Computer Engineering Department, S.V.N.I.T. Surat.

B Tech (CO) –IInd Year semester-III

Course: *Data Structures CO203*

**Assignment-III**

1.) In a classroom of 100 students you have to divide students in 10 sub list of equal students.

Perform the following operation.

1. Insertion of marks.

2. Deletion of marks.

3. Updating of marks.

4. Display marks

5. Replace one sub list with other

6. Compare two sub list

7. Enter nth element in jth list

Code:

*// In a classroom of 100 students you have to divide students*

*//  in 10 sub list of equal students.*

*#include* <stdio.h>

int student[101], i, j; *// global student array*

*// 1. Insertation of marks.*

void fx1()

{

    int cnt = 0;

    printf("Enter the Number of Students whose Data is to be Inserted :\n");

    scanf("%d", &cnt);

*for* (int i = 0; i < cnt; i++)

    {

        int rno;

        printf("Enter the Roll Number of Student :\n");

        scanf("%d", &rno);

*if* (rno <= 0 && rno >= 101)

        {

            printf("Enter Valid Roll Number!\n");

        }

*else*

        {

            int mark = 0;

            printf("Enter the Marks of Student :\n");

            scanf("%d", &mark);

            student[rno] = mark;

        }

    }

*return*;

}

*// 2. Deletion of marks.*

void fx2()

{

    int cnt = 0;

    printf("Enter the Number of Students whose Data is to be Deleted :\n");

    scanf("%d", &cnt);

*for* (int i = 0; i < cnt; i++)

    {

        int rno;

        printf("Enter the Roll Number of Student :\n");

        scanf("%d", &rno);

*if* (rno <= 0 && rno >= 101)

        {

            printf("Enter Valid Roll Number!\n");

        }

*else*

        {

            student[rno] = -1;

        }

    }

*return*;

}

*// 3. Updation of marks.*

void fx3()

{

    int cnt = 0;

    printf("Enter the Number of Students whose Data is to be Updated :\n");

    scanf("%d", &cnt);

*for* (int i = 0; i < cnt; i++)

    {

        int rno;

        printf("Enter the Roll Number of Student :\n");

        scanf("%d", &rno);

*if* (rno <= 0 && rno >= 101)

        {

            printf("Enter Valid Roll Number!\n");

        }

*else*

        {

            int mark = 0;

            printf("Enter the Updated Marks of Student :\n");

            scanf("%d", &mark);

            student[rno] = mark;

        }

    }

*return*;

}

*// 4. Display marks*

void fx4()

{

    int cnt = 0;

    printf("Enter the Number of Students whose Data is to be Displayed :\n");

    scanf("%d", &cnt);

*for* (int i = 0; i < cnt; i++)

    {

        int rno;

        printf("Enter the Roll Number of Student :\n");

        scanf("%d", &rno);

*if* (rno <= 0 && rno >= 101)

        {

            printf("Enter Valid Roll Number!\n");

        }

*else*

        {

*if* (student[rno] != -1)

                printf("The Marks of Student with Roll No %d is : %d\n", rno, student[rno]);

*else*

            {

                printf("The Marks of Student Not Entered by Admin!!\n");

            }

        }

    }

*return*;

}

*// 5. Replace one sub list with other*

void fx5()

{

    int r1;

    printf("Enter Sublist to be Replaced :\n");

    scanf("%d", &r1);

    printf("Enter 10 Students Marks for this SubList\n");

*for* (int i = (r1 - 1) \* 10 + 1; i <= r1 \* 10; i++)

    {

        scanf("%d", &student[i]);

    }

*return*;

}

*// 6. Compare two sub list*

void fx6()

{

    int r1, r2;

    printf("Enter Two Sublist to be Compared :\n");

    scanf("%d %d", &r1, &r2);

    int id1 = (r1 - 1) \* 10 + 1;

    int ed1 = r1 \* 10;

    int id2 = (r2 - 1) \* 10 + 1;

    int ed2 = r2 \* 10;

*while* (id1 <= ed1 && id2 <= ed2)

    {

*if* (student[id1] != student[id2])

        {

            printf("Both Sublist are Not Equal\n");

*return*;

        }

        id1++;

        id2++;

    }

    printf("Both Sublist are Equal\n");

*return*;

}

*// 7. Enter nth element in jth list*

void fx7()

