**C PROGRAMMING ASSIGNMENT:**

**16**

DATE: 17.12.21

SUBMITTED BY: -

NAME: MUKTESH MISHRA

BRANCH: CSE

SECTION: B22

ROLL NO.: 21052258

**1.//wap to copy contents of one array to another array in reverse order**

***Code:***

#include <stdio.h>

*int* main()

{

*int* n, ar1[100], ar2[100], i;

    printf("Specify the length of array\n");

    scanf("%d", &n);

    //taking input of elements

    for (*int* i = 0; i < n; i++)

    {

        printf("Enter the number at index %d\n", i);

        scanf("%d", &ar1[i]);

    }

    //copying elements

    for (i = n; i >= 0; i--)

        ar2[i] = ar1[n - (i+1)];

    //displaying array

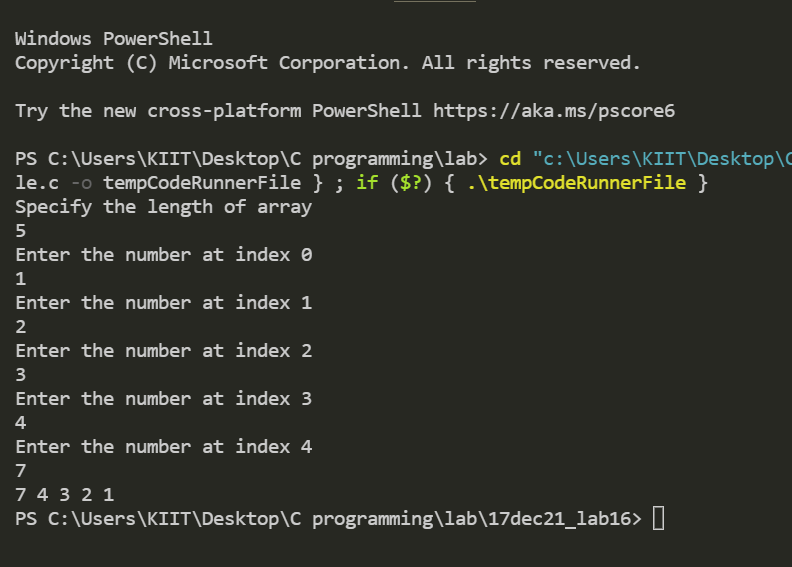
    for (i = 0; i < n; i++)

        printf("%d ", ar2[i]);

    return 0;

}

***Output:***



2.Wap to print the distance between 10 coordinates stored in arrays.

***Code:***

#include <stdio.h>

#include <math.h>

*int* main()

{

*float* dist = 0, y1, x1;

*int* x[10][2], i, j;

    printf("Enter the coordinates of 10 points : \n\n");

    for (*int* i = 0; i < 10; ++i)

    {

        for (*int* j = 0; j < 2; ++j)

        {

            scanf("%d", &x[i][j]);

        }

    }

    printf("\nEntered coordinates : \n");

    for (*int* i = 0; i < 10; ++i)

    {

        for (*int* j = 0; j < 2; ++j)

            printf("%d ", x[i][j]);

        printf("\n");

    }

    for (*int* i = 0; i < 9; i++)

    {

        x1 = pow(x[i + 1][0] - x[i][0], 2);

        y1 = pow(x[i + 1][1] - x[i][1], 2);

        dist += sqrt(x1 + y1);

