C-PROGRAMMING LABORATORY ASSIGNMENTS

CONTENTS:

Submitted by:-

NAME : Sandeep Kumar Auddy

ADMISSION NUMBER : 19JE0741

SECTION : G

BTech 1st Year (Mining Engineering)

IIT-ISM DHANBAD

(Lab-3)

ASSIGNMENT: 1

Controls and Loops

```c
//=============( QUESTION : 1 : CONTROL_AND_LOOPING)=================
#include <stdio.h>
int main()
{
    char ch;
    float x, y, z;
    float total, avg, small, large;
    int flag = 1;
    do
    {
        printf("\nEnter your choice...(A/B/C/D/E)");
        scanf(" %c", &ch);
        switch(ch)
        {
            case 'A':
                printf("\nEnter 3 numbers:\n");
                scanf("%f%f%f", &x, &y, &z);
                break;
            case 'B':
                total = x+y+z;
                printf("\nTotal = %f",total);
                break;
            case 'C':
                avg = (x+y+z)/3;
                printf("\nAverage = %f",avg);
                break;
            case 'D':
                if(x<=y && x<=z) small = x;
                else if(y<=z) small = y;
                else small = z;
                printf("\nSmallest Number = %f",small);
                break;
            case 'E':
                if(x>=y && x>=z) large = x;
                else if(y>=z) large = y;
                else large = z;
                printf("\nLargest Number = %f",large);
                break;
            default:
                flag = 0;
        }
    }
    while(flag==1);

    return 0;
}
```

OUTPUT – QUESTION : 1 : CONTROL_AND_LOOP

```
Enter your choice...(A/B/C/D/E)A
Enter 3 numbers:
6 7 8

Enter your choice...(A/B/C/D/E)C
Average = 7.000000

Enter your choice...(A/B/C/D/E)A
Enter 3 numbers:
8 88 67

Enter your choice...(A/B/C/D/E)E
Largest Number = 88.000000
********************************************************************************
```

```c
//=============( QUESTION : 2 : CONTROL_AND_LOOPING)=================
#include <stdio.h>

int main()
{
    float unit, effectiveUnit, factor, offset, price;
    printf("Enter the consumption Unit :");
    scanf("%f", &unit);
    //Itz a boring task indeed...
    if(unit<=150)
    {
        effectiveUnit = unit;
        factor = 3;
        offset = 0;
    }
    else if(unit<=350)
    {
        effectiveUnit = unit-150;
        factor = 3.75;
        offset = 100;
    }
    else if(unit<=450)
    {
        effectiveUnit = unit-350;
        factor = 4;
        offset = 250;
    }
    else if(unit<=600)
    {
        effectiveUnit = unit-450;
        factor = 4.25;
        offset = 300;
    }
    else
    {
        effectiveUnit = unit-600;
        factor = 5;
        offset = 400;
    }

    price = offset + (factor*effectiveUnit); // y = mx+c
    printf("\nYour Price = Rs.%f",price);
    return 0;
}
```

OUTPUT – QUESTION : 2 : CONTROL_AND_LOOP

```
Enter the consumption Unit :457


Your Price = Rs.329.750000
```

****

```c
//=============( QUESTION : 3 : CONTROL_AND_LOOPING)=================
#include <stdio.h>

int main()
{
    int monthNum;
    printf("Enter the Month Number : ");
    scanf("%d", &monthNum);

    char monthsArray[12][10] = {
        "January", "February", "March", "April", "May", "June", "July", "August",
"September", "October", "November", "December"
    };

    printf("\nThe Name of the month is = %s", (monthsArray + monthNum - 1));
    return 0;
}
```

OUTPUT – QUESTION : 3 : CONTROL_AND_LOOP

```
Enter the Month Number : 7


The Name of the month is = July
```

****

```c
//=============( QUESTION : 4 : CONTROL_AND_LOOPING)=================

#include <stdio.h>

int main()
{
    int num;
    printf("Enter a number : ");
    scanf("%d", &num);

    int temp = num;
    int count = 0;

    if(num==0)
    {
        printf("Num of Digits = %d", 1);
        return 0;
    }
    while(temp>0)
    {
        count++;
        temp/=10;
    }
    printf("Num of Digits = %d", count);

    return 0;
}
```

OUTPUT – QUESTION : 4 : CONTROL_AND_LOOP

```
Enter a number : 67807
Num of Digits = 5
```

```c
//=============( QUESTION : 5 : CONTROL_AND_LOOPING)=================

#include <stdio.h>

int main()
{
    int N = 5;
    for(int i=1; i<=N; i++)printf("* ");
    printf("\n");
    for(int i=1; i<=N-2; i++)
    {
        printf("* ");
        for(int j=1; j<=N-2; j++)printf("  ");
        printf("*\n");
    }
    for(int i=1; i<=N; i++)printf("* ");

    return 0;
}
```

OUTPUT – QUESTION : 5 : CONTROL_AND_LOOP

```
* * * * *
*       *
*       *
*       *
* * * * *
```

****

```c
//=============( QUESTION : 6 : CONTROL_AND_LOOPING)=================

#include <stdio.h>

int main()
{
    int N = 5;
    printf("$ ");
    for(int i=2; i<=N; i++)printf("* ");
    printf("\n");
    for(int i=1; i<=N-2; i++)
    {
        printf("* ");
        for(int j=1; j<=N-2; j++)
        {
            if(i==j)printf("$ ");
            else printf("  ");
        }
        printf("*\n");
    }
    for(int i=1; i<=N-1; i++)printf("* ");
    printf("$");

    return 0;
}
```

OUTPUT – QUESTION : 6 : CONTROL_AND_LOOP

```
$ * * * *
* $     *
*   $   *
*     $ *
* * * * $
```

****

```c
//=============( QUESTION : 7 : CONTROL_AND_LOOPING)=================

#include <stdio.h>

int main()
{
    int N = 5;
    for(int i=1; i<=N; i++)
    {
        for(int j=1; j<=i; j++)printf("* ");
        printf("\n");
    }
    for(int i=N-1; i>=1; i--)
    {
        for(int j=1; j<=i; j++)printf("* ");
        printf("\n");
    }
    return 0;
}
```

OUTPUT – QUESTION : 7 : CONTROL_AND_LOOP

```
*
* *
* * *
* * * *
* * * * *
* * * *
* * *
* *
*
```

****

```c
//=============( QUESTION : 8 : CONTROL_AND_LOOPING)=================
#include <stdio.h>
#define N 5
void main()
{
    int num;
    int arr[N];

    printf("Enter a %d digit number : ", N);
    scanf("%d", &num);
    printf("\n");

    int index = N-1;
    while(num>0)
    {
        arr[index] = num%10;
        num/=10;
        index--;
    }

    //for(int i=0; i<N; i++)printf("%d ", arr[i]);

    for(int i=0; i<N; i++)
    {
        for(int j=i; j<N; j++)printf("%d", arr[j]);
        for(int k=1; k<=i; k++)printf(" ");
        printf("\t");
        for(int j=0; j<=i; j++)printf("%d", arr[j]);
        printf("\n");
    }
}
```

OUTPUT – QUESTION : 8 : CONTROL_AND_LOOP

```
Enter a 5 digit number : 60475


60475    6
0475     60
475      604
75       6047
5        60475
```

****

(Lab-4)

ASSIGNMENT: 2

Arrays and Strings

```c
//=============( QUESTION : 1 : ARRAYS and STRINGS )================
#include <stdio.h>

void main()
{
    int n1, n2;
    int arr1[50];
    int arr2[50];
    int arr[100];

    printf("Enter the size of Array_1 : ");
    scanf("%d", &n1);
    printf("Enter the size of Array_2 : ");
    scanf("%d", &n2);

    printf("Enter Elements into the 1st Array :\n");
    for(int i=0; i<n1; i++)scanf("%d", arr1+i);
    printf("Enter Elements into the 2nd Array :\n");
    for(int i=0; i<n2; i++)scanf("%d", arr2+i);

    //========== MERGING ELEMENTS ================
    int index = 0;
    for(int i=0; i<n1; i++)arr[index++] = arr1[i];
    for(int j=0; j<n2; j++)arr[index++] = arr2[j];

    printf("\nMerged Array in Reversed Order :\n");
    for(int i=index-1; i>=0; i--)printf("%d, ", arr[i]);
}
```

OUTPUT – QUESTION : 1 : ARRAYS

```
Enter the size of Array_1 : 4

Enter the size of Array_2 : 6

Enter Elements into the 1st Array :

2 4 -5 89

Enter Elements into the 2nd Array :

-20 78 7 23 45 -54


Merged Array in Reversed Order :

-54, 45, 23, 7, 78, -20, 89, -5, 4, 2,
```

