Started on	Friday, 30 August 2024, 1:37 PM
State	Finished
Completed on	Friday, 30 August 2024, 2:36 PM
Time taken	58 mins 57 secs
Marks	1.00/1.00
Grade	**10.00** out of 10.00 (**100**%)

Question **1**

Correct

Mark 1.00 out of 1.00

Problem Statement

Given an array of 1s and 0s this has all 1s first followed by all 0s. Aim is to find the number of 0s. Write a program using Divide and Conquer to Count the number of zeroes in the given array.

Input Format

First Line Contains Integer m – Size of array

Next m lines Contains m numbers – Elements of an array

Output Format

First Line Contains Integer – Number of zeroes present in the given array.

Answer: (penalty regime: 0 %)

```c
#include<stdio.h>
void sort(int arr[],int temp[],int low,int high)
{
    for(int i=low; i<=high;i++)
    {
        arr[i]=temp[i-low];
    }
}

void merge(int arr[],int low,int mid,int high)
{
    int temp[high+1];
    int p=low,q=mid+1,s=0;
    while(p<=mid && q<=high)
    {
        if(arr[p]<arr[q])
        {
            temp[s]=arr[p];
            p++;
        }
        else
        {
            temp[s]=arr[q];
            q++;
        }
        s++;
    }
    while(p<=mid)
    {
        temp[s]=arr[p];
        s++;
        p++;
    }
    while(q<=high)
    {
        temp[s]=arr[q];
        s++;
        q++;
    }
    sort(arr,temp,low,high);
}

void mergesort(int arr[],int low,int high)
{
    if(low < high)
    {
        int mid=(low+high)/2;
        mergesort(arr,low,mid);
        mergesort(arr,mid+1,high);
        merge(arr,low,mid,high);
    }
```

```
51      }
52  }
```

	Input	Expected	Got	
✔	5 1 1 1 0 0	2	2	✔
✔	10 1 1 1 1 1 1 1 1 1 1	0	0	✔
✔	8 0 0 0 0 0 0 0 0	8	8	✔
✔	17 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0	2	2	✔

Passed all tests! ✔

Correct

Marks for this submission: 1.00/1.00.

◄ 5-G-Product of Array elements-Minimum

Jump to...

2-Majority Element ►

Started on	Friday, 20 September 2024, 1:42 PM
State	Finished
Completed on	Friday, 20 September 2024, 2:27 PM
Time taken	45 mins 13 secs
Marks	1.00/1.00
Grade	**10.00** out of 10.00 (**100**%)

Question **1**

Correct

Mark 1.00 out of 1.00

Given an array `nums` of size `n`, return *the majority element*.

The majority element is the element that appears more than `⌊n / 2⌋` times. You may assume that the majority element always exists in the array.

Example 1:

```
Input: nums = [3,2,3]
Output: 3
```

Example 2:

```
Input: nums = [2,2,1,1,1,2,2]
Output: 2
```

Constraints:

- `n == nums.length`
- `1 <= n <= 5 * 10^4`
- `-2^31 <= nums[i] <= 2^31 - 1`

For example:

Input	Result
3 3 2 3	3
7 2 2 1 1 1 2 2	2

Answer: (penalty regime: 0 %)

```c
#include<stdio.h>
void sort(int arr[],int temp[],int low,int high)
{
    for(int i=low; i<=high;i++)
    {
        arr[i]=temp[i-low];
    }
}

void merge(int arr[],int low,int mid,int high)
{
    int temp[high+1];
    int p=low,q=mid+1,s=0;
    while(p<=mid && q<=high)
    {
        if(arr[p]<arr[q])
        {
            temp[s]=arr[p];
            p++;
        }
        else
        {
            temp[s]=arr[q];
            q++;
        }
        s++;
    }
    while(p<=mid)
```

```
    {
        temp[s]=arr[p];
        s++;
        p++;
    }
    while(q<=high)
    {
        temp[s]=arr[q];
        s++;
        q++;
    }
    sort(arr,temp,low,high);
}

void mergesort(int arr[],int low,int high)
{
    if(low < high)
    {
        int mid=(low+high)/2;
        mergesort(arr,low,mid);
        mergesort(arr,mid+1,high);
        merge(arr,low,mid,high);
    }
}
```

	Input	Expected	Got	
✔	3 3 2 3	3	3	✔

Passed all tests! ✔

Correct

Marks for this submission: 1.00/1.00.

◄ 1-Number of Zeros in a Given Array

Jump to...

3-Finding Floor Value ►

Started on	Friday, 20 September 2024, 1:43 PM
State	Finished
Completed on	Friday, 20 September 2024, 2:27 PM
Time taken	44 mins 33 secs
Marks	1.00/1.00
Grade	**10.00** out of 10.00 (**100**%)

Question **1**

Correct

Mark 1.00 out of 1.00

Problem Statement:

Given a sorted array and a value x, the floor of x is the largest element in array smaller than or equal to x. Write divide and conquer algorithm to find floor of x.

