************************
*Function               :       main
*Input parameters       :       no parameters
*RETURNS                :       0 on success
*************************************************************************/

int main()
{
    int iNum, iSp_cnt, iNum_cnt, iVal, i, j, k;
    printf("Enter the number of rows : ");
    scanf("%d", &iNum);
    iSp_cnt = iNum - 1;
    iNum_cnt = 1;

    for(i=0;i<iNum;i++)
    {
        iVal = 1;
        for(j=0;j<iSp_cnt;j++)
        {
            printf("  ");
        }
        for(k=0;k<iNum_cnt;k++)
```

```
        {
            if(k <= iNum_cnt/2)
            {
                printf("%d ", iVal);
                iVal++;
            }
            else
            {
                iVal--;
                printf("%d ", iVal-1);
            }
        }
        printf("\n");
        iSp_cnt--;
        iNum_cnt += 2;
    }
    return 0;
}
```

Output

```
/*****************************

Enter the number of rows : 3
    1
  1 2 1
1 2 3 2 1

Enter the number of rows : 4
      1
    1 2 1
  1 2 3 2 1
1 2 3 4 3 2 1

Enter the number of rows : 5
        1
      1 2 1
    1 2 3 2 1
  1 2 3 4 3 2 1
1 2 3 4 5 4 3 2 1
*************************************/
```

Question 5

Binary Search

Implement Binary Search on Integers.

C Code

```c
/*****************************************************************************
*File                   : A05BinarySearchIntegers.c
*Description    : Program to perform a binary search on 1D Array of Integers
*Author         : Prabodh C P
*Compiler               : gcc compiler, Ubuntu 22.04
*Date                   : 10 August 2022
*****************************************************************************/

#include<stdio.h>
#include<stdlib.h>

/*****************************************************************************
*Function                       :       main
*Input parameters       :       no parameters
*RETURNS                        :       0 on success
*****************************************************************************/

int main(void)
{
        int iaArr[100],iNum,i,iMid,iLow,iHigh,iFound,iKey;

        printf("\nEnter the number of elements\n");
        scanf("%d",&iNum);

        printf("\nEnter the elements in ascending order\n");
        for(i=0;i<iNum;i++)
                scanf("%d",&iaArr[i]);

        printf("\nEnter the key element\n");
        scanf("%d",&iKey);

        iFound = 0;
        iLow = 0;
        iHigh = iNum-1;
```

```
        while(iLow <= iHigh)
        {
                iMid = (iLow+iHigh)/2;
                if(iKey == iaArr[iMid])
                {
                        iFound = 1;
                        break;
                }
                else if(iKey < iaArr[iMid])
                {
                        iHigh = iMid-1;
                }
                else
                {
                        iLow = iMid+1;
                }
        }

        if(iFound)
                printf("\nKey element %d found at position %d\n",iKey,iMid+1);
        else
                printf("\nKey element not found\n");

        return 0;
}
```

Output

```
/*****************************
Enter the number of elements
5

Enter the elements in ascending order
1 3 5 7 9

Enter the key element
8

Key element not found
================================================
Enter the number of elements
4

Enter the elements in ascending order
1 4 5 7

Enter the key element
5

Key element 5 found at position 3
================================================
Enter the number of elements
6
Enter the elements in ascending order
2 4 6 8 9 10
```

```
Enter the key element
9

Key element 9 found at position 5
*************************************/
```

Question 6

Matrix Multiplication

Implement Matrix multiplication and validate the rules of multiplication.

C Code

```
/*****************************************************************************
*File                   : A06MatrixMul.c
*Description    : Program to implement Matrix Multiplication
*Author         : Prabodh C P
*Compiler               : gcc compiler, Ubuntu 22.04
*Date                   : 10 August 2022
*****************************************************************************/