{

    int n, j, mark;

    printf("Enter the jth List Number :\n");

    scanf("%d", &j);

    printf("Enter the nth Element Number :\n");

    scanf("%d", &n);

    printf("Enter the Mark of Student :\n");

    scanf("%d", &mark);

    int idx = (j - 1) \* 10 + n;

*if* (idx >= 0 && idx <= 100)

    {

        student[idx] = mark;

        printf("Marks Entered Successfully!\n");

    }

*else*

    {

        printf("Enter Valid jth List or nth Element!\n");

    }

*return*;

}

int main()

{

*// Marks of Student Not Initialized = -1*

*for* (int i = 0; i <= 100; i++)

    {

        student[i] = -1;

    }

    printf("Perform the following operation.\n");

    printf("1 -> Insertation of marks.\n");

    printf("2 -> Deletion of marks.\n");

    printf("3 -> Updation of marks.\n");

    printf("4 -> Display marks\n");

    printf("5 -> Replace one sub list with other\n");

    printf("6 -> Compare two sub list\n");

    printf("7 -> Enter nth element in jth list\n");

    char ch = 'Y';

*while* (ch == 'Y' || ch == 'y')

    {

        int choice = 0;

        printf("Enter your Choice : ");

        scanf("%d", &choice);

*switch* (choice)

        {

*case* 1:

            fx1();

*break*;

*case* 2:

            fx2();

*break*;

*case* 3:

            fx3();

*break*;

*case* 4:

            fx4();

*break*;

*case* 5:

            fx5();

*break*;

*case* 6:

            fx6();

*break*;

*case* 7:

            fx7();

*break*;

*default*:

            printf("Enter Valid Choice [1-7] Only!");

*break*;

        }

        fflush(stdin); *// for character input*

        printf("Want to Do Another Operation?(Y/N)\n");

        scanf("%c", &ch);

    }

*return* 0;

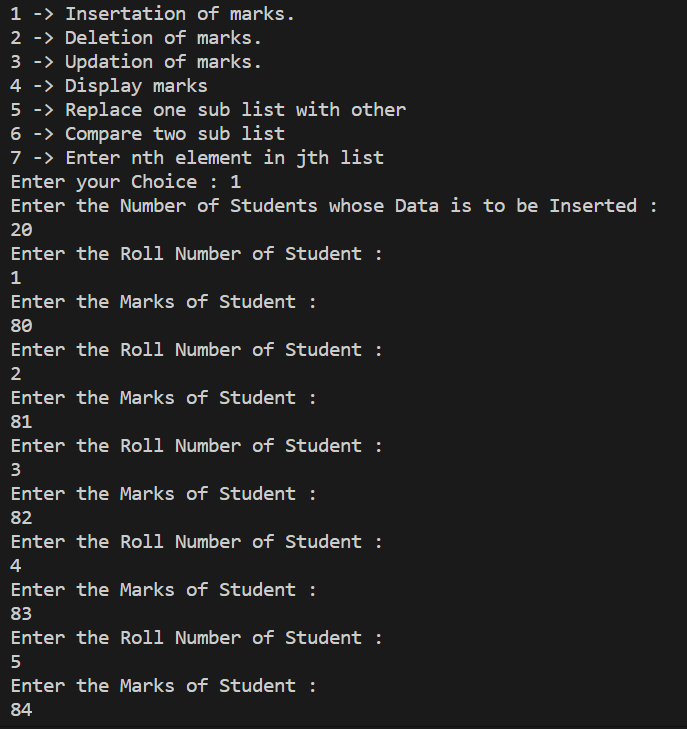
}

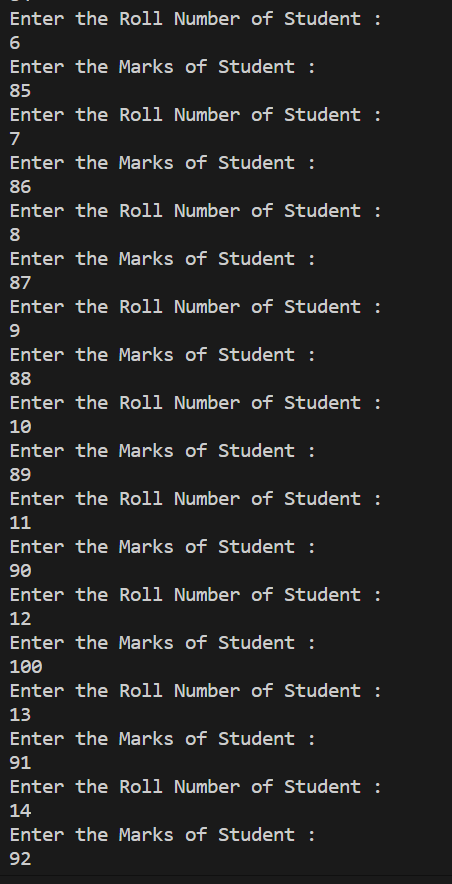
OUTPUT ON TEST DATA:

