

Chapter-7
Programming Exercise

7.1) Write a program for fitting a straight line through n points $(x_i, y_i), i=1, 2, 3, \dots$. The straight line equation is:

$$Y = mx + c$$

and the values of m and c are given by:

$$m = (\frac{n}{n} \sum (x_i, y_i)) - (\sum x_i)(\sum y_i) / n^2 (\sum x_i^2) - (\sum x_i)^2$$

All summations are from 1 to n .

```
#include <stdio.h>
void main()
{
    int i, n=10, v1, v2, x[10], y[10];
    float float total_x, total_y, total_xy, total_x2;
    float m, c, temp, temp1;
    printf ("Enter the values for x:");
    for(i=0; i<10; i++)
    {
        scanf ("%d", &v1);
        x[i] = v1;
    }
    printf ("Enter the values for y:");
    for(i=0; i<10; i++)
    {
        scanf ("%d", &v2);
        y[i] = v2;
    }
    total_x = 0;
    total_y = 0;
    total_xy = 0;
    total_x2 = 0;
```

```
for (i=0; i<10; i++)
```

```
{
```

```
    scanf ("%f.%f,%f",
```

```
        &x[i], &y[i],
```

```
}
```

```
total_x = total_y = total_xy = total_x2 = 0;
```

```
for (i=0; i<10; i++)
```

```
{
```

```
    total_x = total_x + x[i];
```

```
    total_y = total_y + y[i];
```

```
    total_xy = total_xy + (x[i] * y[i]);
```

```
    total_x2 = total_x2 + (x[i] * x[i]);
```

```
}
```

```
temp = total_x * total_y;
```

```
temp1 = total_x * total_x;
```

```
m = ((n * total_xy) - (temp)) / ((n * total_x2) - temp1);
```

```
c = ((total_y) - (m * total_x)) / n;
```

```
printf ("In the equation of the straight  
line is: ");
```

```
printf ("Y = %.fx + %.f",
```

```
getch();
```

```
}
```

Output:

Enter the values for x :

1 2 3 4 5 6 7 8 9 10

Enter the values for y :

12 3 4 5 6 7 8 9 10

The equation

of the straight line is:

$$Y = 1.00000X + 0.00000$$

The daily maximum temperature recorded in 10 cities of January during the month have been tabulated as follows.

City

Day

1

2

3

;

3)

Write a program to read the table

into a two-dimensional array

and to find the city and day corresponding to.

(a) the highest temperature.

(b) the lowest temperature.

Program:

```
#include <stdio.h>
Void main()
{
    int temp[2][2];
    int i, j, City1, City2, MaxTemp, MinTemp;
    printf ("Enter temperature :- \n");
    for (i=0; i<2; i++)
    {
        printf (" for City %d->\n", i);
        for (j=0; j<2; j++)
        {
            printf ("for Day %d->", j);
            scanf ("%d", &temp[i][j]);
        }
    }
    printf (" Temperature Matrix :- \n");
    printf (" City In      ");
    for (i=0; i<2; i++)
        printf (" %d ", i+1);
    printf ("\n Day In      ");
    for (i=0; i<2; i++)
        printf (" %d ", i+1);
```

```

    printf ("%.2f ", i+1);
    for (j=0; j<2; j++)
    {
        printf ("%.2f", Temp[i][j]);
    }
    printf ("\n");
}

```

```

MinTemp = MaxTemp = Temp[0][0];
City1 = 0;
City2 = 0;
for (i=0; i<2; i++)
{
    for (j=0; j<2; j++)
    {
        if (MaxTemp < Temp[i][j])
        {
            MaxTemp = Temp[i][j];
            City1 = j + 1;
        }
        if (MinTemp > Temp[i][j])
        {
            MinTemp = Temp[i][j];
            City2 = j + 1;
        }
    }
}

```

```
printf ("In Highest Temperature of City %d is
```

```
    %d\n", City1, MaxTemp);
```

```
printf ("Lowest Temperature of City %d is %d\n",
```

```
    City2, MinTemp);
```

```
getch();
```

```
}
```

7.3)

An election is contested by 5 candidates.