#include<stdio.h>
#include<stdlib.h>

/*****************************************************************************
*Function                       :       main
*Input parameters       :       no parameters
*RETURNS                        :       0 on success
*****************************************************************************/

int main(void)
{
        int iM, iN, iP, iQ, i, j, k, iaMat1[10][10], iaMat2[10][10];
        int iaProd[10][10] = {0};


        printf("\n*********************************************************");
        printf("\n*\tPROGRAM TO IMPLEMENT MATRIX MULIPLICATION\t*\n");
        printf("*********************************************************");

        printf("\nEnter the order of Matrix1\n");
        scanf("%d%d",&iM,&iN);

        printf("\nEnter the order of Matrix2\n");
        scanf("%d%d",&iP,&iQ);

        if( iN != iP)
```

```
        {
                printf("\nMatrix Multiplication not possible\n");
                exit(0);
        }


        printf("\nEnter the elements of Matrix 1\n");
        for(i=0;i<iM;i++)
                for(j=0;j<iN;j++)
                        scanf("%d",&iaMat1[i][j]);

        printf("\nEnter the elements of Matrix 2\n");
        for(i=0;i<iP;i++)
                for(j=0;j<iQ;j++)
                        scanf("%d",&iaMat2[i][j]);


        for(i=0;i<iM;i++)
        {
                for(j=0;j<iQ;j++)
                {
                        for(k=0;k<iN;k++)
                        {
                                iaProd[i][j] += iaMat1[i][k] * iaMat2[k][j];
                        }
                }
        }

/**************************************************************************
**********************
                |*|             |*|
a00  a01  a02|*|b00  b01  b02|*|
                |*|             |*|
a10  a11  a12|*|b10  b11  b12|*|
                |*|             |*|
a20  a21  a22|*|b20  b21  b22|*|
                |*|             |*|

(a00*b00+a01*b10+a02*b20) (a00*b01+a01*b11+a02*b21) (a00*b02+a01*b12+a02*b22)
(a10*b00+a11*b10+a12*b20) (a10*b01+a11*b11+a12*b21) (a10*b02+a11*b12+a12*b22)
(a20*b00+a21*b10+a22*b20) (a20*b01+a21*b11+a22*b21) (a20*b02+a21*b12+a22*b22)
**************************************************************************
********************/


        printf("\nMatrix 1\n");

        for(i=0;i<iM;i++)
        {
                for(j=0;j<iN;j++)
                {
                        printf("%d\t",iaMat1[i][j]);
                }
                printf("\n");
        }
        printf("\n");
```

```
        printf("\nMatrix 2\n");

        for(i=0;i<iP;i++)
        {
                for(j=0;j<iQ;j++)
                {
                        printf("%d\t",iaMat2[i][j]);
                }
                printf("\n");
        }
        printf("\n");

        printf("\nThe Product matrix is is \n");

        for(i=0;i<iM;i++)
        {
                for(j=0;j<iQ;j++)
                {
                        printf("%d\t",iaProd[i][j]);
                }
                printf("\n");
        }
        printf("\n");
        return 0;
}
```

Output

```
/*************************************

*********************************************************
*       PROGRAM TO IMPLEMENT MATRIX MULIPLICATION     *
*********************************************************
Enter the order of Matrix1
2 3

Enter the order of Matrix2
4 5

Matrix Multiplication not possible

*********************************************************
*       PROGRAM TO IMPLEMENT MATRIX MULIPLICATION     *
*********************************************************
Enter the order of Matrix1
2 3

Enter the order of Matrix2
3 2

Enter the elements of Matrix 1
1 2 3
4 5 6

Enter the elements of Matrix 2
1 2
```

```
3 4
5 6

Matrix 1
1       2       3
4       5       6

Matrix 2
1       2
3       4
5       6

The Product matrix is is
22      28
49      64

*********************************************************
*       PROGRAM TO IMPLEMENT MATRIX MULIPLICATION     *
*********************************************************
Enter the order of Matrix1
2 2

Enter the order of Matrix2
2 2

Enter the elements of Matrix 1
1 2
3 4

Enter the elements of Matrix 2
1 0
0 1

Matrix 1
1       2
3       4

Matrix 2
1       0
0       1

The Product matrix is is
1       2
3       4
*************************************/
```

Question 7

Compute Sine and Cosine of an Angle

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.

C Code