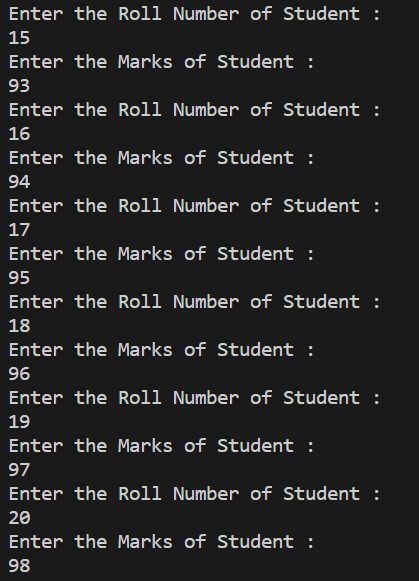
Step 1: Insert Mark of 20 Students:

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| List 1 | 80 | 81 | 82 | 83 | 84 | 85 | 86 | 87 | 88 | 89 |
| List 2 | 90 | **100** | 91 | 92 | 93 | 94 | 95 | 96 | 97 | 98 |

(Actually Roll Number 12 is Me!! Therefore 100!)



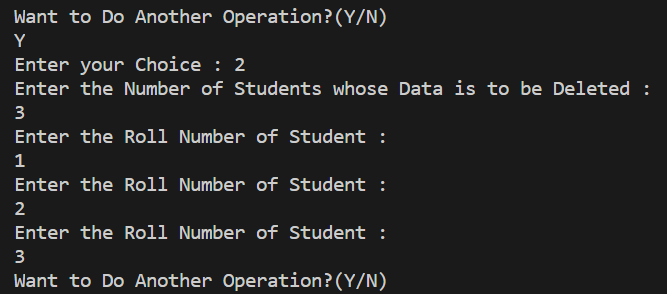




Step 2: Delete the Marks of Student with Roll Numbers 1, 2, 3;

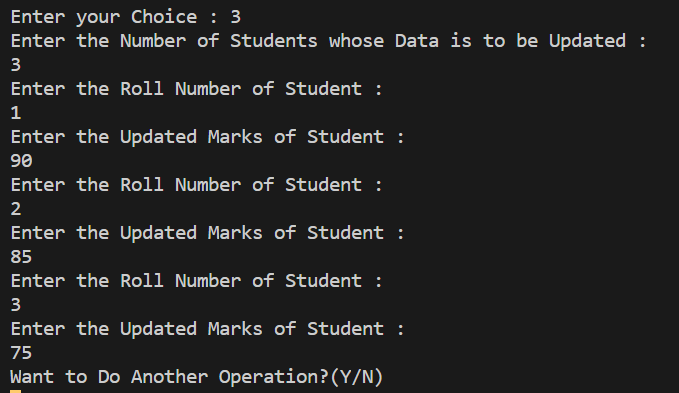
|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| List 1 | **-1** | **-1** | **-1** | 83 | 84 | 85 | 86 | 87 | 88 | 89 |
| List 2 | 90 | 100 | 91 | 92 | 93 | 94 | 95 | 96 | 97 | 98 |

(-1 denotes that Admin has not inserted the Student Marks)