The candidate are numbered one 1 to 5 and the voting is done by marking the candidate number on the ballot paper. Write a program to read the ballots and count the votes cast for each candidate using an array variable count. In case a number, read is outside the range 1 to 5, the ballot should be considered as a 'spoilt ballot'

and the program should also count the number of spoilt ballots.

Program:

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

```
void main()
```

```
{
```

```
int i, vote[5], c1=0, c2=0, c3=0, c4=0,  
c5=0, Count=0, Count_SP=0, v;
```

```
printf ("Enter your votes for 5 Candidates");
```

```
for (i=1; i<=5; i++)
```

```
{
```

```
scanf ("%d", &v);
```

```
vote[i]=v;
```

```
}
```

```
for (i=1; i<=5; i++)
```

```
{
```

```
if (vote[i]==1)
```

```
c1=c1+1;
```

```
else
```

```
{
```

```
if (vote[i]==2)
```

```
c2=c2+1;
```

```
else
```

```
{
```

```
if (vote[i]==3)
```

```
c3=c3+1;
```

```
else
```

```
{
```

```
if (vote[i]==4)  
c4=c4+1;
```

else

if (vote[i] == 5)

c5 = c5 + 1;

}

}

}

printf ("\\n votes to candidate 1 = %d", c1);

printf ("\\n votes to candidate 2 = %d", c2);

printf ("\\n votes to candidate 3 = %d", c3);

printf ("\\n votes to candidate 4 = %d", c4);

printf ("\\n votes to candidate 5 = %d", c5);

for (i=1; i<=5; i++)

{

if (vote[i] == 5)

count = count + 1;

else

count - SP = count - SP + 1;

}

printf ("The number of valid votes is: %d", count);

printf ("\\n The number of spoilt votes is: %d", count - SP);

getch()

{

Enter

4
3

1
8

2

Your votes for 5 candidates:

Votes to Candidate 1: 2

Votes to Candidate 2: 1

Votes to Candidate 3: 1

Votes to Candidate 4: 0

Votes to Candidate 5: 0

The number

The number of valid votes is: 4

The annual examination is: 1.

10 students results of

are tabulated as follows.

Roll No.	Subject 1	Subject 2	Subject 3
1	85	78	80
2	76	72	70
3	88	80	82
4	74	70	72
5	80	75	78
6	72	68	70
7	82	78	80
8	70	65	68
9	84	76	78
10	76	70	72

74)

Write a program to read the data and determine the following:

- Total marks obtained by each student.
- The highest marks in each subject and the roll no. of the student who secured it.
- The student who obtained the highest total marks.

Program:

```
#include <stdio.h>
Void main()
{
    int i, roll, m1, m2, m3, sub[Max], sub2[Max], sub3[Max];
    int total, sub1, total-sub2, total-sub3, total1[Max];
    int max, max1, max2, max3, roll1, roll2, roll3;

    printf ("Enter the marks for subject of all
            the students:");
    for(i=0; i<MAX; i++)
        scanf ("%d", &sub[i]);
    printf ("Enter the marks for subject 2
            of all the students:");
    for (i=0; i<MAX; i++)
        scanf ("%d", &sub2[i]);
```

printf ("Enter the marks for subject 2
of all the students");

for (i=0; i<MAX; i++)

scanf ("%d", &sub2[i]);

printf ("Enter the marks for subject 3
of all the students");

for (i=0; i<MAX; i++)

scanf ("%d", &sub3[i]);

total - sub1 = total - sub2 = total - sub3 = 0;

for (i=0; i<MAX; i++)

total - sub1 = total - sub1 + sub1[i];

total - sub2 = total - sub2 + sub2[i];

total - sub3 = total - sub3 + sub3[i];

~~total - sub~~

total[i] = sub1[i] + sub2[i] + sub3[i];

}

for (i=0; i<MAX; i++)