****

```c
//=============( QUESTION : 2 : ARRAYS and STRINGS )================
#include<stdio.h>
int main()
{
    int a,b,p,q,i,j,choice;
    int matrix1[10][10],matrix2[10][10],add[10][10],transpose1[10][10],transpo
se2[10][10],sum=0,sub=0;
    printf("Enter the number of rows and columns of matrix1\n");
    scanf("%d%d",&a,&b);
    printf("Enter the elements of matrix1\n");
    for(i=0;i<a;i++)
    {
        for(j=0;j<b;j++)
        scanf("%d",&matrix1[i][j]);
    }
    printf("Enter the number of rows and columns of matrix2\n");
    scanf("%d%d",&p,&q);
    printf("Enter the elements of matrix2:\n");
    for(i=0;i<p;i++)
    {
        for(j=0;j<q;j++)
        scanf("%d",&matrix2[i][j]);
    }
    printf("Enter option: 1 for addition, 2 for subtraction, 3 for transpose\n
");
    scanf("%d",&choice);
    switch(choice)
    {
        case 1:
        if(a!=p||b!=q)
        printf("The matrices can not be added\n");
        else
            {
                for(i=0;i<a;i++)
                {
                    for(j=0;j<b;j++)
                    {
                        sum=sum+matrix1[i][j]+matrix2[i][j];
                        add[i][j]=sum;
                        sum=0;
                    }
                }

                for(i=0;i<a;i++)
                {
                    for(j=0;j<q;j++)
                    printf("%d\t",add[i][j]);
                    printf("\n");
```

```
            }
            break;
        }
case 2:
    if(a!=p||b!=q)
    printf("The matrices can not be subtracted\n");
    else
        {
            for(i=0;i<a;i++)
            {
                for(j=0;j<b;j++)
                {
                    sub=sub+matrix1[i][j]-matrix2[i][j];
                    add[i][j]=sub;
                    sub=0;
                }
            }
        }
        for(i=0;i<a;i++)
        {
            for(j=0;j<q;j++)
            printf("%d\t",add[i][j]);
            printf("\n");
        }
case 3:
    {for (i=0;i<a;++i)
    for (j=0;j<b;++j) {
        transpose1[j][i] = matrix1[i][j];
    }
    for (i=0;i<p;++i)
    for (j=0;j<q;++j) {
        transpose2[j][i] = matrix2[i][j];
    }
printf("Transpose of the matrix1:\n");
for (i=0;i<b;++i)
    for (j=0;j<a;++j) {
        printf("%d  ",transpose1[i][j]);
        if(j==a-1)
            printf("\n");
    }
    printf("Transpose of the matrix2:\n");
for (i=0;i<q;++i)
    for (j=0;j<p;++j) {
        printf("%d  ",transpose2[i][j]);
        if(j==a-1)
            printf("\n");
    }
return 0;
```

```
    }
    }
}
```

OUTPUT – QUESTION : 2 : ARRAYS

```
Enter the number of rows and columns of matrix1
2 3
Enter the elements of matrix1
4 5 6
7 8 9
Enter the number of rows and columns of matrix2
2 3
Enter the elements of matrix2:
-4 5 8
7 -10 12
Enter option: 1 for addition, 2 for subtraction, 3 for transpose
2
8       0       -2
0       18      -3
Transpose of the matrix1:
4  7
5  8
6  9
Transpose of the matrix2:
-4  7
5  -10
8  12
```

****

```c
//=============( QUESTION : 3 : ARRAYS and STRINGS )================
#include<stdio.h>
int main()
{
    int arr[10][10];
    int r,c,i,j,sum=0;
    printf("Enter matrix row and column size:\n");
    scanf("%d%d",&r,&c);
    if(r!=c)
        printf("Does not posses a principal diagonal\n");
    else
    {
        printf("Enter array elements:\n");
        for(i=0;i<r;i++)
        {
            for(j=0;j<c;j++)
            scanf("%d",&arr[i][j]);
        }
        for(i=0;i<r;i++)
        {
            for(j=0;j<c;j++)
            if(i>j)
            sum=sum+arr[i][j];
        }
        printf("Sum of elemnets below the principal diagonal are: %d\n",sum);
    }
    return 0;
}
```

OUTPUT – QUESTION : 3 : ARRAYS

```
Enter matrix row and column size:

3 3

Enter array elements:

4 2 5

-3 0 5

0 89 -3

Sum of elemnets below the principal diagonal are: 86
```

****

```c
//=============( QUESTION : 4 : ARRAYS and STRINGS )================

// Dear EXAMINER, please NOTE:
// That, PROGRAMMING is used to make our work easy...
// In this question, we were supposed to take a 2D array called MARKS which co
ntained 20*5 elements...
// Entering 100 values into that matrix was time consuming...
// Hence, a stadnard library function rand() of C from the library called stdl
ib.h was used to generate Random marks,
// and feed them into MARKS array...

#include <stdio.h>
#include <stdlib.h>
#define NUM_STUDENTS  20
#define NUM_SUBJECTS  5
#define PASS_MARKS  50

void main()
{
    int MARKS[NUM_STUDENTS][NUM_SUBJECTS];
    int pass[NUM_STUDENTS];

    //=========================================================================
=
    // AUTO-INPUT ...
    for(int i=0; i<NUM_STUDENTS; i++)
    {
        for(int j=0; j<NUM_SUBJECTS; j++) MARKS[i][j] = 11 + (rand() % 90);
    }
    // DISPLAY MATRIX ...
    for(int i=0; i<NUM_STUDENTS; i++)
    {
        printf("\nStudent: %d: ", i);
        for(int j=0; j<NUM_SUBJECTS; j++)
        {
            printf("%d, " ,MARKS[i][j]);
        }
    }
    // INPUT ...
    // printf("\nEnter the Marks into the matrix...\n");
    // for(int i=0; i<NUM_STUDENTS; i++)
    // {
    //     printf("\nStudent no.: %d  :", i);
    //     for(int j=0; j<NUM_SUBJECTS; j++)
    //     {
    //         scanf("%d", &MARKS[i][j]);
    //     }
    // }
```

```
    //=========================================================================
=
    // DISPLAYING AVERAGE IN EACH SUBJECT ...

    printf("\n\nDISPLAYING AVERAGE IN EACH SUBJECT ...");
    for(int j=0; j<NUM_SUBJECTS; j++)
    {
        int sum = 0;
        for(int i=0; i<NUM_STUDENTS; i++)sum += MARKS[i][j];
        printf("\nAverage in SUBJECT : %d = %f", j, ((sum*1.0)/NUM_STUDENTS));
    }

    //=========================================================================
=
    // DISPLAYING AVERAGE OF EACH STUDENT ...

    printf("\n\nDISPLAYING AVERAGE OF EACH STUDENT ...");
    for(int i=0; i<NUM_STUDENTS; i++)
    {
        int sum = 0;
        for(int j=0; j<NUM_SUBJECTS; j++)sum += MARKS[i][j];

        if(sum >= (50*NUM_SUBJECTS))pass[i] = 1;
        else pass[i] = 0;

        printf("\nAverage of STUDENT : %d = %f", i, ((sum*1.0)/NUM_SUBJECTS));
    }
    //=========================================================================
=
    // DISPLAYING THE NUMBER OF STUDENTS WHO HAVE SCORED BELOW 50 IN THEIR AVE
RAGE...

    printf("\n\nDISPLAYING THE NUMBER OF STUDENTS WHO HAVE SCORED BELOW 50 IN
THEIR AVERAGE...");
    int count = 0;
    for(int i=0; i<NUM_STUDENTS; i++)
    {
        if(pass[i]==0)count++;
    }
    printf("\nNumber of Students who have scored below 50 in their average = %
d", count);

    //=========================================================================
=
    // DISPLAYING THE SCORES OBTAINED BY EVERY STUDENT ...

    printf("\n\nDISPLAYING THE SCORES OBTAINED BY EVERY STUDENT ...");
```

```
    for(int i=0; i<NUM_STUDENTS; i++)
    {
        int sum = 0;
        for(int j=0; j<NUM_SUBJECTS; j++)sum += MARKS[i][j];
        printf("\nScore obtained by STUDENT : %d = %d", i, sum);
    }
}
```