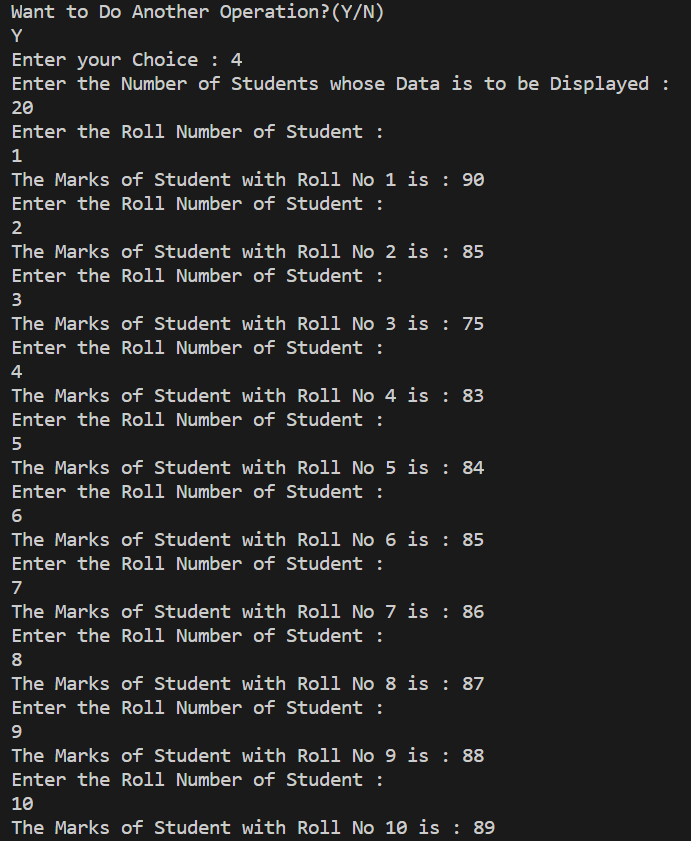
Step 3: Update the Marks of Student with Roll Number 1, 2, 3.

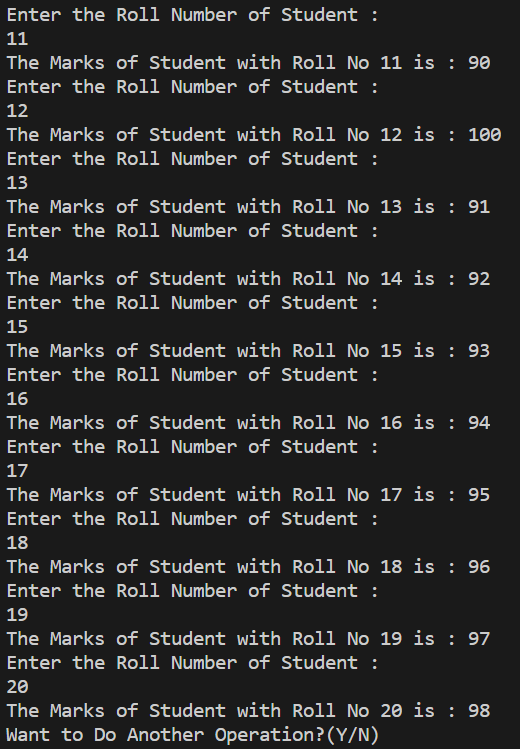
|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| List 1 | **90** | **85** | **75** | 83 | 84 | 85 | 86 | 87 | 88 | 89 |
| List 2 | 90 | 100 | 91 | 92 | 93 | 94 | 95 | 96 | 97 | 98 |



Step 4: Display of Student Marks from Roll Number 1 to 20

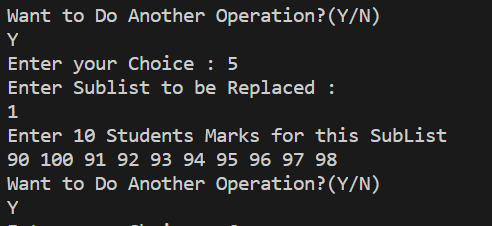
|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| List 1 | **90** | **85** | **75** | 83 | 84 | 85 | 86 | 87 | 88 | 89 |
| List 2 | 90 | 100 | 91 | 92 | 93 | 94 | 95 | 96 | 97 | 98 |



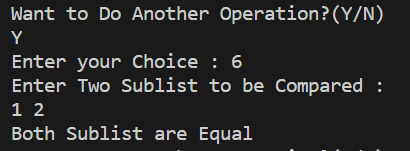


Step 5: Replace the Sub-list 1 with Values of Sub-list 2

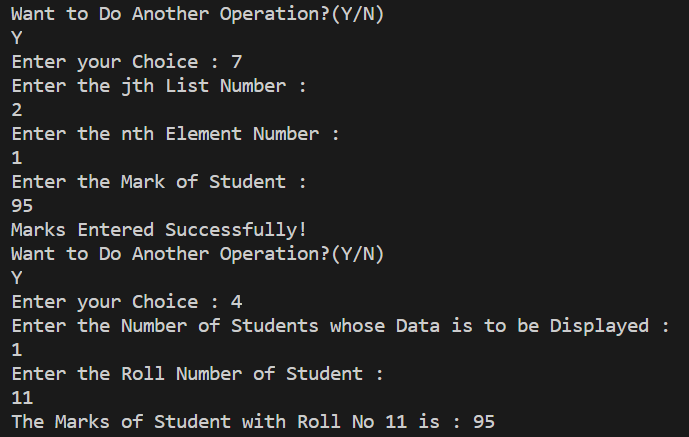
|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| List 1 | **90** | **100** | **91** | **92** | **93** | **94** | **95** | **96** | **97** | **98** |
| List 2 | 90 | 100 | 91 | 92 | 93 | 94 | 95 | 96 | 97 | 98 |



Step 6: Comparing Two Sub-Lists 1 & 2 (which we made Equal!)