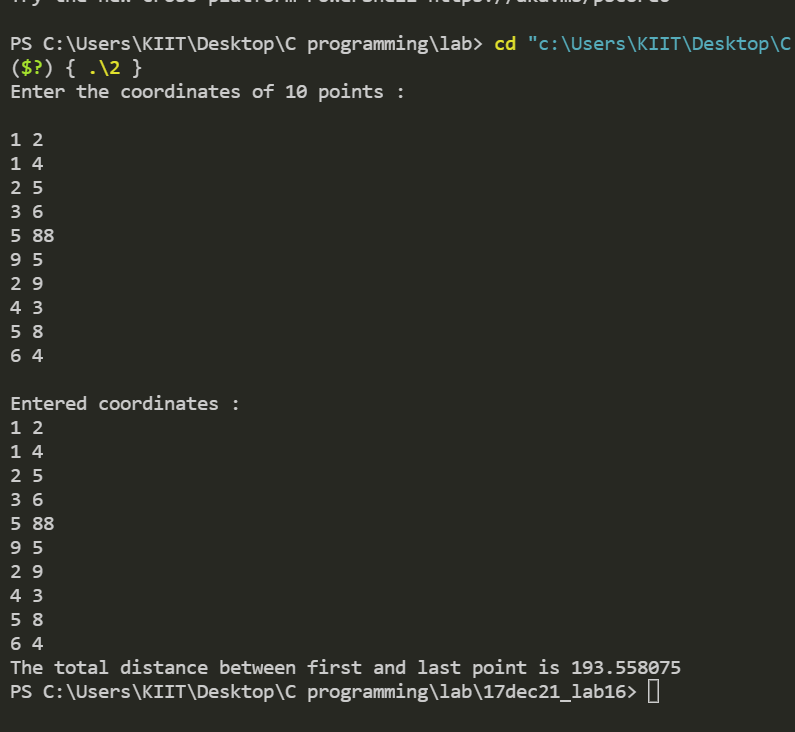
    }

    printf("The total distance between first and last point is %f", dist);

    return 0;

}

***Output:***



3. WAP to find the largest number in a matrix

***Code:***

#include <stdio.h>

*int* main(*int* *argc*, *char* const \**argv*[])

{

*int* a[5][5], largest = 0;

    for (*int* i = 0; i < 5; i++)

    {

        for (*int* j = 1; j <= 5; j++)

        {

            printf("Enter number at %d and %d position of array", i, j);

            scanf("%d", &a[i][j]);

        }

    }

    for (*int* i = 0; i < 5; i++)

    {

        for (*int* j = 1; j <= 5; j++)

        {

            if (a[i][j] > largest)

            {

                largest = a[i][j];

            }

        }

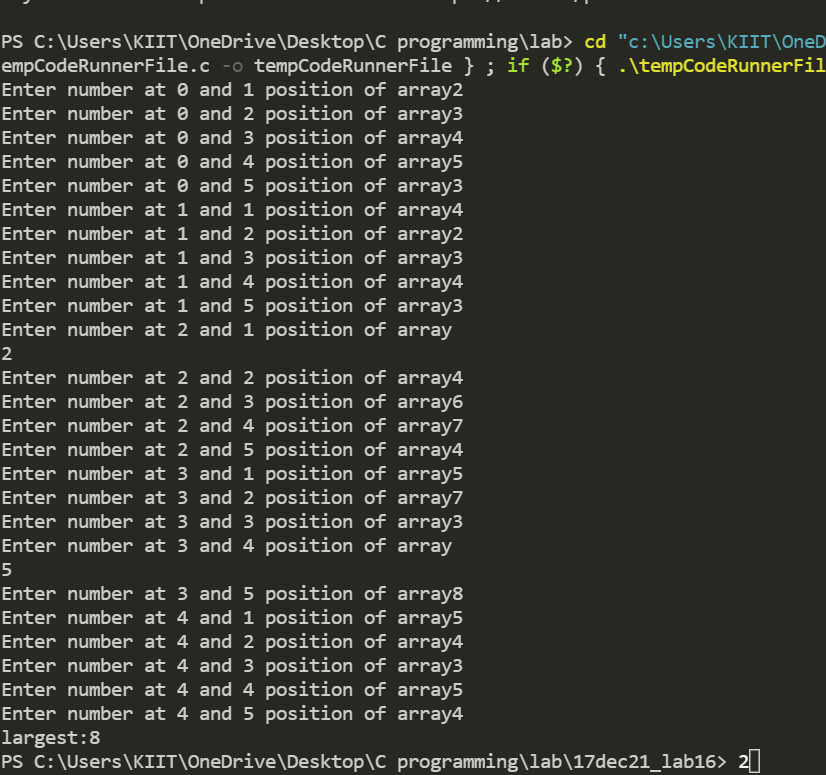
    }

    printf("largest:%d\n", largest);

    return 0;