OUTPUT – QUESTION : 4 : ARRAYS

```
Student: 0: 52, 28, 45, 51, 100,

Student: 1: 75, 59, 29, 63, 85,

Student: 2: 46, 76, 72, 98, 72,

Student: 3: 52, 36, 73, 68, 47,

Student: 4: 92, 35, 43, 74, 33,

Student: 5: 63, 62, 97, 19, 16,

Student: 6: 58, 47, 22, 29, 80,

Student: 7: 33, 28, 30, 36, 95,

Student: 8: 94, 62, 13, 14, 44,

Student: 9: 85, 32, 72, 94, 39,

Student: 10: 88, 25, 93, 98, 68,

Student: 11: 90, 94, 32, 90, 69,

Student: 12: 87, 76, 61, 53, 29,

Student: 13: 57, 51, 43, 15, 69,

Student: 14: 97, 56, 61, 80, 81,

Student: 15: 61, 77, 62, 14, 49,

Student: 16: 20, 34, 65, 75, 47,

Student: 17: 61, 27, 97, 82, 39,

Student: 18: 35, 50, 67, 84, 58,

Student: 19: 39, 19, 23, 80, 82,


DISPLAYING AVERAGE IN EACH SUBJECT ...

Average in SUBJECT : 0 = 64.250000

Average in SUBJECT : 1 = 48.700000

Average in SUBJECT : 2 = 55.000000

Average in SUBJECT : 3 = 60.850000

Average in SUBJECT : 4 = 60.100000
```

```
DISPLAYING AVERAGE OF EACH STUDENT ...
Average of STUDENT : 0 = 55.200000
Average of STUDENT : 1 = 62.200000
Average of STUDENT : 2 = 72.800000
Average of STUDENT : 3 = 55.200000
Average of STUDENT : 4 = 55.400000
Average of STUDENT : 5 = 51.400000
Average of STUDENT : 6 = 47.200000
Average of STUDENT : 7 = 44.400000
Average of STUDENT : 8 = 45.400000
Average of STUDENT : 9 = 64.400000
Average of STUDENT : 10 = 74.400000
Average of STUDENT : 11 = 75.000000
Average of STUDENT : 12 = 61.200000
Average of STUDENT : 13 = 47.000000
Average of STUDENT : 14 = 75.000000
Average of STUDENT : 15 = 52.600000
Average of STUDENT : 16 = 48.200000
Average of STUDENT : 17 = 61.200000
Average of STUDENT : 18 = 58.800000
Average of STUDENT : 19 = 48.600000

DISPLAYING THE NUMBER OF STUDENTS WHO HAVE SCORED BELOW 50 IN THEIR AVERAGE
...
Number of Students who have scored below 50 in their average = 6

DISPLAYING THE SCORES OBTAINED BY EVERY STUDENT ...
Score obtained by STUDENT : 0 = 276
Score obtained by STUDENT : 1 = 311
Score obtained by STUDENT : 2 = 364
Score obtained by STUDENT : 3 = 276
Score obtained by STUDENT : 4 = 277
Score obtained by STUDENT : 5 = 257
Score obtained by STUDENT : 6 = 236
```

```
Score obtained by STUDENT : 7 = 222
Score obtained by STUDENT : 8 = 227
Score obtained by STUDENT : 9 = 322
Score obtained by STUDENT : 10 = 372
Score obtained by STUDENT : 11 = 375
Score obtained by STUDENT : 12 = 306
Score obtained by STUDENT : 13 = 235
Score obtained by STUDENT : 14 = 375
Score obtained by STUDENT : 15 = 263
Score obtained by STUDENT : 16 = 241
Score obtained by STUDENT : 17 = 306
Score obtained by STUDENT : 18 = 294
Score obtained by STUDENT : 19 = 243
********************************************************************************
```

```c
//============( Question : 5 : ARRAYS and STRINGS )====================
#include <stdio.h>
#include <string.h>

void main()
{
    char str1[50];
    char str2[50];
    char str[100];
    int n;

    printf("Enter 2 Strings...\n");
    scanf("%s", str1);
    scanf("%s", str2);
    printf("\nEnter the value of n : ");
    scanf("%d", &n);

    for(int i=0; i<strlen(str1); i++)str[i]=str1[i];
    for(int i=0; i<n; i++)str[i+strlen(str1)]=str2[i];
    str[strlen(str1)+n]='\0';

    printf("String after CONCATENATION:\n%s",str);

}
```

OUTPUT – QUESTION : 5 : STRINGS

```
Enter 2 Strings...

From

IITISM


Enter the value of n : 3

String after CONCATENATION:

From IIT
```

****

```c
//============( Question : 6 : ARRAYS and STRINGS )====================
#include <stdio.h>
#include <string.h>

void main()
{
    char str[100];
    printf("Enter a string...\n");
    gets(str);

    int digits = 0;
    int upper = 0;
    int lower = 0;
    int sp = 0;

    const int ZERO = '0';
    const int SMALL_A = 'a';
    const int CAPS_A = 'A';

    for(int i=0; i<strlen(str); i++)
    {
        char ch = str[i];
        int n = ch;
        if(n>=ZERO && n<ZERO+10)digits++;
        else if(n>=CAPS_A && n<CAPS_A+26)upper++;
        else if(n>=SMALL_A && n<SMALL_A+26)lower++;
        else sp++;
    }

    printf("\nNumber of DIGITS = %d", digits);
    printf("\nNumber of UPPERCASE chars = %d", upper);
    printf("\nNumber of LOWERCASE chars = %d", lower);
    printf("\nNumber of SPECIAL chars = %d", sp);
}
```

OUTPUT – QUESTION : 6 : STRINGS

```
Enter a string...
Mr. John was 18 years old when he came to New York


Number of DIGITS = 2
Number of UPPERCASE chars = 4
Number of LOWERCASE chars = 32
Number of SPECIAL chars = 12
```

****

```c
//============( Question : 7 : ARRAYS and STRINGS )====================
#include <stdio.h>
#include <string.h>

void main()
{
    char str[60];
    char sub[60];
    char newStr[100];
    int index;
    printf("Enter a string...\n");
    gets(str);
    printf("Enter the sub-string...\n");
    gets(sub);
    printf("Enter the index-of-insertion : ");
    scanf("%d", &index);

    int i;
    for(i=0; i<index; i++)newStr[i] = str[i];
    for(int j=0; j<strlen(sub); j++)
    {
        newStr[i] = sub[j];
        i++;
    }
    for(int j=index; j<strlen(str); j++)
    {
        newStr[i] = str[j];
        i++;
    }
    printf("\nThe new Sting is:\n%s", newStr);
}
```

OUTPUT – QUESTION : 7 : STRINGS

```
Enter a string...

Sandeep Auddy

Enter the sub-string...

Kumar

Enter the index-of-insertion : 7


The new Sting is:

SandeepKumar  Auddy
```

****

```c
//============( Question : 8 : ARRAYS and STRINGS )====================
#include <stdio.h>
#include <string.h>
#include <ctype.h>

void main()
{
    char arr1[100];
    char arr2[100];
    int n;
    printf("Enter a String to check...\n");
    gets(arr1);
    printf("Enter the value of n : ");
    scanf("%d", &n);

    for(int i=0; i<n; i++)arr2[i] = toupper(arr1[strlen(arr1)-n+i]);
    arr2[n] = '\0';

    printf("New String :\n%s", arr2);
}
```

OUTPUT – QUESTION : 8 : STRINGS

```
Enter a String to check...

My name is Sandeep Kumar Auddy 755@

Enter the value of n : 10

New String :

AUDDY 755@
```

****

(Lab-5)

ASSIGNMENT: 3

Functions

```c
//=============( QUESTION : 1 : FUNCTIONS )================

#include <stdio.h>

int _prime(int n)
{
    int count = 0;
    for(int i=1; i<=n; i++)
    {
        if(n%i==0)count++;
    }
    return (count==2)?1:0;
}

void main()
{
    int num;
    printf("Enter a number to check : ");
    scanf("%d", &num);
    if(_prime(num)==1)printf("Prime");
    else printf("Not Prime");
}
```

OUTPUT – QUESTION : 1 : FUNCTIONS

```
Enter a number to check : 21

Not Prime
```

********** **********

```c
//=============( QUESTION : 2 : FUNCTIONS )================
#include <stdio.h>

int _leap_year(int year)
{
    int flag;

    if(year%100!=0)
    {
        if(year%4==0) flag = 1;
        else flag = 0;
    }
    else
    {
        if(year%400==0) flag = 1;
        else flag = 0;
    }
    return flag;
}

void main()
{
    int yearNum;
    printf("Enter a year number to check : ");
    scanf("%d", &yearNum);
    if(_leap_year(yearNum)==1)printf("Leap Year");
    else printf("Not a Leap Year");
}
```

OUTPUT – QUESTION : 2 : FUNCTIONS

```
Enter a year number to check : 1800

Not a Leap Year
```

****

```c
//=============( QUESTION : 3 : FUNCTIONS )================
#include <stdio.h>

int getRev(int num)
{
    int rev = 0;
    while(num>0)
    {
        int d = num % 10;
        rev = (rev * 10) + d;
        num/=10;
    }
    return rev;
}

void main()
{
    int num;
    printf("Enter a number : \n");
    scanf("%d", &num);
    printf("\nReverse of %d is: %d", num, getRev(num));
}
```

OUTPUT – QUESTION : 3 : FUNCTIONS

```
Enter a number :
5748


Reverse of 5748 is: 8475
********************************************************************************* ***
```

```c
//=============( QUESTION : 4 : FUNCTIONS )=================
#include <stdio.h>

int manipulate_CallByRef(int* x)
{
    *x = *x + 5;
    printf("\nvalue of x inside manipulate_CallByRef: %d", *x);
}

void main()
{
    int num = 45;
    printf("\nvalue of num inside main (before fxn callByRef): %d", num);
    manipulate_CallByRef(&num);
    printf("\nvalue of num inside main (after fxn callByRef): %d", num);
}
```

OUTPUT – QUESTION : 4 : FUNCTIONS

```
value of num inside main (before fxn callByRef): 45

value of x inside manipulate_CallByRef: 50

value of num inside main (after fxn callByRef): 50
```

