

```
/*
Find Maximum and Minimum element from an array of integer using
Divide and Conquer approach as well as dynamic programming approach.
ARKAPRATIM GHOSH 13000121058
*/
#include<stdio.h>
#include<stdlib.h>
#include<ctype.h>
#include<math.h>
#include<limits.h>

int maximum(int i,int j,int *p,int *max)
{
    int mid;
    if(i==j)
    {
        *max=p[i];
        return *max;
    }
    else if(i==j-1)
    {
        if(p[i]>=p[j]) *max=p[i];
        else *max=p[j];
        return *max;
    }
    else
    {
        mid=(i+j)/2;
        *max=maximum(i,mid,p,max);
        *max=maximum(mid+1,j,p,max);
        return *max;
    }
}

int minimum(int i,int j,int *p,int *min)
{
    int mid;
    if(i==j)
    {
        *min=p[i];
        return *min;
    }
    else if(i==j-1)
    {
        if(p[i]<=p[j]) *min=p[i];
        else *min=p[j];
        return *min;
    }
    else
```

```
{
    mid=(i+j)/2;
    *min=minimum(i,mid,p,min);
    *min=minimum(mid+1,j,p,min);
    return *min;
}

void main()
{
    int *p,n;
    int i;
    int max;
    int min;
    printf("Enter the number of elements: ");
    scanf("%d",&n);
    fflush(stdin);
    p=(int *)malloc(n*sizeof(int));
    for(i=0;i<n;i++)
    {
        printf("Enter element %d: ",(i+1));
        scanf("%d",&p[i]);
        fflush(stdin);
    }
    max=maximum(0,n-1,p,&max);
    min=minimum(0,n-1,p,&min);
    printf("Maximum element is %d\n",max);
    printf("Minimum element is %d\n",min);
}
```