Step 7: 2nd List 1st Element = 11 Roll Number whose marks I changed to 95.



2.) Write a program

(1) To generate sub matrices from the multidimensional matrix.

(2) Take any two (I took all Sub-Matrices) sub matrices and perform the addition of two sub matrices.

For example input Array is A [4] [4] and B [2] [2]

(3) Find the occurrence of B in A.

The rows or columns in A are not required to be consecutive.

Code:

*// Write a program (1) to generate sub matrices from the multidimensional matrix.*

*// (2) Take any two(I took all) sub matrices and perform the addition of two sub matrices.*

*// For example input Array is A[4][4] and B[2][2]*

*// (3) Find the occurrence of B in A.*

*#include* <stdio.h>

int mat[101][101], B[101][101], add[101][101];

void print(int rst, int cst, int red, int ced)

{

    printf("Sub-Matrix [%d ... %d] [%d ... %d] :\n", rst, red, cst, ced);

    int addrow = 0;

    int addcol = 0;

*for* (int i = rst; i <= red; i++)

    {

*for* (int j = cst; j <= ced; j++)

        {

            printf("%d ", mat[i][j]);

            add[addrow][addcol] += mat[i][j];

            addcol++;

        }

        printf("\n");

        addrow++;

        addcol = 0;

    }

*return*;

}

int occurances(int rst, int cst, int red, int ced, int B[101][101])

{

    int ans = 0;

    int flag = 1;

*// printf("Sub-Matrix [%d ... %d] [%d ... %d] is being Compared :\n", rst, red, cst, ced);*

    int brow = 0;

    int bcol = 0;

*for* (int i = rst; i <= red; i++)

    {

*for* (int j = cst; j <= ced; j++)

        {

*if* (mat[i][j] != B[brow][bcol])

            {

                flag = 0;

*break*;

            }

*else*

            {

                bcol++;

            }

        }

        brow++;

        bcol = 0;

*if* (flag == 0)

        {

*break*;

        }

    }

*if* (flag)

        ans = 1;

*return* ans;

}

int main()

{

    int m, n;

    printf("Enter the Dimensions of Matrix (mXm) (<=100) : \n");

    scanf("%d", &m); *// m rows x m columns*

    printf("Enter the Values of (mXm) Matrix :\n");

*for* (int i = 0; i < m; i++)

    {

*for* (int j = 0; j < m; j++)

        {

            printf("mat[%d][%d] = ", i, j);

            scanf("%d", &mat[i][j]);

        }

    }

    printf("\n");

    printf("Task 1 --> Find All Sub-Matrices \n\n");

    printf("Enter the Dimensions of Sub-Matrix (nXn) (<=%d) : \n", m);

    scanf("%d", &n); *// n rows x n columns*

*if* (n > m)

    {

        printf("Enter Valid n value!!\n");

    }

*else*

    {

*// printf("%dX%d Parent Matrixes is :\n", m, m);*

*// print(0,0,m-1,m-1);*

        printf("%dX%d Sub Matrixes are :\n", n, n);

        int rstart = 0, rend = n - 1;

        int cstart = 0, cend = n - 1;

*for* (; rend <= m - 1; rend++)

        {

            rstart = rend - n + 1;

            cend = n - 1;

*for* (; cend <= m - 1; cend++)

            {

                cstart = cend - n + 1;

                print(rstart, cstart, rend, cend);

            }

        }

    }

    printf("Task 2 --> Sum of all Subsets \n\n");

    printf("Add Matrix(%d X %d)\n", n, n);

*for* (int i = 0; i < n; i++)

    {

*for* (int j = 0; j < n; j++)

        {

            printf("%d ", add[i][j]);

        }

        printf("\n");

    }

    printf("\n");

    printf("Task 3 --> To Find the Occurance of B in A\n\n");

    printf("Enter Values of Sub-Matrix B (%dX%d) (%d <= %d):\n", n, n, n, m);