****

```c
//=============( QUESTION : 5 : FUNCTIONS )=================
#include <stdio.h>

void swap(int x, int y)
{
    int temp = x;
    x = y;
    y = temp;
    printf("\nValues inside fxn swap: x=%d, y=%d", x, y);
}

void main()
{
    int x = 5;
    int y = 10;
    printf("\nValues before calling fxn swap: x=%d, y=%d", x, y);
    swap(x,y);
    printf("\nValues after calling fxn swap: x=%d, y=%d", x, y);
}
```

OUTPUT – QUESTION : 5 : FUNCTIONS

```
Values before calling fxn swap: x=5, y=10

Values inside fxn swap: x=10, y=5

Values after calling fxn swap: x=5, y=10
```

****

```c
//=============( QUESTION : 6 : FUNCTIONS )================
#include <stdio.h>
int factorial(int n)
{
    if(n==0)return 1;
    int prod = 1;
    for(int i=1; i<=n; i++)prod *= i;
    return prod;
}

int factorial_recursion(int n)
{
    if(n==0 || n==1)return 1;
    return n * factorial_recursion(n-1);
}
void main()
{
    int num1, num2;
    printf("Enter 2 number : \n");
    scanf("%d", &num1);
    scanf("%d", &num2);
    printf("\nFactorial of %d is :%d", num1, factorial(num1));
    printf("\nFactorial of %d is :%d", num2, factorial(num2));
}
```

OUTPUT – QUESTION : 6 : FUNCTIONS

```
Enter 2 number :
6
4


Factorial of 6 is :720
Factorial of 4 is :24
```

****

```c
//=============( QUESTION : 7 : FUNCTIONS )================
#include <stdio.h>
void fibonacci(int n)
{
    int x = 0;
    int y = 1;
    printf("%d, ",x);
    printf("%d, ",y);
    for(int i=3; i<=n; i++)
    {
        int t = x+y;
        x = y;
        y = t;
        printf("%d, ",t);
    }
}
int fibonacci_recursion(int n)
{
    if(n==1)return 0;
    if(n==2)return 1;

    int term = fibonacci_recursion(n-1) + fibonacci_recursion(n-2);
    return term;
}
void main()
{
    int num1, num2;
    printf("Enter the number of terms for 2 series : \n");
    scanf("%d", &num1); scanf("%d", &num2);
    printf("\nSeries having %d terms is :\n", num1);
    fibonacci(num1);
    printf("\nSeries having %d terms is :\n", num2);
    for(int i=1; i<=num2; i++)
        printf("%d, ",fibonacci_recursion(i));
}
```

OUTPUT – QUESTION : 7 : FUNCTIONS

```
Enter the number of terms for 2 series :
7
10


Series having 7 terms is :
0, 1, 1, 2, 3, 5, 8,
Series having 10 terms is :
0, 1, 1, 2, 3, 5, 8, 13, 21, 34,
```

****

```c
//=============( QUESTION : 8 : FUNCTIONS )================
#include <stdio.h>
#include <math.h>

int factorial(int n)
{
    if(n==0)return 1;
    int prod = 1;
    for(int i=1; i<=n; i++)prod *= i;
    return prod;
}
float calculateSine(float x)
{
    const int N = 15;//...Number of Steps

    int sign = 1;
    int index = 1;
    float term = 0;
    for(int i=1; i<=N; i++)
    {
        term += (sign * pow(x,index) / factorial(index));
        sign *= (-1);
        index+=2;
    }
    return term;
}

void main()
{
    float x;
    printf("Enter the value of x : \n");
    scanf("%f", &x);
    float y = calculateSine(x);
    printf("The required value = %f", y);
}
```

OUTPUT – QUESTION : 8 : FUNCTIONS

```
Enter the value of x :

1.57

The required value = 0.999416
```

****

(Lab-6)

ASSIGNMENT: 4

Pointers

```c
//=============( QUESTION : 1 : POINTERS )================
#include <stdio.h>

void main()
{
    char ch;
    char *chp = &ch;
    printf("Enter a character to check :\n");
    scanf(" %c", chp);

    printf("You entered: %c", *chp);
    switch(*chp)
    {
        case 'a': case 'e': case 'i': case 'o': case 'u':
        case 'A': case 'E': case 'I': case 'O': case 'U':
            printf("\nThe entered character is VOWEL");
            break;
        default:
            printf("\nThe entered character is CONSONANT");
    }
}
/*
*/
```

OUTPUT – QUESTION : 1 : USER_DEFINED_DATATYPE

```
Enter a character to check :

U

You entered: U

The entered character is VOWEL

********************************************************************************
```

```c
//=============( QUESTION : 2 : POINTERS )================
#include <stdio.h>

#define A 65
#define Z 90
#define a 97
#define z 122
#define CONVERSION_FACTOR 32

void main()
{

    char greeting[] = "gOOd mORning";
    printf("Greeting (before modification):\n%s", greeting);
    char *index = &greeting[0];
    while(*index != '\0')
    {
        if((*index >= A) && (*index <= Z)) *index += CONVERSION_FACTOR;
        else if((*index >= a) && (*index <= z)) *index -= CONVERSION_FACTOR;
        index++;
    }
    printf("\nGreeting (after modification):\n%s", greeting);
}
/*



*/
```

OUTPUT – QUESTION : 2 : USER_DEFINED_DATATYPE

```
Greeting (before modification):

gOOd mORning

Greeting (after modification):

GooD MorNING
```

****

```c
//=============( QUESTION : 3 : POINTERS )================
#include <stdio.h>

#define N 5

void main()
{

    char text[] = "Oxford University Press";
    char subText[N];

    printf("Original Text:\n%s", text);
    char *index = &text[0];
    while(*index != '\0') index++;

    for(int i=0; i<=N; i++) // Equality used so as to include the NULL charact
er also
    {
        *(subText + i) = *(index - N + i);
    }
    printf("\nSubText (last %d characters):\n%s", N, subText);
}

/*



*/
```

OUTPUT – QUESTION : 3 : USER_DEFINED_DATATYPE

```
Original Text:

Oxford University Press

SubText (last 5 characters):

Press
********************************************************************************
```

```c
//=============( QUESTION : 4 : POINTERS )================
#include <stdio.h>
#include <conio.h>
#include <string.h>

int main()
{
    int arr[50];
    int n;
    printf("Enter the length of the array: ");
    scanf("%d", &n);
    printf("Enter the elements in the array: \n");
    for(int i=0; i<n; i++) scanf("%d", arr+i);

    int insertIndex;
    int insertElement;
    printf("Enter the INDEX OF INSERTION : ");
    scanf("%d", &insertIndex);
    if(insertIndex<0 || insertIndex>n)
    {
        printf("INVALID INDEX...");
        return -1;
    }
    printf("Enter the ELEMENT TO BE INSERTED : ");
    scanf("%d", &insertElement);

    for(int i=n-1; i>=insertIndex; i--) *(arr + i + 1) = *(arr + i);
    *(arr + insertIndex) = insertElement;
    n++;

    printf("ARRAY after INSERTION of: %d at INDEX: %d is...\n", insertElement,
 insertIndex);
    for(int i=0; i<n; i++)printf("%d ",*(arr+i));

    return 0;
}
```

OUTPUT – QUESTION : 4 : USER_DEFINED_DATATYPE

```
Enter the length of the array: 4

Enter the elements in the array:

5 7 -4 0

Enter the INDEX OF INSERTION : 2

Enter the ELEMENT TO BE INSERTED : 89

ARRAY after INSERTION of: 89 at INDEX: 2 is...

5 7 89 -4 0
```

****

```c
//=============( QUESTION : 5 : POINTERS )================
#include <stdio.h>
#include <conio.h>
#include <string.h>

void main()
{

    char text[100];
    char revText[100];

    printf("Enter a String:\n");
    gets(text);

    int n = strlen(text);
    char *index1 = text + n - 1;
    char *index2 = revText;

    for(int i=0; i<n; i++)
    {
        *index2 = *index1;
        index1--; index2++;
    }
    *index2 = '\0';
    printf("Reversed String:\n%s", revText);
}

/*



*/
```

OUTPUT – QUESTION : 5 : USER_DEFINED_DATATYPE

```
Enter a String:

Sandeep Kumar Auddy

Reversed String:

ydduA ramuK peednaS

********************************************************************************
```

```c
//=============( QUESTION : 6 : POINTERS )================
#include <stdio.h>
#include <conio.h>
#include <string.h>

void main()
{
    int arr1[100], arr2[100];
    int n1, n2;

    // INPUT ...
    printf("Enter no. of elements in 1st-ARRAY : ");
    scanf("%d", &n1);
    for(int i=0; i<n1; i++) scanf("%d", (arr1+i));
    printf("Enter no. of elements in 2nd-ARRAY : ");
    scanf("%d", &n2);
    for(int i=0; i<n2; i++) scanf("%d", (arr2+i));

    // COMPARISION ...
    int n = (n1>=n2) ? n2 : n1 ;
    int flag = 0;
    int i;
    for(i=0; i<n; i++)
    {
        flag = *(arr1 + i) - *(arr2 + i);
        if(flag != 0) break;
    }
    if((flag == 0) && (n1 != n2)) flag = (n1>n2) ? *(arr1+i) : *(arr2+i) ;

    printf("COMPARISON VALUE = %d", flag);
}

/*


*/
```

