|  |
| --- |
| **Subject: DAA Class: S.E.(Comp)**    **Practical No.: 5 Date:** |

**AIM:** Implement selection method to find Minimum/ Maximum, Kth smallest element.

**Title:** Write algorithm and program for selection method to find Minimum/Maximum smallest element.

**Theory:**

Given an array and a number k where k is smaller than the size of the array, we need to find the k’th smallest element in the given array. It is given that all array elements are distinct.

**Example:**

Input: arr[] = {7, 10, 4, 3, 20, 15}

k = 3

Output: 7

Input: arr[] = {7, 10, 4, 3, 20, 15}

k = 4

Output: 10

A Simple Solution is to sort the given array using a O(nlogn) sorting algorithm like [Merge Sort](http://geeksquiz.com/merge-sort/), [Heap Sort](http://geeksquiz.com/heap-sort/), etc and return the element at index k-1 in the sorted array. Time Complexity of this solution is O(nLogn).

**Algorithm:**

1. Divide the array in to n/5 lists of 5 elements each.
2. Find the median in each sub array of 5 elements.
3. Recursively ﬁnd the median of all the medians, lets call it M
4. Partition the array in to two sub array 1st sub-array contains the elements larger than M , lets say this sub-array is a1 , while other sub-array contains the elements smaller then M., lets call this sub-array a2.
5. If k <= |a1|, return selection (a1,k).
6. If k− 1 = |a1|, return M.
7. If k> |a1| + 1, return selection(a2,k −a1 − 1).

**Source Code:**

#include<stdio.h>

#include<conio.h>

int main()

{

int array[100],length,i,j,temp,n;

clrscr();

printf("Enter The length Of The Array: ");

scanf("%d",&length);

printf("\nEnter The Numbers: ");

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

{

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

}

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

{

for(j=0;j<length-1;j++)

{

if(array[j]>array[j+1])

{

temp=array[j];

array[j]=array[j+1];

array[j+1]=temp;

}

}

}

printf("The new array is: ");

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

{

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

}

printf("\nEnter Which Smallest Number You want?\n");

scanf("%d",&n);

printf("\nThe %d th smallest number is: %d",n,array[n-1]);

getch();

return 0;

}

**Output:**