```c
/*****************************************************************************
*File                   : A07SineCosAngle.c
*Description    : Program to calculate Sin(x)/Cos(x) using Taylor series
*Author         : Prabodh C P
*Compiler               : gcc compiler, Ubuntu 22.04
*Date                   : 10 August 2022
*****************************************************************************/
#include<stdio.h>
#include<stdlib.h>
#include <math.h>
/*****************************************************************************
*Function                       :       main
*Input parameters       :       no parameters
*RETURNS                        :       0 on success
*****************************************************************************/
int main()
{
        float fAngD, fAngR;
        float fTerm, fNum, fDen, fSVal,fCVal;
        int i,iNum;
        printf("\nEnter the Angle : ");       scanf("%f",&fAngD);
        printf("\nEnter the Number of terms : ");
        scanf("%d",&iNum);
        printf("\nInput Angle = %g\n",fAngD);
        printf("No of terms = %d\n",iNum);

        fAngR= (fAngD*M_PI)/180 ;

        //Calculation of Sine of an angle using Taylor's series
        fNum=fAngR;
        fDen=1.0;
        fSVal =0.0;
        fTerm=fNum/fDen;
```

```
        for(i=1;i<=iNum;i++)
        {
                fSVal = fSVal + fTerm;
                fNum = -fNum * fAngR * fAngR ;
                fDen = fDen * (2*i) * (2*i+1);
                fTerm = fNum/fDen;
        }

        //Calculation of Cosine of an angle using Taylor's series
        fNum=1.0;
        fDen=1.0;
        fCVal =0.0;
        fTerm=1.0;
        for(i=1;i<=iNum;i++)
        {
                fCVal = fCVal + fTerm;
                fNum = -fNum * fAngR * fAngR ;
                fDen = fDen * (2*i) * (2*i-1);
                fTerm = fNum/fDen;
        }

        printf("\nCalculated value is :\nSin(%g)/Cos(%g) = %g\n",fAngD, fAngD,
fSVal/fCVal);
        printf("\nBuilt In function value is :\nSin(%g)/Cos(%g) = %g\n",
fAngD, fAngD, sin(fAngR)/cos(fAngR));

        return 0;
}
```

Output

```
/**************************************

Enter the Angle : 60
Enter the Number of terms : 12

Input Angle = 60        No of terms = 12
Calculated value is :
Sin(60)/Cos(60) = 1.73205

Built In function value is :
Sin(60)/Cos(60) = 1.73205
==========================================

Enter the Angle : 30
Enter the Number of terms : 3

Input Angle = 30        No of terms = 3
Calculated value is :
Sin(30)/Cos(30) = 0.577334

Built In function value is :
Sin(30)/Cos(30) = 0.57735
==========================================

Enter the Angle : 45
```

```
Enter the Number of terms : 11

Input Angle = 45       No of terms = 11
Calculated value is :
Sin(45)/Cos(45) = 1

Built In function value is :
Sin(45)/Cos(45) = 1
************************************/
```

Question 8

Bubble sort.

Sort the given set of N numbers using Bubble sort.

C Code

```
/*****************************************************************************
*File                   : A08BubbleSort.c
*Description    : Program to implement Bubble Sort Algorithm
*Author         : Prabodh C P
*Compiler               : gcc compiler, Ubuntu 22.04
*Date                   : 10 August 2022
*****************************************************************************/

#include<stdio.h>
#include<stdlib.h>

/*****************************************************************************
*Function                       :       main
*Input parameters       :       no parameters
*RETURNS                        :       0 on success
*****************************************************************************/

int main(void)
{
        int iNum, i, j, iaArr[10], iTemp;


        printf("\n************************************************");
        printf("\n*\tPROGRAM TO IMPLEMENT BUBBLE SORT\t*\n");
        printf("************************************************");

        printf("\nEnter no of elements\n");
        scanf("%d",&iNum);

        printf("\nEnter the elements\n");
        for(i=0;i<iNum;i++)
                scanf("%d",&iaArr[i]);
```

```
        for(i=0;i<iNum;i++)
        {
                for(j=0;j<iNum-i-1;j++)
                {
                        if(iaArr[j] > iaArr[j+1])
                        {
                                iTemp = iaArr[j];
                                iaArr[j] = iaArr[j+1];
                                iaArr[j+1] = iTemp;
                        }