OUTPUT – QUESTION : 6 : USER_DEFINED_DATATYPE

```
Enter no. of elements in 1st-ARRAY : 3

6 -6 9

Enter no. of elements in 2nd-ARRAY : 5

6 -6 9 -20 500

COMPARISON VALUE = -20
```

****

```c
//=============( QUESTION : 7 : POINTERS )================
#include <stdio.h>

char* deleteOccurence(char* string, char repChar)
{
    char newString[50];

    char *index1 = &string[0];
    char *index2 = &newString[0];

    while(*index1 != '\0')
    {
        char ch = *index1;
        if(ch!=repChar)
        {
            *index2 = *index1;
            index2++;
        }
        index1++;
    }
    *index2 = '\0';
    //printf("\n%s", newString);

    index1 = &string[0];
    index2 = &newString[0];
    while(*index2 != '\0')
    {
        *index1 = *index2;
        index1++;
        index2++;
    }
    *index1 = '\0';
    return &string[0];
}

void main()
{
    char string[50];
    char repChar;

    printf("Enter a String : \n");
    //scanf("%s", string);
    gets(string);
    printf("Enter the character to delete : ");
    scanf(" %c", &repChar);

    char *newString = deleteOccurence(string, repChar);
```

```
    printf("String after deletion :\n%s",newString);
}
/*
*/
```

OUTPUT – QUESTION : 7 : USER_DEFINED_DATATYPE

```
Enter a String :

My name is Sandeep Kumar Auddy

Enter the character to delete : a

String after deletion :

My nme is Sndeep Kumr Auddy
```

****

```c
//=============( QUESTION : 8 : POINTERS )================
#include <stdio.h>
#include <conio.h>
#include <string.h>

void main()
{
    int m, n;
    int A[50][50], B[50][50];
    int sum[50][50];

    // INPUT ...
    printf("Enter no. of rows and columns : ");
    scanf("%d %d", &m, &n);
    printf("Enter the values in 1st MATRIX\n");
    for(int i=0; i<m; i++)
    {
        for(int j=0; j<n; j++)
            scanf("%d", (*(A+i)+j));
    }
    printf("Enter the values in 2nd MATRIX\n");
    for(int i=0; i<m; i++)
    {
        for(int j=0; j<n; j++)
            scanf("%d", (*(B+i)+j));
        printf("\n");
    }
    printf("1st-MATRIX\n");
    for(int i=0; i<m; i++)
    {
        for(int j=0; j<n; j++)
            printf("%d\t", *(*(A+i)+j));
        printf("\n");
    }
    printf("2nd-MATRIX\n");
    for(int i=0; i<m; i++)
    {
        for(int j=0; j<n; j++)
            printf("%d\t", *(*(B+i)+j));
        printf("\n");
    }
    // ADDITION ...
    for(int i=0; i<m; i++)
    {
        for(int j=0; j<n; j++)
            *(*(sum+i)+j) = *(*(A+i)+j) + *(*(B+i)+j);
        printf("\n");
    }
```

```
    printf("\nSummation Matrix:\n");
    for(int i=0; i<m; i++)
    {
        for(int j=0; j<n; j++)
            printf("%d\t", *(*(sum+i)+j));
        printf("\n");
    }
}
```

OUTPUT – QUESTION : 8 : USER_DEFINED_DATATYPE

```
Enter no. of rows and columns : 2 2

Enter the values in 1st MATRIX

3 4 5 6

Enter the values in 2nd MATRIX

3 -4 -6 10


1st-MATRIX

3       4

5       6

2nd-MATRIX

3       -4

-6      10


Summation Matrix:

6       0

-1      16
```

****

(Lab-7)

ASSIGNMENT: 5

User-defined Datatypes

```c
//=============( QUESTION : 1 : USER_DEFINED_DATATYPE )================
#include<stdio.h>
struct emp
{
    char name[30];
    int age;
    float salary;
};
int main()
{
    struct emp e;
    printf("Enter the name of the employee:\n");
    scanf("%s",e.name);
    fflush(stdin);
    printf("Enter age of the employee:\n");
    scanf("%d",&e.age);
    printf("Enter salary of the employee:\n");
    scanf("%f",&e.salary);
    printf("The details you entered are:\n");
    printf("Employee name: %s\n",e.name);
    printf("Age: %d\n",e.age);
    printf("Salary: %.2f\n",e.salary);
    return 0;
}
```

OUTPUT – QUESTION : 1 : USER_DEFINED_DATATYPE

```
Enter the name of the employee:

Sandeep

Enter age of the employee:

19

Enter salary of the employee:

80000

The details you entered are:

Employee name: Sandeep

Age: 19

Salary: 80000.00
```

****

```c
//=============( QUESTION : 2 : USER_DEFINED_DATATYPE )================
#include<stdio.h>
struct dept
{
    char name[25];int age;float salary;char contact[25];
};
int main()
{
    struct dept b[25];
    int n,i;
    printf("Enter the number of faculty in your department:\n");
    scanf("%d",&n);
    for(i=0;i<n;i++)
    {
        printf("Employee%d\n",i+1);
        printf("Enter name:\n");
        scanf("%s",b[i].name);
        printf("Enter age:\n");
        scanf("%d",&b[i].age);
        printf("Enter salary:\n");
        scanf("%f",&b[i].salary);
        printf("Enter contact:\n");
        scanf("%s",&b[i].contact);
    }
    printf("Entered details are:\n");
    for(i=0;i<n;i++)
    {
        printf("Employee%d:\n",i+1);
        printf("Name:%s Age:%d Salary:%.2f contact:%s\n",b[i].name,b[i].age,b[
i].salary,b[i].contact);
    }
    return 0;
}
```

OUTPUT – QUESTION : 2 : USER_DEFINED_DATATYPE

```
Enter the number of faculty in your department:
3
Employee1
Enter name:
Sandeep
Enter age:
19
Enter salary:
5000
Enter contact:
8340393937
Employee2
Enter name:
Zeeshan
Enter age:
22
Enter salary:
3000
Enter contact:
9878012345
Employee3
Enter name:
Nitish
Enter age:
24
Enter salary:
3500
Enter contact:
9430709788
Entered details are:
Employee1:
Name:Sandeep Age:19 Salary:5000.00 contact:8340393937
Employee2:
Name:Zeeshan Age:22 Salary:3000.00 contact:9878012345
Employee3:
Name:Nitish Age:24 Salary:3500.00 contact:9430709788
```

****

```c
//=============( QUESTION : 3 : USER_DEFINED_DATATYPE )================
#include <stdio.h>
#include <string.h>
#define N 5

typedef struct
{
    char name[50];
    char address[100];
    char grade;
    int numRooms;
    int charge;
} Hotel;

Hotel hotels[N] = {
    {
        "Skylark",
        "Bank-More",
        'A',
        590,
        4000
    },
    {
        "Seventeen degree",
        "Sriram Mall Dhansar",
        'B',
        505,
        3000
    },
    {
        "Cocoon",
        "Forest Colony",
        'A',
        300,
        1200
    },
    {
        "Aroti Resturant",
        "Hirapur Dhanbad",
        'B',
        50,
        600
    },
    {
        "Prabhat Hotel",
        "Bank More",
```

```
        'C',
        60,
        200
    }
};


void displayData(Hotel hotel)
{
    printf("\nNAME: %s", hotel.name);
    printf("\nADDRESS: %s", hotel.address);
    printf("\nGRADE: %c", hotel.grade);
    printf("\nNUMBER_Of_ROOMS: %d", hotel.numRooms);
    printf("\nCHARGE_per_DAY (in Rs.): %d\n", hotel.charge);
}


void main()
{
    char gd;
    printf("Enter the desired GRADE of the HOTEL : ");
    scanf("%c", &gd);
    for(int i=0; i<N; i++)
    {
        if(hotels[i].grade == gd) displayData(hotels[i]);
    }

    int p;
    printf("\nEnter the desired CUTOFF-PRICE of the HOTEL : ");
    scanf("%d", &p);
    for(int i=0; i<N; i++)
    {
        if(hotels[i].charge < p) displayData(hotels[i]);
    }
}
```

OUTPUT – QUESTION : 3 : USER_DEFINED_DATATYPE

```
Enter the desired GRADE of the HOTEL : B

NAME: Seventeen degree
ADDRESS: Sriram Mall Dhansar
GRADE: B
NUMBER_Of_ROOMS: 505
CHARGE_per_DAY (in Rs.): 3000

NAME: Aroti Resturant
ADDRESS: Hirapur Dhanbad
GRADE: B
NUMBER_Of_ROOMS: 50
CHARGE_per_DAY (in Rs.): 600

Enter the desired CUTOFF-PRICE of the HOTEL : 1000

NAME: Aroti Resturant
ADDRESS: Hirapur Dhanbad
GRADE: B
NUMBER_Of_ROOMS: 50
CHARGE_per_DAY (in Rs.): 600

NAME: Prabhat Hotel
ADDRESS: Bank More
GRADE: C
NUMBER_Of_ROOMS: 60
CHARGE_per_DAY (in Rs.): 200
********************************************************************************
```

```c
//=============( QUESTION : 4 : USER_DEFINED_DATATYPE )================
#include <stdio.h>
#include <string.h>

struct S
{
    int i;
    char ch;
    double d;
};
union U
{
    int i;
    char ch;
    double d;
};

int main()
{
    printf("\nSize of the structure is %d", sizeof(struct S));
    printf("\nSize of the union is %d", sizeof(union U));
    return 0;
}
```