{

printf ("The total marks obtained

by the student %d is %d\n", i+1, total[i]);

```
max1 = sub1[0];
max2 = sub2[0];
max3 = sub3[0];
max = total[0];
roll1 = 0;
roll2 = 0;
roll3 = 0;
roll = 0;
for (i=0; i<MAX; it++)
{
    if (max1 < sub1[i]))
    {
        max1 = sub1[i];
        roll1 = i+1;
    }
    if (max2 < sub2[i])
    {
        max2 = sub2[i];
        roll2 = it+1;
    }
    if (max3 < sub3[i])
    {
        max3 = sub3[i];
        roll3 = it+1;
    }
}
```

```
if (max < total[i])  
{  
    max = total[i];  
    roll = i;  
}  
}  
}
```

printf ("The highest marks in subject 1 is a
y.d and the roll number is y.d", max, roll);

printf ("The highest marks in subject 2 is y.d
and the roll number is y.d", max2, roll2);

printf ("The highest marks in subject 3 is y.d
and the roll number is y.d", max3, roll3);

printf ("The highest total marks is y.d
and the roll number is y.d", maxroll);
getch();

}

7.5) Given are one dimensional arrays A
and B which are sorted in ascending
order. Write a program to merge
them into a single sorted array C
that contains every item from array

A and B, in ascending order
Program:

```
#include <stdio.h>
#define MAX 50
void main()
{
    int a[MAX], b[MAX], c[MAX];
    int ar, br, cr, n, m, mn;
    ar = br = cr = 0;
    printf("Enter no. of elements of array: ");
    scanf("%d %d", &n, &m);
    printf("Enter elements of first array: ");
    for (i=0; i<n; i++)
        scanf("%d", &a[i]);
    printf("Enter elements of second array: ");
    for (i=0; i<m; i++)
        scanf("%d", &b[i]);
    mn = mntr;
    while (ar < n && br < m)
    {
        if (a[ar] < b[br])
            c[cr] = a[ar];
        else
            c[cr] = b[br];
        ar++;
        br++;
        cr++;
    }
    for (i=0; i<cr; i++)
        printf("%d ", c[i]);
```

```
{ if (a[an] < b[bn])
```

```
{     c[cx] = a[an];  
     ax++;
```

```
}
```

```
else
```

```
{
```

```
    c[cx] = b[bn];  
    bx++;
```

```
}
```

```
cx++;
```

```
}
```

```
if (ax == n)
```

```
{
```

```
    while (bx < m)
```

```
{
```

```
    c[cn] = b[bn];
```

```
    bx++;
```

```
    cn++;
```

```
}
```

```
else
```

```
{
```

```
    while (cn < n)
```

```

    {
        c[n] = a[n];
        a++ ;
        c++ ;
    }
}

printf ("the sorted array is :\n");
for (i=0; i<n; i++)
    printf ("%d", c[i]);
getch();
}

```

7.6) write a program that will read the values of elements of A and B and produce the product matrix C.

Program:

```

#include <stdio.h>
#define MAX 10;
void main ()
{
    int a[MAX][MAX], b[MAX][MAX], c[MAX][MAX];

```

```

int i, j, k, row, col;
printf ("Enter row of matrix");
scanf ("%d", &row);
printf ("Enter column of matrix");
scanf ("%d", &col);
printf ("Enter first matrix[n]");
for(i=0; i<row; i++)
    for(j=0; j<col; j++)
        scanf ("%d", &a[i][j]);
printf ("\n Enter second matrix[n]");
for(i=0; i<row; i++)
    for(j=0; j<col; j++)
        scanf ("%d", &b[i][j]);
printf ("The first matrix is : \n");
for (j=0; j<col; j++)
    printf ("%d", a[i][j]);
}
printf ("\n");

```

```

printf ("In Second matrix is\n");
for (i=0; i<row; i++)
{
    for (j=0; j<col; j++)
        printf ("%d", b[i][j]);
    printf ("\n");
}
for (i=0; i<row; i++)
{
    for (j=0; j<col; j++)
    {
        c[i][j]=0;
        for (k=0; k<col; k++)
            c[i][j] = c[i][j] + a[i][k] * b[k];
    }
}
printf ("Multiplication is\n");
for (i=0; i<row; i++)
{
    for (j=0; j<col; j++)
        printf ("%d", c[i][j]);
    printf ("\n");
}
getch();