*for* (int i = 0; i < n; i++)

    {

*for* (int j = 0; j < n; j++)

        {

            printf("B[%d][%d] = ", i, j);

            scanf("%d", &B[i][j]);

        }

    }

    printf("\n");

    int occur = 0; *// count the Number of Occurances*

    int rstart = 0, rend = n - 1;

    int cstart = 0, cend = n - 1;

*for* (; rend <= m - 1; rend++)

    {

        rstart = rend - n + 1;

        cend = n - 1;

*for* (; cend <= m - 1; cend++)

        {

            cstart = cend - n + 1;

            occur += occurances(rstart, cstart, rend, cend, B);

        }

    }

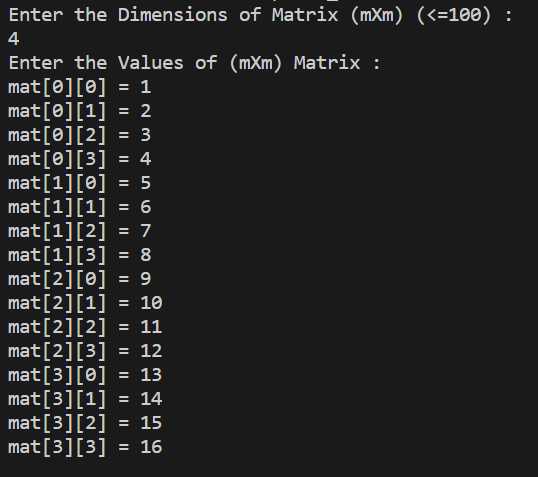
    printf("The Occurance of B in A is : %d", occur);

*return* 0;

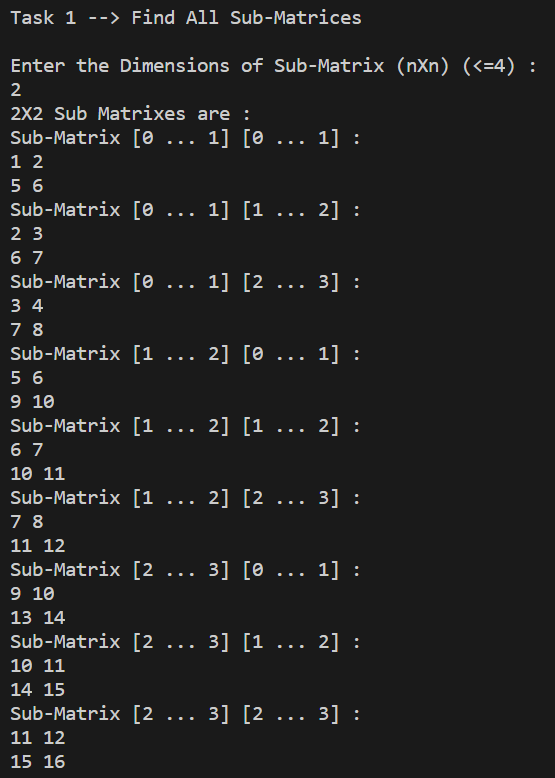
}

TEST CASE:

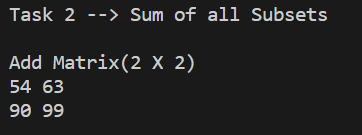
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Row\Column** | **0** | **1** | **2** | **3** |
| **0** | 1 | 2 | 3 | 4 |
| **1** | 5 | 6 | 7 | 8 |
| **2** | 9 | 10 | 11 | 12 |
| **3** | 13 | 14 | 15 | 16 |



TASK 1: Find all Sub-Matrices of Size 2

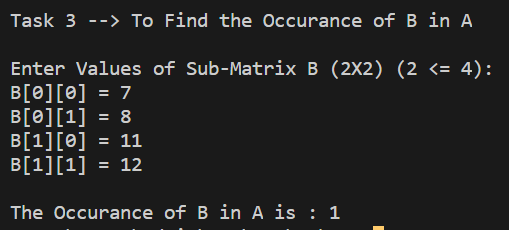


TASK 2: Sum of all the (2X2) Sub-Matrices



TASK 3: Find Occurrences of (2X2) Matrix in Parent Matrix

I took one of the Subset from parent Matrix so Answer is 1.



Submitted By:

Roll Number: **U19CS012** (*D-12*)

Name: *Bhagya Rana*