/*Code to show the program trace*/
/*                      printf("\nIteration i=%d, j=%d\n",i,j);*/
/*                      for(k=0;k<iNum;k++)*/
/*                              printf("%d\t",iaArr[k]);*/
                }

        }

        printf("\nThe Sorted array is \n");

        for(i=0;i<iNum;i++)
                printf("%d\t",iaArr[i]);

        printf("\n");
        return 0;
}
```

Output

```
/**************************************
***********************************************
*       PROGRAM TO IMPLEMENT BUBBLE SORT      *
***********************************************
Enter no of elements
5

Enter the elements
2 1 6 5 7

The Sorted array is
1       2       5       6       7

***********************************************
*       PROGRAM TO IMPLEMENT BUBBLE SORT      *
***********************************************
Enter no of elements
6

Enter the elements
9 7 5 3 1 0

The Sorted array is
0       1       3       5       7       9
**************************************/
```

Question 9

String Operations

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

C Code

```
/*************************************************************************
*File                   : A09StringFunctions.c
*Description    : Program to implement string operations as functions
*Author         : Prabodh C P
*Compiler               : gcc compiler, Ubuntu 22.04
*Date                   : 10 August 2022
*************************************************************************/

#include<stdio.h>
#include<stdlib.h>

int fnMyStrCmp(const char*, const char*);
void fnMyStrCat(char*, const char*);
int fnMyStrLen(const char*);

/*************************************************************************
*Function               :       main
*Input parameters       :       no parameters
*RETURNS                :       0 on success
*************************************************************************/

int main()
{
        int iChoice;
        char acStr1[30], acStr2[30];
        int iLen;
        printf("\n====================\n");
        printf("STRING OPERATIONS");
        printf("\n====================\n");
        for(;;)
        {
                printf("\nEnter two strings\n");
                printf("\nString 1 : ");       scanf("%s", acStr1);
```

```
                printf("\nString 2 : ");       scanf("%s", acStr2);
                printf("\n1.String Compare\n2.String Concatenate\n3.String
Length");
                printf("\nEnter your choice : ");      scanf("%d", &iChoice);
                switch(iChoice)
                {
                        case 1: if(fnMyStrCmp(acStr1, acStr2) == 0)
                                                printf("\nTwo strings are
equal");
                                        else if(fnMyStrCmp(acStr1, acStr2) > 0)
                                                printf("\nString %s is greater
than String %s", acStr1, acStr2);
                                        else
                                                printf("\nString %s is greater
than String %s", acStr2, acStr1);
                                        break;

                        case 2: fnMyStrCat(acStr1, acStr2);
                                        printf("\nConcatenated String is\n%s",
acStr1);
                                        break;

                        case 3: iLen = fnMyStrLen(acStr1);
                                        printf("\nLength of String %s is %d\n",
acStr1, iLen);
                                        iLen = fnMyStrLen(acStr2);
                                        printf("\nLength of String %s is %d\n",
acStr2, iLen);
                                        break;

                }
                printf("\nPress 1 to continue and 0 to quit : ");
                scanf("%d", &iChoice);
                if(0==iChoice)
                {
                        break;
                }
        }
        return 0;
}

/*****************************************************************************
*Function                       : fnMyStrCmp
*Description            : Function that compares the two strings s1 and s2.
*Input parameters       :
*               const char *s1, const char *s2 - two strings to be compared
*RETURNS                :
*       1 if s1 is greater than s2.
*       0 if s1 matches s2.
*       -1 if s1 is less than s2.
*****************************************************************************/

int fnMyStrCmp(const char *s1, const char *s2)
{
        int k;
        for(k=0; s1[k] == s2[k] && s1[k]!='\0'&& s2[k]!='\0'; k++);
```

```
        if( k==(fnMyStrLen(s1)) && k==(fnMyStrLen(s2)) )
        {
                return 0;
        }
        else if(s1[k] > s2[k])
        {
                return 1;
        }
        else
        {
                return -1;
        }
}