```

7.8

Write a program that fills a five by five as follows:

- (1) Upper left triangle with +1s
 - (2) Lower right triangle with -1s
 - (3) Right diagonal with zeros.
- Display the contents of the matrix using not more than two printf statements.
- Program:

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

```
void main()
```

```
{
```

```
    int A[5][5];
```

```
    int a, i, k, j;
```

```
    a = 3;
```

```
    for (i = 0; i <= 3; i++)
```

```
{
```

```
        for (j = 0; j <= a; j++)
```

```
{
```

```
            A[i][j] = +1;
```

```
}
```

```
a = -j;
```

```
}
```

```
j=4;  
for (i=0; i<=4; i++)
```

```
{  
    A[i][j] = 0;  
    j--;  
}
```

```
a=4;  
for (i=1; i<=4; i++)
```

```
{  
    for (j=4; j>=a; j--):
```

```
{  
    A[i][j] = 1;  
}  
a--;
```

```
{  
printf ("Array is:-\n\n");
```

```
for (i=0; i<=4; i++)
```

```
{  
    for (j=0; j<=4; j++)
```

```
        printf ("%d", A[i][j]);
```

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

```
}  
getch();
```

7.8) write a program to implement Selection Sort Program.

```
#include <stdio.h>
Void main()
{
    int A[10];
    int i, k, Small, Loc;
    printf ("Enter elements of Array, --(n)");
    for (i=0; i<=9; i++)
        scanf ("%d", &A[i]);
    for (k=0; k<=9; k++)
    {
        Small = A[k];
        Loc = k;
        for (i = k+1; i<=9; i++)
            if (Small > A[i])
                Small = A[i];
        Loc = i;
        A[Loc] = A[k];
    }
}
```

```
A[K] = Small;
```

```
{
```

```
printf (" Sorted Array Is :-\n");
```

```
for(i=0; i<=9; i++)
```

```
    printf ("%d", A[i]);
```

```
getch();
```

```
}
```

7.9: Write a program to implement Binary Search Algorithm.

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

```
void main()
```

```
{
```

```
int str[10];
```

```
int i, Beg, End, Mid, Item;
```

```
Beg = 0;
```

```
End = 9;
```

```
Mid = (Beg + End) / 2;
```

```
printf ("Enter Any Sorted Array:-\n");
```

```
for(i=0; i<10; i++)
```

```

scanf ("%s", &Str[i]);
printf ("Enter Item Which u want to Search:-");
scanf ("%s", &Item);
while ((Item != Str[mid]) && (Beg <= End))
{
    if (Item < Str[mid])
        End = Mid - 1;
    else
        Beg = Mid + 1;
    Mid = (Beg + End) / 2;
}
if (Beg > End)
    printf ("Item Not found\n");
else
    printf ("%s found At Index %d\n", Item, Mid);
getch();

```

7.10) Write a program that will compute the length of a given character string.

```

#include <stdio.h>
void main()

```

```

2
char Str[50];
int i, Len;
Len = 0;
printf ("Enter a string : -\n");
scanf ("%[^ \n]s", &Str);
for (i = 0; Str[i] != '\0'; i++)
    Len = Len + 1;
printf ("Length of string is %d", Len);
getch();
}

```

7.11
 Write a program that will count the number of occurrences of a specified character in a given line of text.

Program:

```

#include<stdio.h>
void main()

char Str[50], CheckChar;
int i, Count, Len;
Count = 0;
printf ("Enter a string : -\n");

```

```

scanf ("%[^\\n]s", &str);
len = strlen(str);
fflush (stdin);
printf ("Enter a character: --(n)");
scanf ("%c", &checkChar);
for (i=0; i<len; i++)
    if (CheckChar == str[i])
        count = count + 1;
printf ("Number of occurrences of %c is %d", checkChar, count);
}

```

7.12) Write a Program to read size $m \times n$ and print its transpose.

```

#include<stdio.h>
void main()
{
    int A[MAX][MAX], C[MAX][MAX];
    int row, col, i, j;
    printf ("Enter Ent Number of Rows: --ln");

