

NAME: Aditya Somani
PRN: 71901204L
ROLL: T1851061
SL-5(Group'B')

Assignment No. (B)1

AIM: Write a program to find Maximum and Minimum element in an array using Divide and Conquer strategy and verify the time complexity.

OBJECTIVE:

1. To understand the concept of Divide and Conquer.
2. To find Maximum and Minimum element in an array.
3. To find Time Complexity of algorithms

THEORY:

What is Divide and Conquer?

Divide-and-conquer is a top-down technique for designing algorithms that consists of dividing the problem into smaller subproblems hoping that the solutions of the subproblems are easier to find and then composing the partial solutions into the solution of the original problem.

Little more formally, divide-and-conquer paradigm consists of following major phases:

- Divide: Breaking the problem into several sub-problems that are similar to the original problem but smaller in size,
- Conquer: Solve the sub-problem recursively (successively and independently), and then
- Combine: Combine these solutions to subproblems to create a solution to the original problem.

Advantages

1. Solving difficult problems
2. Algorithm efficiency parallelism
3. Memory access
4. Roundoff control

Maximum and Minimum element in an array:

In this approach, the array is divided into two halves. Then using recursive approach maximum and minimum numbers in each halves are found. Later, return the maximum of two maxima of each half and the minimum of two minima of each half.

In this given problem, the number of elements in an array is $y-x+1$, where y is greater than or equal to x . The function $\text{Maxmin}(x,y)$ will return the maximum and minimum values of an array $\text{numbers}[x...y]$. Complexity Analysis:

Let $T(n)$ be the number of comparisons made by $\text{Maxmin}(x,y)$, where the number of elements $n=y-x+1$.

If $T(n)$ represents this no., then the resulting recurrence relations is:

$$T(n) = T\left(\frac{n}{2}\right) + T\left[\frac{n}{2}\right] + 2 \quad n > 2$$
$$T(n) = 1 \quad n = 1$$

So time complexity is $O(n)$ after solving above recurrence equation.

PROGRAM:-

```
#include<stdio.h>
#include<stdio.h>
int max, min;
int a[100];
void maxmin(int i, int j)
{
    int max1, min1, mid; if(i==j)
    {
        max = min = a[i];
    }
    else
    {
        if(i == j-1)
        {
            if(a[i] < a[j])
            {
                max = a[j];
                min = a[i];
            }
            else
            {
                max = a[i];
                min = a[j];
            }
        }
        else
        {
            mid = (i+j)/2; maxmin(i, mid);
            max1 = max; min1 = min; maxmin(mid+1, j);
            if(max < max1)
            {
                max = max1;
            }
            if(min > min1)
            {
                min = min1;
            }
        }
    }
}

int main ()
{
    int i, num;
    printf ("\nEnter the total number of numbers : ");
    scanf ("%d",&num);
    printf ("Enter the numbers : \n"); for (i=1;i<=num;i++)
    {
        scanf ("%d",&a[i]);
    }

    max = a[0];
    min = a[0]; maxmin(1, num);
    printf ("Minimum element in an array : %d\n", min); printf ("Maximum element in an array : %d\n",
    max); return 0;
}
```

OUTPUT:-

```
1 #include<stdio.h>
2 #include<stdio.h>
3 int max, min;
4 int a[100];
5 void maxmin(int i, int j)
6 {
7     int max1, min1, mid; if(i==j)
8     {
9         max = min = a[i];
10    }
11    else
12    {
13        if(i == j-1)
14        {
15            if(a[i] < a[j])
16            {
17                max = a[j];
18                min = a[i];
19            }
20            else
```

```
/tmp/BFFxaJ0532.o
Enter the total number of numbers : 5
Enter the numbers :
12
23
34
35
67
Minimum element in an array : 12
Maximum element in an array : 67

Activate Windows
Go to Settings to activate Windows.
```

```
1 #include<stdio.h>
2 #include<stdio.h>
3 int max, min;
4 int a[100];
5 void maxmin(int i, int j)
6 {
7     int max1, min1, mid; if(i==j)
8     {
9         max = min = a[i];
10    }
11    else
12    {
13        if(i == j-1)
14        {
15            if(a[i] < a[j])
16            {
17                max = a[j];
18                min = a[i];
19            }
20            else
```

```
/tmp/BFFxaJ0532.o
Enter the total number of numbers : 10
Enter the numbers :
1223
3445
89799
2121
445654
1212
56547
24325
756876
898723
Minimum element in an array : 1212
Maximum element in an array : 898723

Activate Windows
Go to Settings to activate Windows.
```