/*****************************************************************************
*Function                       : fnMyStrCat
*Description            : function that appends the src string to the dest
string
*Input parameters       :
*               char *dest - first string
*               const char *src - second string
*RETURNS                        : nothing
*****************************************************************************/

void fnMyStrCat(char *dest, const char *src)
{
        int dest_len, i;
        dest_len = fnMyStrLen(dest);
        for (i = 0 ; src[i] != '\0' ; i++)
                dest[dest_len + i] = src[i];
        dest[dest_len + i] = '\0';
}

/*****************************************************************************
*Function                       : fnMyStrLen
*Description            : function that calculates the length of a string
*Input parameters       :
*               const char *str - string whose length needs to be found
*RETURNS                        :
*       integer which is the length of the string
*****************************************************************************/

int fnMyStrLen(const char *str)
{
        int iLen;
        for(iLen=0; str[iLen] != '\0'; iLen++);
        return iLen;
}
```

Output

```
/***************************************
=====================
STRING OPERATIONS
=====================
```

```
Enter two strings

String 1 : shiva

String 2 : shankar

1.String Compare
2.String Concatenate
3.String Length
Enter your choice : 2

Concatenated String is
shivashankar
Press 1 to continue and 0 to quit : 1

Enter two strings

String 1 : ramesh

String 2 : sumesh

1.String Compare
2.String Concatenate
3.String Length
Enter your choice : 1

String sumesh is greater than String ramesh
Press 1 to continue and 0 to quit : 1

Enter two strings

String 1 : sam

String 2 : samantha

1.String Compare
2.String Concatenate
3.String Length
Enter your choice : 3

Length of String sam is 3

Length of String samantha is 8

Press 1 to continue and 0 to quit : 0
************************************/
```

Question 10

C Structures

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

C Code

```c
/*****************************************************************************
*File                   : A10StudentStructure.c
*Description    : Program to implement structure and compute average marks
*Author         : Prabodh C P
*Compiler               : gcc compiler, Ubuntu 22.04
*Date                   : 10 August 2022
*****************************************************************************/

#include<stdio.h>
#include<stdlib.h>
#define STRSIZE 30

typedef struct
{
        char cName[STRSIZE];
        char cUSN[11];
        int iMarks;
}STUDENT_TYPE;

/*****************************************************************************
*Function                       :       main
*Input parameters       :       no parameters
*RETURNS                        :       0 on success
*****************************************************************************/

int main(void)
{
        STUDENT_TYPE students[100];
        int iNum, i;
        double dAvg = 0.0;

        printf("\nEnter the number of students : ");
        scanf("%d", &iNum);
```

```
        printf("\nEnter the Student details\n");
        for(i=0;i<iNum;i++)
        {
                printf("\n################################");
                printf("\nName : ");    scanf("%s", students[i].cName);
                printf("\nUSN : ");     scanf("%s", students[i].cUSN);
                printf("\nMarks : ");   scanf("%d", &students[i].iMarks);
                dAvg += students[i].iMarks;
        }

        dAvg /= iNum;

        printf("\nThe average marks for the class is : %g\n", dAvg);

        for(i=0;i<iNum;i++)
        {
                printf("\n################################");
                printf("\nName\t: %s", students[i].cName);
                printf("\nUSN\t: %s", students[i].cUSN);
                printf("\nMarks\t: %d", students[i].iMarks);
                if(students[i].iMarks < dAvg)
                        printf("\nThe student has scored below average\n");
                else
                        printf("\nThe student has scored above average\n");
        }

        return 0;
}
```

Output

```
/*************************************
Enter the number of students : 4
Enter the Student details

========================================
Name : Raju
USN : 1SI17CS036
Marks : 67
========================================
Name : Michael
USN : 1SI17CS045
Marks : 87
========================================
Name : Sahana
USN : 1SI17CS405
Marks : 77
========================================
Name : Jonathan
USN : 1SI17CS025
Marks : 83

The average marks for the class is : 78.5

========================================
```

```
Name    : Raju
USN     : 1SI17CS036
Marks   : 67
The student has scored below average
=======================================
Name    : Michael
USN     : 1SI17CS045
Marks   : 87
The student has scored above average
=======================================
Name    : Sahana
USN     : 1SI17CS405
Marks   : 77
The student has scored below average
=======================================
Name    : Jonathan
USN     : 1SI17CS025
Marks   : 83
The student has scored above average
*************************************/
```

Question 11

Pointers and Arrays

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

C Code

