1. Problem Statement: Selection Sort

**Problem Analysis:**

The selection sort algorithm sorts an array by repeatedly finding the minimum element (considering ascending order) from unsorted part and putting it at the beginning. The algorithm maintains two subarrays in a given array.

*1.* The subarray which is already sorted.

*2.* Remaining subarray which is sorted.

Following are the steps involved in selection sort(for sorting a given array in ascending order):

*1.* Starting from the first element, we search the smallest element in the array, and replace it with the element in the first position.

*2.* We then move on to the second position, and look for smallest element present in the subarray, starting from index 1, till the last index.

*3.* We replace the element at the second position in the original array, or we can say at the first position in the subarray, with the second smallest element.

*4.* This is repeated, until the array is completely sorted.

**Algorithm:**

SelectionSort(A,n)

*//Sort the array A[1:n] into non-decreasing order.*

{

for i:= 1 to n do;

{

iMin: = i;

for j: = i+1 to n do;

{

if(A[j]<A[iMin]) then iMin: = j;

temp: = A[i];

A[i]: = A[iMin];

A[iMin]: = temp;

}

}

}

**Source Code:**

#include<iostream>

using namespace std;

void SelectionSort(int A[], int n)

{

for(int i=0;i<n-1;i++)

{

int iMin = i;

for(int j=i+1;j<n;j++)

{

if(A[j]<A[iMin])

iMin = j;

}

int temp = A[i];

A[i] = A[iMin];

A[iMin] = temp

}

}

int main()

{

int A[30],i,n;

cout<<"Enter array size: \n";

cin>>n;

cout<<"Enter array elements: \n";

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

cin>>A[i];

SelectionSort(A,n);

cout<<"Sorted array elements: \n";

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

cout<<A[i]<<" ";

return 0;

}

**Sample Input:**

Enter array size:

7

Enter array elements:

99 55 11 88 22 77 33

**Sample Output:**

Sorted array elements:

11 22 33 55 77 88 99