```

```

Scarf ("y.d", &Row);
printf ("Enter Number of Column: -->n");
Scarf ("y.d", &Col);
printf ("Enter matrix: -->n");
for (i=0; i<Row; i++)
    for (j=0; j<Col; j++)
        Scarf ("y.d", &AC[i][j]);
printf ("Matrix: -->n");
for (i=0; i<Row; i++)
{
    for (j=0; j<Col; j++)
        printf (" y.d", AC[i][j]);
    printf ("\n");
}
for (i=0; i<Col; i++)
    for (j=0; j<Row; j++)
        CC[i][j] = A[j][i];
printf ("Transpose of matrix: -->n");
for (i=0; i<Col; i++)

```

```

    {
        for (j=0; j<Row; j++)
            printf ("%d", C[i][j]);
        printf ("\n");
    }
    getch();
}

```

7.13)

Every book published by International Publishers should carry an International Standard Book Number (ISBN). It is a 10 character part number as shown below.

0-07-041183-2

The first part denotes the region, the second represents publisher, the third identifies the book and the fourth is the check digit. The check digit is computed as follows:

$$\begin{aligned}
 \text{Sum} = & (1 * \text{first digit}) + (2 * \text{second digit}) \\
 & + (3 * \text{third digit}) + \dots + (9 * \text{ninth digit})
 \end{aligned}$$

Check digit is the remainder when sum is divided by 11. Write a program that reads a given ISBN number and check whether it represents a valid ISBN.

Program:

```
#include <stdio.h>
void main()
{
    int ISBN[11];
    int i, j, Sum, Check_Dig;
    Sum = 0;
    printf ("Enter ISBN Number: -\n");
    for (i=1; i<=10; i++)
        scanf ("%d", &ISBN[i]);
    for (i=1; i<=9; i++)
        Sum = Sum + (i * ISBN[i]);
    Check_Dig = Sum % 11;
    if (Check_Dig == ISBN[10])
        printf ("Valid ISBN\n");
    else
        printf ("Invalid ISBN\n");
    getch(); getch();
}
```

714) Write a program to read two matrices A and B and print the following.

a) $A+B$ and

b) $A-B$.

Program:

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

```
void main()
```

```
{
```

```
    int A[MAX][MAX], B[MAX][MAX], C[MAX][MAX]
```

```
    int Row, Col, i, j;
```

```
    printf("Enter Number of Rows: -\n");
```

```
    scanf("%d", &Row);
```

```
    printf("Enter Number of Column: -\n");
```

```
    scanf("%d", &Col);
```

```
    printf("Enter first Matrix : -\n");
```

```
    for (i=0; i<Row; i++)
```

```
        for (j=0; j<Col; j++)
```

```
            scanf("%d", &A[i][j]);
```

```
    printf("Enter Second Matrix : -\n");
```

```

for (i=0; i<Row; i++)
    for (j=0; j<col; j++)
        scanf ("%d", &B[i][j]);
printf ("First Matrix:-\n");
for (i=0; i<Row; i++)
{
    for (j=0; j<col; j++)
        printf ("%d", A[i][j]);
    printf ("\n");
}
printf ("Second Matrix:-\n");
for (i=0; i<Row; i++)
{
    for (j=0; j<col; j++)
        printf ("%d", B[i][j]);
    printf ("\n");
}
for (i=0; i<Row; i++)
    for (j=0; j<col; j++)
        C[i][j] = A[i][j] + B[i][j];
printf ("Addition of Matrix:-\n");

```

```

for (i=0; i<Row; i++)
{
    for (j=0; j<Col; j++)
        printf("x.d", C[i][j]);
    printf("\n");
}

for (i=0; i<Row; i++)
{
    for (j=0; j<Col; j++)
        C[i][j] = A[i][j] - B[i][j];
}

for (i=0; i<Row; i++)
{
    for (j=0; j<Col; j++)
        printf("x.d", C[i][j]);
    printf("\n");
}

getch();
}

```