```
/*****************************************************************************
*File                   : A11MeanVarianceSD.c
*Description    : Program to compute Mean, Variance and Standard Deviation
                                    using pointer to an array
*Author         : Prabodh C P
*Compiler               : gcc compiler, Ubuntu 22.04
*Date                   : 10 August 2022
*****************************************************************************/
#include<stdio.h>
#include<stdlib.h>
#include<math.h>
/*****************************************************************************
*Function                       :       main
*Input parameters       :       no parameters
*RETURNS                        :       0 on success
*****************************************************************************/
int main(void)
{
    int i,iNum;
    float fMean = 0.0f, fVariance = 0.0f, fSd = 0.0f,faArray[100],fSum=0.0f;
    float *fptr;

    printf("\nEnter the number of Values : ");
    scanf("%d",&iNum);
    fptr = faArray;
/*    fptr = (float*)malloc(iNum*sizeof(float));*/
    printf("\nEnter %d values\n", iNum);
    for(i=0; i<iNum; i++)
    {
        scanf("%f",fptr+i);
        fSum += *(fptr+i);              //fSum += fptr[i]; this is also valid
    }
    fMean = fSum/iNum;
```

```
    for(i=0; i<iNum; i++)
    {
        fVariance += (fptr[i] - fMean)*(fptr[i] - fMean);
                //fVariance += (*(fptr+i) - fMean)*(*(fptr+i) - fMean);
    }
    fVariance /= iNum;
    fSd = sqrt(fVariance);
    printf("\nThe values entered are");
    for(i=0; i<iNum; i++)
    {
        printf("\t%g",fptr[i]);         //printf("\n\t%f",*(fptr+i));
    }
    printf("\n");

    printf("\n************************************\n");
    printf("\tSum\t = \t%g\n\tMean\t = \t%g\n\tVariance = \t%g\nStandard
Deviation = \t%g",fSum,fMean,fVariance,fSd);
    printf("\n************************************\n");
    return 0;
}
```

Output

```
/**************************************
Enter the number of Values : 4

Enter 4 values
1.1 2.2 3.3 4.4

The values entered are 1.1     2.2     3.3     4.4

************************************
        Sum      =      11
        Mean     =      2.75
        Variance =      1.5125
Standard Deviation =    1.22984
************************************
============================================================
Enter the number of Values : 5

Enter 5 values
5.345 6.765 7.234 8.675 9.765

The values entered are 5.345   6.765   7.234   8.675   9.765

************************************
        Sum      =      37.784
        Mean     =      7.5568
        Variance =      2.34995
Standard Deviation =    1.53295
************************************
************************************/
```

Question 12

File Copy

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

C Code

```c
/*****************************************************************************
*File                   : A12FileCopy.c
*Description    : Program to copy a text file to another
*Author         : Prabodh C P
*Compiler               : gcc compiler, Ubuntu 22.04
*Date                   : 22 December 2022
*****************************************************************************/

#include<stdio.h>
#include<stdlib.h>

/*****************************************************************************
*Function                       :       main
*Input parameters       :       no parameters
*RETURNS                        :       0 on success
*****************************************************************************/
int main(void)
{
    FILE *fp1,*fp2;
    int ch;
    char fname1[100], fname2[100];
    printf("\nEnter File name to be copied\n");
    scanf("%s",fname1);
    fp1 = fopen(fname1,"r");

    if(fp1 == NULL)
    {
        printf("\nInput File %s doesn't exist\n", fname1);
        exit(0);
    }

    printf("\nEnter target File name\n");
    scanf("%s",fname2);
```

```
    fp2 = fopen(fname2,"w");
    while((ch=fgetc(fp1)) != EOF)
    {
        fputc(ch,fp2);
    }
    printf("\nFile %s successfully created\n",fname2);

    fclose(fp1);
    fclose(fp2);

    return 0;
}
```

Output

```
/************************************************
Enter File name to be copied
out9.c

Enter target File name
out99.c

File out99.c successfully created

================================================

Enter File name to be copied
secret.txt

Input File secret.txt doesn't exist

************************************************/
```