**Insertion sort**

#include <stdio.h>

#include <stdlib.h>

void Swap(int \*x, int \*y)

{

    int temp = \*x;

    \*x = \*y;

    \*y = temp;

}

void InsertionSort(int A[], int n)

{

    for(int i=1; i<n; i++)

    {

        int j=i-1;

        int x=A[i];

        while(j>-1 && A[j]>x)

        {

            A[j+1]=A[j];

            j--;

        }

        A[j+1]=x;

    }

}

int main()

{

    int n;

    printf("Enter the number of elements : ");

    scanf("%d", &n);

    int A[n];

    printf("Enter the elements : \n");

    for (int i = 0; i < n; i++)

    {

        scanf("%d", &A[i]);

    }

    InsertionSort(A, n);

    printf("Sorted Array : \n");

    for(int i=0; i<n; i++)

    {

        printf("%d\t",A[i]);

    }

    return 0;

}

**output:**

Enter the number of elements : 9

Enter the elements :

8 4 2 6 7 5 3 9 1

4 8 2 6 7 5 3 9 1

2 4 8 6 7 5 3 9 1

2 4 6 8 7 5 3 9 1

2 4 6 7 8 5 3 9 1

2 4 5 6 7 8 3 9 1

2 3 4 5 6 7 8 9 1

2 3 4 5 6 7 8 9 1

1 2 3 4 5 6 7 8 9

Sorted Array :

1 2 3 4 5 6 7 8 9

**Selection sort**

#include <stdio.h>

#include <stdlib.h>

void Swap(int \*x, int \*y)

{

    int temp = \*x;

    \*x = \*y;

    \*y = temp;

}

void SelectionSort(int A[], int n)

{

    int i, j, k;

    for (i = 0; i < n - 1; i++)

    {

        for (j = k = i; j < n; j++)

        {

            if (A[j] < A[k])

                k = j;

        }

        Swap(&A[i], &A[k]);

    }

}

int main()

{

    int n;

    printf("Enter the number of elements : ");

    scanf("%d", &n);

    int A[n];

    printf("Enter the elements : \n");

    for (int i = 0; i < n; i++)

    {

        scanf("%d", &A[i]);

    }

    SelectionSort(A, n);

    printf("Sorted Array : \n");

    for(int i=0; i<n; i++)

    {

        printf("%d \t",A[i]);

    }

    return 0;

}

**Output:**

Enter the number of elements : 9

Enter the elements :

8 4 2 6 7 5 3 9 1

1 4 2 6 7 5 3 9 8

1 2 4 6 7 5 3 9 8

1 2 3 6 7 5 4 9 8

1 2 3 4 7 5 6 9 8

1 2 3 4 5 7 6 9 8

1 2 3 4 5 6 7 9 8

1 2 3 4 5 6 7 9 8

1 2 3 4 5 6 7 8 9

1 2 3 4 5 6 7 8 9

Sorted Array :

1 2 3 4 5 6 7 8 9