



ARIFA H 2024-CSE

A2

Started on Wednesday, 6 August 2025, 10:21 AM

State Finished

Completed on Wednesday, 6 August 2025, 10:44 AM

Time taken 22 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

Convert the following algorithm into a program and find its time complexity using the counter method.

```
void function (int n)
{
    int i= 1;
```

```
    int s =1;
```

```
    while(s <= n)
    {
        i++;
        s += i;
    }
}
```

Note: No need of counter increment for declarations and scanf() and count variable printf() statements.

Input:

A positive Integer n

Output:

Print the value of the counter variable

For example:

Input	Result
9	12

Answer: (penalty regime: 0 %)

```
1 #include<stdio.h>
2 void function(int);
3 int count;
4 int main()
5 {
6     int n;
7     scanf("%d",&n);
8     function(n);
9     printf("%d",count);
10
11
12 }
13 void function(int n)
14 {
15     int i=1;
16     count++;
17     int s=1;
18     count++;
19     while(s<=n){
20         count++;
21         i++;
22         count++;
23         s+=i;
24         count++;
25     }count++;
26
27 }
28
29
```

	Input	Expected	Got	
✓	9	12	12	✓
✓	4	9	9	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

[Back to Course](#)



ARIFA H 2024-CSE

A2

Started on Wednesday, 6 August 2025, 10:44 AM

State Finished

Completed on Wednesday, 6 August 2025, 11:00 AM

Time taken 15 mins 41 secs

Marks 1.00/1.00

Grade **10.00** out of 10.00 (**100%**)

Question 1 | Correct Mark 1.00 out of 1.00

Convert the following algorithm into a program and find its time complexity using the counter method.

```
void func(int n)
{
    if(n==1)
    {
        printf("*");
    }
    else
    {
        for(int i=1; i<=n; i++)
        {
            for(int j=1; j<=n; j++)
            {
                printf("*");
                printf("*");
                break;
            }
        }
    }
}
```

Note: No need of counter increment for declarations and scanf() and count variable printf() statements.

Input:

A positive Integer n

Output:

Print the value of the counter variable

Answer: (penalty regime: 0 %)

```
1 #include<stdio.h>
2 void function(int);
3 int count=0;
4 int main(){
5     int n;
6     scanf("%d",&n);
7     function(n);
8     printf("%d",count);
9 }
10 void function( int n){
11     if(n==1){
12         count++;
13         //printf("*");
14     }
15     else{
16         count++;
17         for(int i=1;i<=n;i++){
18             count++;
19             for(int j=0;j<=n;j++){
20                 count++;
21                 //printf("*");
22                 count++;
23                 //printf("*");
24                 count++;
25                 break;
26             }count++;
27         }count++;
28 }
```

29 }

	Input	Expected	Got	
✓	2	12	12	✓
✓	1000	5002	5002	✓
✓	143	717	717	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

[Back to Course](#)



ARIFA H 2024-CSE

A2

Started on Wednesday, 6 August 2025, 11:00 AM

State Finished

Completed on Wednesday, 6 August 2025, 11:09 AM

Time taken 9 mins 30 secs

Marks 1.00/1.00

Grade 10.00 out of 10.00 (100%)

Question 1 | Correct Mark 1.00 out of 1.00

Convert the following algorithm into a program and find its time complexity using counter method.

```
Factor(num) {
{
    for (i = 1; i <= num; ++i)
    {
        if (num % i == 0)
        {
            printf("%d ", i);
        }
    }
}
```

Note: No need of counter increment for declarations and scanf() and counter variable printf() statement.

Input:

A positive Integer n

Output:

Print the value of the counter variable

Answer:

```
1 #include<stdio.h>
2 void factor(int);
3 int count=0;
4 int main()
5 {
6     int n;
7     scanf("%d",&n);
8     factor(n);
9     printf("%d",count);
10 }
11 void factor(int n){
12     for(int i=1;i<=n;i++)
13     {
14         count++;
15         if(n%i==0){
16             count++;
17             //printf("%d",i);
18         }
19     }count++;
20 }
```

	Input	Expected	Got	
✓	12	31	31	✓
✓	25	54	54	✓
✓	4	12	12	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

[Back to Course](#)



ARIFA H 2024-CSE

A2

Started on Wednesday, 6 August 2025, 11:10 AM

State Finished

Completed on Wednesday, 6 August 2025, 11:14 AM

Time taken 4 mins 28 secs

Marks 1.00/1.00

Grade **10.00** out of 10.00 (**100%**)

Question 1 | Correct Mark 1.00 out of 1.00

Convert the following algorithm into a program and find its time complexity using counter method.

```
void function(int n)
{
    int c= 0;
    for(int i=n/2; i<n; i++)
        for(int j=1; j<n; j = 2 * j)
            for(int k=1; k<n; k = k * 2)
                c++;
}
```

Note: No need of counter increment for declarations and scanf() and count variable printf() statements.