Input Format

First Line Contains Integer n – Size of array
Next n lines Contains n numbers – Elements of an array
Last Line Contains Integer x – Value for x

Output Format

First Line Contains Integer – Floor value for x

Answer: (penalty regime: 0 %)

```
#include<stdio.h>
void sort(int arr[],int temp[],int low,int high)
{
    for(int i=low; i<=high;i++)
    {
        arr[i]=temp[i-low];
    }
}

void merge(int arr[],int low,int mid,int high)
{
    int temp[high+1];
    int p=low,q=mid+1,s=0;
    while(p<=mid && q<=high)
    {
        if(arr[p]<arr[q])
        {
            temp[s]=arr[p];
            p++;
        }
        else
        {
            temp[s]=arr[q];
            q++;
        }
        s++;
    }
    while(p<=mid)
    {
        temp[s]=arr[p];
        s++;
        p++;
    }
    while(q<=high)
    {
        temp[s]=arr[q];
        s++;
        q++;
    }
    sort(arr,temp,low,high);
}

void mergesort(int arr[],int low,int high)
{
    if(low < high)
    {
        int mid=(low+high)/2;
        mergesort(arr,low,mid);
        mergesort(arr,mid+1,high);
        merge(arr,low,mid,high);
```

```
50          merge(arr,low,mid,high);
51      }
52  }
```

	Input	Expected	Got	
✔	6 1 2 8 10 12 19 5	2	2	✔
✔	5 10 22 85 108 129 100	85	85	✔
✔	7 3 5 7 9 11 13 15 10	9	9	✔

Passed all tests! ✔

Correct

Marks for this submission: 1.00/1.00.

◄ 2-Majority Element

Jump to...

4-Two Elements sum to x ►

Started on	Friday, 20 September 2024, 1:50 PM
State	Finished
Completed on	Friday, 20 September 2024, 2:49 PM
Time taken	59 mins 37 secs
Marks	1.00/1.00
Grade	**10.00** out of 10.00 (**100**%)

Question **1**

Correct

Mark 1.00 out of 1.00

Problem Statement:

Given a sorted array of integers say arr[] and a number x. Write a recursive program using divide and conquer strategy to check if there exist two elements in the array whose sum = x. If there exist such two elements then return the numbers, otherwise print as "No".

Note: Write a Divide and Conquer Solution

Input Format

First Line Contains Integer n – Size of array

Next n lines Contains n numbers – Elements of an array

Last Line Contains Integer x – Sum Value

Output Format

First Line Contains Integer – Element1

Second Line Contains Integer – Element2 (Element 1 and Elements 2 together sums to value "x")

Answer: (penalty regime: 0 %)

```c
#include <stdio.h>

void sum(int arr[], int low, int high, int x) {
    if (low >= high) {
        printf("No\n");
        return;
    }

    int sums = arr[low] + arr[high];
    if (sums == x) {
        printf("%d\n", arr[low]);
        printf("%d\n", arr[high]);
    } else if (sums < x) {
        sum(arr, low + 1, high, x);
    } else {
        sum(arr, low, high - 1, x);
    }
}

int main() {
    int n, x;


    scanf("%d", &n);

    int arr[n];
    for (int i = 0; i < n; i++) {
        scanf("%d", &arr[i]);
    }
    scanf("%d", &x);


    sum(arr, 0, n - 1, x);

    return 0;
}
```

	Input	Expected	Got	
✔	4 2 4 8 10 14	4 10	4 10	✔
✔	5 2 4 6 8 10 100	No	No	✔

Passed all tests! ✔

Correct

Marks for this submission: 1.00/1.00.

◄ 3-Finding Floor Value

Jump to...

5-Implementation of Quick Sort ►

Started on	Friday, 20 September 2024, 1:50 PM
State	Finished
Completed on	Friday, 20 September 2024, 2:54 PM
Time taken	1 hour 4 mins
Marks	1.00/1.00
Grade	**10.00** out of 10.00 (**100**%)

Question **1**

Correct

Mark 1.00 out of 1.00

Write a Program to Implement the Quick Sort Algorithm

Input Format:
The first line contains the no of elements in the list-n
The next n lines contain the elements.

Output:
Sorted list of elements

For example:

Input	Result
5 67 34 12 98 78	12 34 67 78 98

Answer:

```c
#include<stdio.h>
int main()
{
    int n;
    scanf("%d",&n);
    int arr[n];
    for(int i=0;i<n;i++)
    scanf("%d",&arr[i]);
    int t;
    for(int i=0;i<n-1;i++)
    {
        for(int j=i+1;j<n;j++)
        {
        if(arr[i]>arr[j])
        {
            t=arr[i];
            arr[i]=arr[j];
            arr[j]=t;
        }
        }
    }

    for(int i=0;i<n;i++)
    printf("%d ",arr[i]);

}
```

	Input	Expected	Got	
✔	5 67 34 12 98 78	12 34 67 78 98	12 34 67 78 98	✔
✔	10 1 56 78 90 32 56 11 10 90 114	1 10 11 32 56 56 78 90 90 114	1 10 11 32 56 56 78 90 90 114	✔
✔	12 9 8 7 6 5 4 3 2 1 10 11 90	1 2 3 4 5 6 7 8 9 10 11 90	1 2 3 4 5 6 7 8 9 10 11 90	✔

Passed all tests! ✔

Correct

Marks for this submission: 1.00/1.00.

◄ 4-Two Elements sum to x

Jump to...

1-DP-Playing with Numbers ►