OUTPUT – QUESTION : 4 : USER_DEFINED_DATATYPE

```
Size of the structure is 16

Size of the union is 8
```

****

```c
//=============( QUESTION : 5 : USER_DEFINED_DATATYPE )================
#include <stdio.h>

typedef struct {int hr; int min; int sec;} Time;

int compareTimes(Time t1, Time t2)
{
    if(t1.hr == t2.hr)
    {
        if(t1.min == t2.min)
        {
            if(t1.sec == t2.sec)return 0;
            else return (t1.sec - t2.sec);
        }
        else return (t1.min - t2.min);
    }
    else return (t1.hr - t2.hr);
}

void main()
{
    Time start_time, end_time;
    printf("Enter Start Time in (hr, min, sec) : ");
    scanf("%d %d %d", &start_time.hr, &start_time.min, &start_time.sec);
    printf("Enter End Time in (hr, min, sec) : ");
    scanf("%d %d %d", &end_time.hr, &end_time.min, &end_time.sec);

    printf("\nNow continue to enter your Start_Time (End_Time is Fixed)...\n")
;
    while(1)
    {
        scanf("%d %d %d", &start_time.hr, &start_time.min, &start_time.sec);
        if(compareTimes(start_time, end_time)>=0)break;
        printf("GOOD DAY\n");
    }
}
```

OUTPUT – QUESTION : 5 : USER_DEFINED_DATATYPE

```
Enter Start Time in (hr, min, sec) : 12 50 45
Enter End Time in (hr, min, sec) : 21 34 56

Now continue to enter your Start_Time (End_Time is Fixed)...
12 58 45
GOOD DAY
16 34 45
GOOD DAY
19 34 54
GOOD DAY
21 31 22
GOOD DAY
21 34 56
********************************************************************************
```

```c
//=============( QUESTION : 6 : USER_DEFINED_DATATYPE )================
#include <stdio.h>

typedef struct
{
    float x;
    float y;
} POINT;

int determineQuadrant(POINT point)
{
    if(point.x==0 || point.y==0)
    {
        printf("Quadrant cannot be determined !!");
        return -1;
    }
    int boolX = (point.x > 0) ? 1 : 0;
    int boolY = (point.y > 0) ? 1 : 0;

    if(boolX && boolY)printf("1st QUADRANT...!!");
    else if(!boolX && boolY)printf("2nd QUADRANT...!!");
    else if(!boolX && !boolY)printf("3rd QUADRANT...!!");
    else printf("4th QUADRANT...!!");

    return 0;
}
void main()
{
    POINT p;
    printf("Enter the coordinates of a Point to check:\n");
    scanf("%f %f", &p.x, &p.y);
    printf("The given point belongs to :\n");
    determineQuadrant(p);
}
```

OUTPUT – QUESTION : 6 : USER_DEFINED_DATATYPE

```
Enter the coordinates of a Point to check:
56 -2
The given point belongs to :
4th QUADRANT...!!
```

****

```c
//=============( QUESTION : 7 : USER_DEFINED_DATATYPE )=================
#include <stdio.h>
#include <math.h>
#define PIE 3.14

struct Circle
{
    float radius;
};
struct Rectangle
{
    float length;
    float breadth;
};
struct Triangle
{
    float a;
    float b;
    float c;
};

union Shape
{
    struct Circle circle;
    struct Rectangle rectangle;
    struct Triangle triangle;
};

float calcArea_Circle(struct Circle C)
{
    return (PIE * pow(C.radius, 2));
}
float calcArea_Rectangle(struct Rectangle R)
{
    return (R.length * R.breadth);
}
float calcArea_Triangle(struct Triangle T)
{
    float sp = (T.a + T.b + T.c)/2;
    float spA = sp - T.a;
    float spB = sp - T.b;
    float spC = sp - T.c;
    return (sqrt(sp * spA * spB * spC));
}

int main()
{
```

```
    int choice;
    union Shape ipshape;
    printf("Enter a Shape ( 1:Circle, 2:Rectangle, 3:Traingle ) :\n");
    scanf("%d", &choice);

    float area;
    switch(choice)
    {
        case 1:
            printf("Enter the radius: ");
            scanf("%f", &ipshape.circle.radius);
            area = calcArea_Circle(ipshape.circle); break;
        case 2:
            printf("Enter the length and breadth: ");
            scanf("%f %f", &ipshape.rectangle.length, &ipshape.rectangle.bread
th);
            area = calcArea_Rectangle(ipshape.rectangle); break;
        case 3:
            printf("Enter the 3 sides: ");
            scanf("%f %f %f", &ipshape.triangle.a, &ipshape.triangle.b, &ipsha
pe.triangle.c);
            area = calcArea_Triangle(ipshape.triangle); break;

        default:
            printf("Invalid choice !");
            return -1;
    }
    printf("Area of the SHAPE = %f", area);
    return 0;
}
```

OUTPUT – QUESTION : 7 : USER_DEFINED_DATATYPE

```
Enter a Shape ( 1:Circle, 2:Rectangle, 3:Traingle ) :
2
Enter the length and breadth: 34.5 660
34.500000Area of the SHAPE = 22770.000000
```

****

```c
//=============( QUESTION : 8 : USER_DEFINED_DATATYPE )================
#include <stdio.h>
#define N 10

struct Employee
{
    int emp_Id;
    struct Name
    {
        char firstName[50];
        char middleName[50];
        char lastName[50];
    } name;
    struct Address
    {
        char area[50];
        char city[50];
        char state[50];
    } address;
    int age;
    int salary;
    char designation[60];
};

void main()
{
    struct Employee employee[N];
    printf("Enter the Details of the employee...");

    for(int i=0; i<N; i++)
    {
        printf("\nEMPLOYEE no. : %d ...\n", i+1);
        printf("Employee_ID : ");
        scanf("%d", &(employee[i].emp_Id));
        printf("Employee_NAME (First, Middle, Last) : ");
        scanf("%s %s %s", &(employee[i].name.firstName), &(employee[i].name.mi
ddleName), &(employee[i].name.lastName));
        printf("Employee_ADDRESS (Area, City, Jharkhand) : ");
        scanf("%s %s %s", &(employee[i].address.area), &(employee[i].address.c
ity), &(employee[i].address.state));
        printf("Employee_AGE : ");
        scanf("%d", &(employee[i].age));
        printf("Employee_SALARY : ");
        scanf("%d", &(employee[i].salary));
        printf("Employee_DESIGNATION : ");
        scanf("%s", employee[i].designation);
        printf("\n");
    }
```

```
    //printf("Hello ..");
    //printf("\n%s", employee[1].address.city);

    printf("\n------------------ The EMPLOYEES are ...\n");
    for(int i=0; i<N; i++)
    {
        printf("\nEMPLOYEE no. : %d ...", i+1);
        printf("\nEmployee_ID : %d", employee[i].emp_Id);
        //scanf("%d", &(employee[i].emp_Id));
        printf("\nEmployee_NAME :  %s %s %s", employee[i].name.firstName, empl
oyee[i].name.middleName, employee[i].name.lastName);
        //scanf("%s %s %s", &(employee[i].name.firstName), &(employee[i].name.
middleName), &(employee[i].name.lastName));
        printf("\nEmployee_ADDRESS : %s %s %s",
employee[i].address.area, employee[i].address.city, employee[i].address.state)
;
        //scanf("%s %s %s", &(employee[i].address.area), &(employee[i].address
.city), &(employee[i].address.state));
        printf("\nEmployee_AGE : %d", employee[i].age);
        //scanf("%d", &(employee[i].age));
        printf("\nEmployee_SALARY : %d", employee[i].salary);
        //scanf("%d", &(employee[i].salary));
        printf("\nEmployee_DESIGNATION : %s", employee[i].designation);
        //scanf("%s", employee[i].designation);
        printf("\n");
    }
}
```

OUTPUT – QUESTION : 8 : USER_DEFINED_DATATYPE

```
Enter the Details of the employee...

EMPLOYEE no. : 1 ...

Employee_ID : 190741

Employee_NAME (First, Middle, Last) : Sandeep Kumar Auddy

Employee_ADDRESS (Area, City, Jharkhand) : Manaitand Dhanbad Jharkhand

Employee_AGE : 19

Employee_SALARY : 34000

Employee_DESIGNATION : CEO



EMPLOYEE no. : 2 ...

Employee_ID : 190742

Employee_NAME (First, Middle, Last) : Sanal Kumar Roy

Employee_ADDRESS (Area, City, Jharkhand) : BankMore Dhanbad Jharkhand
```

Employee_AGE : 19

Employee_SALARY : 50000

Employee_DESIGNATION : Manager

EMPLOYEE no. : 3 ...