}

***Output:***



1. **WAP to print elements of array in matrix order**

Code:

#include <stdio.h>

*int* main(*int* *argc*, *char* const \**argv*[])

{

*int* a[5][5];

    for (*int* i = 0; i < 5; i++)

    {

        for (*int* j = 1; j <= 5; j++)

        {

            printf("Enter number at %d and %d position of array", i, j);

            scanf("%d", &a[i][j]);

        }

    }

    for (*int* i = 0; i < 5; i++)

    {

        for (*int* j = 1; j <= 5; j++)

        {

            printf("%d\t", a[i][j]);

        }

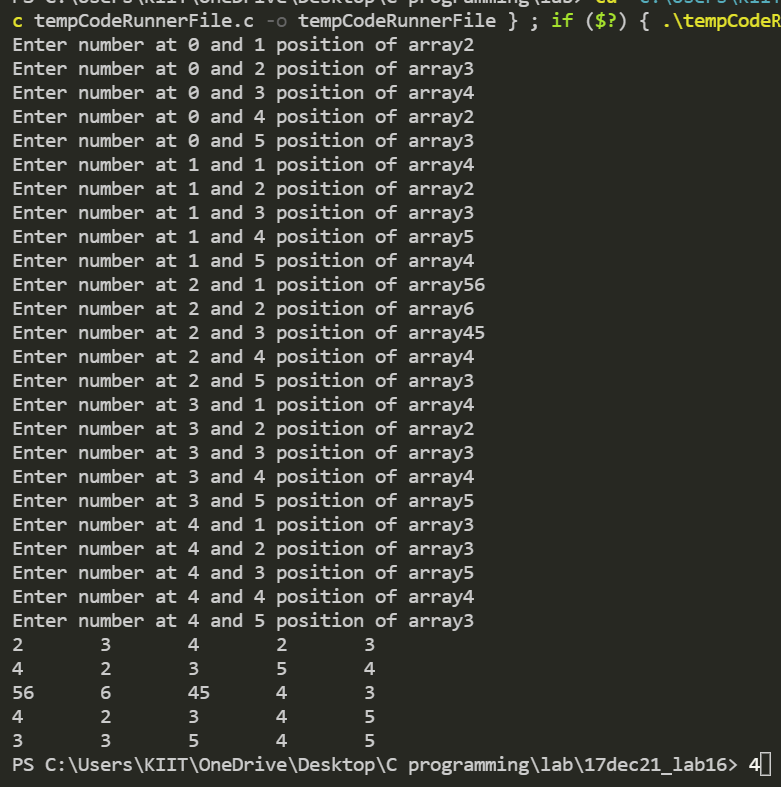
        printf("\n");

    }

    return 0;

}

***Output:***



1. **WAP to transpose a 5x5 matrix**

***Code:***

#include <stdio.h>

*int* main(*int* *argc*, *char* const \**argv*[])

{

*int* A[5][5], B[5][5];

    printf("Enter elements in matrix of size 4x4: \n");

    for (*int* row = 0; row < 5; row++)

    {

        for (*int* col = 0; col < 5; col++)

        {

            scanf("%d", &A[row][col]);

        }

    }

    /\* Find transpose of matrix A\*/

    for (*int* row = 0; row < 5; row++)

    {

        for (*int* col = 0; col < 5; col++)

        {

            B[row][col] = A[col][row];

        }

    }

    printf("The entered matrix is:\n");

    for (*int* row = 0; row < 5; row++)

    {

        for (*int* col = 0; col < 5; col++)

        {

            printf("%d\t", A[row][col]);

        }

        printf("\n");

    }

    printf("The transpose of matrix is:\n");

    for (*int* row = 0; row < 5; row++)

    {

        for (*int* col = 0; col < 5; col++)

        {

            printf("%d\t", B[row][col]);

        }

        printf("\n");

    }

    return 0;

}

***Output:***

