**SELECTION SORT**

**DESCRIPTION**

Selection sort will not require no more than n-1 interchanges. Suppose x is an array of size n stored in memory. The selection sort algorithm first selects the smallest element in the array x and place it at array position 0, then it selects the next smallest element in the array x and place it at position 1.It simply continuous this procedure until it places the largest element in the last position of the array.

The array is passed through (n-1) times and smallest element is placed in its respective position in the array as detailed below.

Pass 1: Find the location j of the smallest element in the array x[0],x[1],......x[n-1], and then interchange x[j] with x[0].Then x[0] is sorted.

Pass 2:Leave the first element and then find the location j of the smallest element in the sub-array x[1],x[2],......x[n-1], and then interchange x[1] with x[j].Then x[0] and x[1] are sorted.

Pass 3:Leave the first two element and then find the location j of the smallest element in the sub-array x[2],x[3],......x[n-1], and then interchange x[2] with x[j].Then x[0], x[1] and x[2]are sorted.

Pass (n-1):Fiind the location j of the smaller of the elements x[n-2] and x[n-1], and then interchange x[j] and x[n-2].Then x[0],x[1],.....x[n-2] are sorted. Of course,during this pass x[n-1] will be the biggest element and then so entire array is sorted.

PROGRAM:-

#include <stdio.h>

void selectionsort(int low int high);

int a[20];

int main()

{

int num,i=0;

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

scanf("%d",&num);

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

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

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

selectionsort(0,num-1);

printf("the elements after sorting:");

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

printf("%d ",a[i]);

return 0;

}

void selectionsort(int low, int high)

{

int i=0,j=0,temp=0,min;

for(i=low;i<=high;i++)

{

min=i;

for(j=i+1;j<=high;j++)

{

if(a[j]<a[min])

min=j;

}

temp=a[i];

a[i]=a[min];

a[min]=temp;

}

}