Employee_ID : 190743

Employee_NAME (First, Middle, Last) : Sayantan Kumar Das

Employee_ADDRESS (Area, City, Jharkhand) : Tamluk Kolkata WestBengal

Employee_AGE : 19

Employee_SALARY : 34000

Employee_DESIGNATION : Manager

EMPLOYEE no. : 4 ...

Employee_ID : 190744

Employee_NAME (First, Middle, Last) : Mandar Sanjay Deokar

Employee_ADDRESS (Area, City, Jharkhand) : Naodaya Aurangabad Maharashtra

Employee_AGE : 19

Employee_SALARY : Manager

Employee_DESIGNATION :

EMPLOYEE no. : 5 ...

Employee_ID : 190745

Employee_NAME (First, Middle, Last) : Amarsh Kumar Jain

Employee_ADDRESS (Area, City, Jharkhand) : Dolba Neemuch MP

Employee_AGE : 19

Employee_SALARY : Programmer

Employee_DESIGNATION :

EMPLOYEE no. : 6 ...

Employee_ID : 190746

Employee_NAME (First, Middle, Last) : Narendra Damodar Modi

Employee_ADDRESS (Area, City, Jharkhand) : Manaitand Dhanbad Jharkhand

Employee_AGE : 65

Employee_SALARY : 500000

Employee_DESIGNATION : PM

EMPLOYEE no. : 7 ...

Employee_ID : 190747

Employee_NAME (First, Middle, Last) : Kumar Aryan Karn

Employee_ADDRESS (Area, City, Jharkhand) : Manaitand Patna Bihar

Employee_AGE : 19

Employee_SALARY : 60000

Employee_DESIGNATION : Programmer

EMPLOYEE no. : 8 ...

Employee_ID : 190748

Employee_NAME (First, Middle, Last) : Patnayak Kumar Jha

Employee_ADDRESS (Area, City, Jharkhand) : Manaitand Dhanbad Jharkhand

Employee_AGE : 20

Employee_SALARY : 20000

Employee_DESIGNATION : Sweeper

EMPLOYEE no. : 9 ...

Employee_ID : 190749

Employee_NAME (First, Middle, Last) : Pankaj Kumar Barnwal

Employee_ADDRESS (Area, City, Jharkhand) : Hirapur Dhanbad Jharkhand

Employee_AGE : 78

Employee_SALARY : 76000

Employee_DESIGNATION : Teacher

EMPLOYEE no. : 10 ...

Employee_ID : 190750

Employee_NAME (First, Middle, Last) : Bambam Shivam Barnwal

Employee_ADDRESS (Area, City, Jharkhand) : Bartand Dhanbad Jharkhand

Employee_AGE : 23

Employee_SALARY : 23000

Employee_DESIGNATION : Sweeper

------------------ The EMPLOYEES are ...

EMPLOYEE no. : 1 ...

Employee_ID : 190741

Employee_NAME (First, Middle, Last) : Sandeep Kumar Auddy

Employee_ADDRESS (Area, City, Jharkhand) : Manaitand Dhanbad Jharkhand

Employee_AGE : 19

Employee_SALARY : 34000

Employee_DESIGNATION : CEO

EMPLOYEE no. : 2 ...

Employee_ID : 190742

Employee_NAME (First, Middle, Last) : Sanal Kumar Roy

Employee_ADDRESS (Area, City, Jharkhand) : BankMore Dhanbad Jharkhand

Employee_AGE : 19

Employee_SALARY : 50000

Employee_DESIGNATION : Manager

EMPLOYEE no. : 3 ...

Employee_ID : 190743

Employee_NAME (First, Middle, Last) : Sayantan Kumar Das

Employee_ADDRESS (Area, City, Jharkhand) : Tamluk Kolkata WestBengal

Employee_AGE : 19

Employee_SALARY : 34000

Employee_DESIGNATION : Manager

EMPLOYEE no. : 4 ...

Employee_ID : 190744

Employee_NAME (First, Middle, Last) : Mandar Sanjay Deokar

Employee_ADDRESS (Area, City, Jharkhand) : Naodaya Aurangabad Maharashtra

Employee_AGE : 19

Employee_SALARY : Manager

Employee_DESIGNATION :

EMPLOYEE no. : 5 ...

Employee_ID : 190745

Employee_NAME (First, Middle, Last) : Amarsh Kumar Jain

Employee_ADDRESS (Area, City, Jharkhand) : Dolba Neemuch MP

Employee_AGE : 19

Employee_SALARY : Programmer

Employee_DESIGNATION :

EMPLOYEE no. : 6 ...

Employee_ID : 190746

Employee_NAME (First, Middle, Last) : Narendra Damodar Modi

Employee_ADDRESS (Area, City, Jharkhand) : Manaitand Dhanbad Jharkhand

Employee_AGE : 65

Employee_SALARY : 500000

Employee_DESIGNATION : PM

EMPLOYEE no. : 7 ...

Employee_ID : 190747

Employee_NAME (First, Middle, Last) : Kumar Aryan Karn

Employee_ADDRESS (Area, City, Jharkhand) : Manaitand Patna Bihar

Employee_AGE : 19

Employee_SALARY : 60000

Employee_DESIGNATION : Programmer

EMPLOYEE no. : 8 ...

Employee_ID : 190748

Employee_NAME (First, Middle, Last) : Patnayak Kumar Jha

Employee_ADDRESS (Area, City, Jharkhand) : Manaitand Dhanbad Jharkhand

Employee_AGE : 20

Employee_SALARY : 20000

Employee_DESIGNATION : Sweeper

EMPLOYEE no. : 9 ...

Employee_ID : 190749

Employee_NAME (First, Middle, Last) : Pankaj Kumar Barnwal

Employee_ADDRESS (Area, City, Jharkhand) : Hirapur Dhanbad Jharkhand

Employee_AGE : 78

Employee_SALARY : 76000

Employee_DESIGNATION : Teacher

EMPLOYEE no. : 10 ...

Employee_ID : 190750

Employee_NAME (First, Middle, Last) : Bambam Shivam Barnwal

Employee_ADDRESS (Area, City, Jharkhand) : Bartand Dhanbad Jharkhand

Employee_AGE : 23

Employee_SALARY : 23000

Employee_DESIGNATION : Sweeper

****

(Lab-8)

ASSIGNMENT: 6

File Management

```c
//==========( QUESTION : 1 : FILE_MANAGEMENT )==========
#include <stdio.h>
#include <string.h>

void main()
{
    FILE* fp;
    fp = fopen("names_q4.txt", "r");

    char names[100][100];
    int index = -1;
    while(1)
    {
        char name[100];
        char *ch = fgets(name, 150, fp);
        if(ch==NULL)break;

        index++;
        strcpy(names[index], name);
    }

    printf("Names as read from the File are as follows:\n");

    for(int i=index; i>=0; i--)printf("\n%s", names[i]);
}
```

OUTPUT – QUESTION : 1 : FILE_MANAGEMENT

```
Names as read from the File are as follows:


Albert Einstein

Robert Frost

Jhon Brown

Bambam Shivam

Akash Agarwal

Sayantan Das

Sanal Roy

Sandeep Kumar Auddy
**********
```

```c
//=============( QUESTION : 2 : FILE_MANAGEMENT )================
#include <stdio.h>

/*
The Program is designed to read our
employee.dat file as used in Question-1
(since in the question it has not been mentioned which binary file to read)
It is Assumed that the Program knows beforehand,
what type of data structure the employee.dat contains

The Program is going to read all the records and then display it on the consol
e...
*/

typedef struct
{
    char name[100];
    int age;
    int basicSalary;
} Employee;

FILE *fp;

int main()
{
    fp = fopen("employee.dat", "rb");
    if(fp==NULL)
    {
        printf("\nUnable to Open the FILE employee.dat");
        return -1;
    }
    while(1)
    {
        Employee emp;
        fread(&emp,sizeof(emp),1,fp);
        if(feof(fp))
        break;

        printf("\nEMPLOYEE_NAME : %s", emp.name);
        printf("\nEMPLOYEE_AGE : %d", emp.age);
        printf("\nEMPLOYEE_BASIC_SALARY : %d\n", emp.basicSalary);
    }
}
```

OUTPUT – QUESTION : 2 : FILE_MANAGEMENT

```
EMPLOYEE_NAME : Sandeep
EMPLOYEE_AGE : 19
EMPLOYEE_BASIC_SALARY : 45000

EMPLOYEE_NAME : Sanal
EMPLOYEE_AGE : 20
EMPLOYEE_BASIC_SALARY : 34000

EMPLOYEE_NAME : Sayantan
EMPLOYEE_AGE : 19
EMPLOYEE_BASIC_SALARY : 44000

EMPLOYEE_NAME : Mandar
EMPLOYEE_AGE : 18
EMPLOYEE_BASIC_SALARY : 23000
********************************************************************** *****
```

```c
//=============( QUESTION : 3 : FILE_MANAGEMENT )=================
#include <stdio.h>
/*
The program counts the NUMBER_OF_CHARACTERS:
in a text file called mssgToDir_q6.txt
which contains a request letter to the Directo of IIT-ISM
not to conduct ONLINE-EXAMS
*/

void main()
{
    FILE *fp = fopen("mssgToDir_q6.txt", "r");
    int count = 0;
    int numLines = 0;
    int numTabs = 0;
    int numSpaces = 0;
     while(1)
    {
        char ch = getc(fp);
        if(ch==EOF)break;
        if(ch=='\n')numLines++;
        else if(ch=='\t')numTabs++;
        else if(ch==' ')numSpaces++;
        else count++;
    }
    printf("In the file: mssgToDir_q6.txt ...\n", count);

    printf("\nNumber of CHARACTERS (excluding spaces,tabs,linfeeds) = %d", cou
nt);
    printf("\nNumber of SPACES = %d", numSpaces);
    printf("\nNumber of TABS = %d", numTabs);
    printf("\nNumber of LINEFEEDS = %d", numLines);
}
```

OUTPUT – QUESTION : 3 : FILE_MANAGEMENT

```
In the file: mssgToDir_q6.txt ...


Number of CHARACTERS (excluding spaces,tabs,linfeeds) = 303

Number of SPACES = 57

Number of TABS = 1

Number of LINEFEEDS = 11

********************************************************************************
```

```c
//=============( QUESTION : 4 : FILE_MANAGEMENT )=================
#include <stdio.h>
#include <string.h>


FILE *fp;
char fname[20];

int createAndWriteFile()
{
    printf("Enter the name of the File to create: ");
    scanf("%s", fname);

    fp = fopen(fname, "w+");
    if(fp==NULL)
    {
        printf("ERROR in OPENING the FILE : %s", fname);
        return -1;
    }
    printf("\nEnter the names of your favourite authors...(terminated by 0000)
\n");
    while(1)
    {
        char nameAuth[32];
        scanf("%s", nameAuth);
        if(strcmp(nameAuth, "0000")==0)break;

        fprintf(fp, "%s\n", nameAuth);
    }
    fclose(fp);
    printf("Writing process ended successfully...");
    return 0;
}
int readFile()
{
    printf("\nPresenting before you your fav Authors :\n");
    fp = fopen(fname, "r");
    if(fp==NULL)
    {
        printf("ERROR in OPENING the FILE : %s", fname);
        return -1;
    }
    while(1)
    {
        char word[60];
        char *ch;
        ch = fgets(word, 60,fp);
        if(ch==NULL)break;
        printf("%s\n", word);
```

```
    }

    fclose(fp);
    printf("Reading process ended successfully...");
    return 0;
}

int main()
{
    createAndWriteFile();
    readFile();
}
```

OUTPUT – QUESTION : 4 : FILE_MANAGEMENT

```
Enter the name of the File to create: literature_q9.txt


Enter the names of your favourite authors...(terminated by 0000)
WilliamShakespeare
RobertFrost
Tulsidas
WilliamWordsworth
CarolNDuffy
0000
Writing process ended successfully...
Presenting before you your fav Authors :
WilliamShakespeare


RobertFrost


Tulsidas


WilliamWordsworth


CarolNDuffy


Reading process ended successfully...
**********************************************************************
```

```c
//=============( QUESTION : 5 : FILE_MANAGEMENT )=================
#include <stdio.h>

typedef struct
{
    int x;
    int y;
    int z;
} Point;

FILE *tfp;
FILE *bfp;

int txtToBin()
{
    tfp = fopen("coordinatesT_q10.txt", "r");
    bfp = fopen("coordinatesB_q10.dat", "wb");
    if(tfp==NULL)
    {
        printf("ERROR in OPENING the FILE : coordinatesT_q10.txt");
        return -1;
    }
    if(bfp==NULL)
    {
        printf("ERROR in OPENING the FILE : coordinatesB_q10.dat");
        return -2;
    }
    printf("\nTransferring contents from .txt to .dat ...\n");
    while(1)
    {
        Point point;
        if(feof(tfp))break;
        fscanf(tfp, "%d %d %d", &point.x, &point.y, &point.z);

        //if(feof(tfp))break;
        //printf("( %d, %d, %d )", point.x, point.y, point.z);
        fwrite(&point,sizeof(point),1,bfp);
    }

    fclose(tfp);
    fclose(bfp);
    printf("\n...Successfully Transferred contents from .txt to .dat\n");
    return 0;
}
int readBinFile()
{
    bfp = fopen("coordinatesB_q10.dat", "rb");
    if(bfp==NULL)
```

```
    {
        printf("ERROR in OPENING the FILE : coordinatesB_q10.dat");
        return -3;
    }
    printf("\nReading contents of .dat ...\n");
    printf("COORDINATES OF SOME POINTS :\n");
    while(1)
    {
        Point point;
        fread(&point,sizeof(point),1,bfp);
        if(feof(bfp))break;
        printf("( %d, %d, %d )\n", point.x, point.y, point.z);
    }

    printf("\n...Successfully Read all contents from .dat\n");
    fclose(bfp);
    return 0;
}
void main()
{
    txtToBin();
    readBinFile();
}
```

OUTPUT – QUESTION : 5 : FILE_MANAGEMENT

```
Transferring contents from .txt to .dat ...


...Successfully Transferred contents from .txt to .dat


Reading contents of .dat ...
COORDINATES OF SOME POINTS :
( 12, 34, 56 )
( -23, 56, 78 )
( 89, -7, 100 )
( 22, 8, -6 )
( 2, 3, 4 )
( 5, 6, 33 )
( -6, -7, -7 )
( 5, -6, -7 )
( 78, 9, -100 )
( 45, 6, -8 )
```

...Successfully Read all contents from .dat

****

```c
//=============( QUESTION : 6 : FILE_MANAGEMENT )=================
#include<stdio.h>
#include<ctype.h>
#include<stdlib.h>
int main()
{
    FILE *fp,*ftemp;
    char ch,c;
    fp=fopen("case.txt","r");
    if(fp==NULL)
        exit(EXIT_FAILURE);
    ftemp=fopen("case_t.txt","w");
    if(ftemp==NULL)
        exit(EXIT_FAILURE);
    while(1)
    {
        ch=fgetc(fp);
        if(ch==EOF)break;
        ch=toupper(ch);
        fputc(ch,ftemp);
    }
    fclose(fp);fclose(ftemp);
    ftemp=fopen("case_t.txt","r");
    fp=fopen("case.txt","w");
    if(fp==NULL)exit(EXIT_FAILURE);
    else
    {
        while(1)
        {
            ch=fgetc(ftemp);
            if(ch==EOF)break;
            fputc(ch,fp);
        }
        fclose(fp);
        fclose(ftemp);
        printf("The lowercase alphabets have been successfully changed to uppe
rcase\n");
        remove("case_t.txt");
    }
    return 0;
}
```

OUTPUT – QUESTION : 6 : FILE_MANAGEMENT

```
The lowercase alphabets have been successfully changed to uppercase
```

****

```c
//=============( QUESTION : 7 : FILE_MANAGEMENT )=================
#include<stdio.h>
#include<stdlib.h>
int main()
{
    int arr[100],n,i,sum=0;
    FILE *fp;
    fp=fopen("NUM.txt","w");
    printf("Enter the number of integers you want to enter:\n");
    scanf("%d",&n);
    for(i=0;i<n;i++)
    {
        printf("Enter an integer:\n");
        scanf("%d",&arr[i]);
    }
    for(i=0;i<n;i++)sum=sum+arr[i];
    for(i=0;i<n;i++)
    fprintf(fp,"%d ",arr[i]);
    fclose(fp);
    fp=fopen("NUM.txt","a+");
    if(fp==NULL)
    {
        printf("ERROR");
        exit(EXIT_FAILURE);
    }
    else
    {
        fprintf(fp,"%d",sum);
        printf("The sum of the integers are appended successfully\n");
    }
    return 0;
}
```

OUTPUT – QUESTION : 7 : FILE_MANAGEMENT

```
Enter the number of integers you want to enter:
5
Enter an integer:
1
Enter an integer:
2
Enter an integer:
3
Enter an integer:
4
Enter an integer:
```

```
5
The sum of the integers are appended successfully
**********************************************************************
```

```c
//=============( QUESTION : 8 : FILE_MANAGEMENT )================
#include <stdio.h>

/*
The Program is designed to read our
employee.dat file as used in Question-1
(since in the question it has not been mentioned which binary file to read)
It is Assumed that the Program knows beforehand,
what type of data structure the employee.dat contains

The Program is going to read the number of records and then display it on the
console...
*/

typedef struct
{
    char name[100];
    int age;
    int basicSalary;
} Employee;

FILE *fp;

int main()
{
    fp = fopen("employee.dat", "rb");
    if(fp==NULL)
    {
        printf("\nUnable to Open the FILE employee.dat");
        return -1;
    }
    printf("Reading the Number of RECORDS from the FILE: employee.dat...\n");
    int count = 0;
    while(1)
    {
        Employee emp;
        fread(&emp,sizeof(emp),1,fp);
        if(feof(fp))
        break;
```

```
        count++;
    }
    printf("The number of Records = %d", count);
}
```

OUTPUT – QUESTION : 8 : FILE_MANAGEMENT

```
Reading the Number of RECORDS from the FILE: employee.dat...

The number of Records = 4

********************************************************************************
```