Input:

A positive Integer n

Output:

Print the value of the counter variable

Answer:

```
1 #include<stdio.h>
2 void function(int);
3 int count=0;
4 int main(){
5     int n;
6     scanf("%d",&n);
7     function(n);
8     printf("%d",count);
9 }
10 void function(int n){
11     int c=0;
12     count++;
13     for(int i=n/2;i<n;i++){
14         count++;
15         for(int j=1;j<n;j=2*j){
16             count++;
17             for(int k=1;k<n;k=k*2){
18                 count++;
19                 c++;
20                 count++;
21             }count++;
22         }count++;
23     }count++;
24 }
```

	Input	Expected	Got	
✓	4	30	30	✓
✓	10	212	212	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

[Back to Course](#)



ARIFA H 2024-CSE

A2

Started on Wednesday, 6 August 2025, 11:14 AM

State Finished

Completed on Wednesday, 6 August 2025, 11:21 AM

Time taken 6 mins 47 secs

Marks 1.00/1.00

Grade 10.00 out of 10.00 (100%)

Question 1 | Correct Mark 1.00 out of 1.00

Convert the following algorithm into a program and find its time complexity using counter method.

```
void reverse(int n)
{
    int rev = 0, remainder;
    while (n != 0)
    {
        remainder = n % 10;
        rev = rev * 10 + remainder;
        n/= 10;

    }
    print(rev);
}
```

Note: No need of counter increment for declarations and scanf() and count variable printf() statements.

Input:

A positive Integer n

Output:

Print the value of the counter variable

Answer:

```
1 #include<stdio.h>
2 void reverse(int);
3 int count=0;
4 int main()
5 {
6     int n;
7     scanf("%d",&n);
8     reverse(n);
9     printf("%d",count);
10 }
11 void reverse(int n){
12     int rev=0,remainder=0;
13     count++;
14     while(n!=0){
15         count++;
16         remainder=n%10;
17         count++;
18         rev=rev*10+remainder;
19         count++;
20         n/=10;
21         count++;
22     }
23     count++;
24     //print(rev);
25     count++;
26 }
```

	Input	Expected	Got	
✓	12	11	11	✓

	Input	Expected	Got	
✓	1234	19	19	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

[Back to Course](#)

[ARIFA H 2024-CSE](#) ▾**A2****Started on** Monday, 22 September 2025, 7:29 PM**State** Finished**Completed on** Monday, 22 September 2025, 7:45 PM**Time taken** 15 mins 49 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 %)

```

1 #include <stdio.h>
2
3 int countZeroes(int arr[], int low, int high, int size) {
4     if (high >= low) {
5         int mid = low + (high - low) / 2;
6         if ((mid == 0 || arr[mid - 1] == 1) && arr[mid] == 0)
7             return size - mid;
8         if (arr[mid] == 1)
9             return countZeroes(arr, mid + 1, high, size);
10        else
11            return countZeroes(arr, low, mid - 1, size);
12    }
13    return 0;
14 }
15
16 int main() {
17     int m;
18     scanf("%d", &m);
19     int arr[m];
20     for (int i = 0; i < m; i++) {
21         scanf("%d", &arr[i]);
22     }
23     int result = countZeroes(arr, 0, m - 1, m);
24     printf("%d\n", result);
25     return 0;
26 }
27

```

	Input	Expected	Got	
✓	5 1 1 1 0 0	2	2	✓

	Input	Expected	Got	
✓	10 1 1 1 1 1 1 1 1 1	0	0	✓
✓	8 0 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 0 0	2	2	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

[Back to Course](#)



ARIFA H 2024-CSE

A2

Started on Monday, 22 September 2025, 7:46 PM**State** Finished**Completed on** Monday, 22 September 2025, 7:52 PM**Time taken** 6 mins 11 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 $\lfloor n / 2 \rfloor$ 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 * 104`
- `-231 <= nums[i] <= 231 - 1`

For example:

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

Answer: (penalty regime: 0 %)

```

1 #include <stdio.h>
2
3 int majorityElement(int* nums, int numsSize) {
4     int count = 0, candidate = 0;
5     for (int i = 0; i < numsSize; i++) {
6         if (count == 0) {
7             candidate = nums[i];
8             count = 1;
9         } else if (nums[i] == candidate) {
10            count++;
11        } else {
12            count--;
13        }
14    }
15    return candidate;
16 }
17
18 int main() {
19     int n;
20     scanf("%d", &n);
21     int nums[n];
22     for (int i = 0; i < n; i++) {
23         scanf("%d", &nums[i]);
24     }
25     int result = majorityElement(nums, n);
26     printf("%d\n", result);
27 }
```

```
27 }  
28 }  
29 }
```

	Input	Expected	Got	
✓	3	3	3	✓
	3 2 3			

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

[Back to Course](#)





ARIFA H 2024-CSE

A2

Started on Monday, 22 September 2025, 7:52 PM

State Finished

Completed on Monday, 22 September 2025, 8:00 PM

Time taken 7 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 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 %)

```

1 #include <stdio.h>
2
3 int findFloor(int arr[], int low, int high, int x) {
4     if (low > high)
5         return -1;
6     if (x >= arr[high])
7         return arr[high];
8     int mid = (low + high) / 2;
9     if (arr[mid] == x)
10        return arr[mid];
11    if (mid > 0 && arr[mid - 1] <= x && x < arr[mid])
12        return arr[mid - 1];
13    if (x < arr[mid])
14        return findFloor(arr, low, mid - 1, x);
15    return findFloor(arr, mid + 1, high, x);
16 }
17
18 int main() {
19     int n, x;
20     scanf("%d", &n);
21     int arr[n];
22     for (int i = 0; i < n; i++) {
23         scanf("%d", &arr[i]);
24     }
25     scanf("%d", &x);
26     int result = findFloor(arr, 0, n - 1, x);
27     printf("%d\n", result);
28     return 0;
29 }
30

```

	Input	Expected	Got	
✓	6	2	2	✓
	1			
	2			
	8			
	10			
	12			
	19			
	5			

	Input	Expected	Got	
✓	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.

[Back to Course](#)



ARIFA H 2024-CSE

A2

Started on Monday, 22 September 2025, 8:47 PM

State Finished

Completed on Monday, 22 September 2025, 9:27 PM

Time taken 40 mins 9 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 %)

```

1 #include <stdio.h>
2
3 int findPair(int arr[], int left, int right, int x, int* a, int* b) {
4     if (left >= right)
5         return 0;
6
7     int sum = arr[left] + arr[right];
8
9     if (sum == x) {
10        *a = arr[left];
11        *b = arr[right];
12        return 1;
13    } else if (sum > x) {
14        return findPair(arr, left, right - 1, x, a, b);
15    } else {
16        return findPair(arr, left + 1, right, x, a, b);
17    }
18}
19
20 int main() {
21     int n;
22     scanf("%d", &n);
23     int arr[n];
24
25     for (int i = 0; i < n; i++)
26         scanf("%d", &arr[i]);
27
28     int x;
29     scanf("%d", &x);
30
31     int a, b;
32     if (findPair(arr, 0, n - 1, x, &a, &b)) {
33         printf("%d\n", a);
34         printf("%d\n", b);
35     } else {
36         printf("No\n");
37     }
38
39     return 0;
40}
41

```

	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.

[Back to Course](#)



ARIFA H 2024-CSE

A2

Started on Monday, 22 September 2025, 9:27 PM

State Finished

Completed on Monday, 22 September 2025, 9:56 PM

Time taken 29 mins 24 secs

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	12 34 67 78 98
67 34 12 98 78	

Answer:

```

1 #include <stdio.h>
2
3 void swap(int* a, int* b) {
4     int t = *a;
5     *a = *b;
6     *b = t;
7 }
8
9 int partition(int arr[], int low, int high) {
10    int pivot = arr[high];
11    int i = low - 1;
12
13    for (int j = low; j < high; j++) {
14        if (arr[j] < pivot) {
15            i++;
16            swap(&arr[i], &arr[j]);
17        }
18    }
19
20    swap(&arr[i + 1], &arr[high]);
21    return i + 1;
22 }
23
24 void quickSort(int arr[], int low, int high) {
25    if (low < high) {
26        int pi = partition(arr, low, high);
27        quickSort(arr, low, pi - 1);
28        quickSort(arr, pi + 1, high);
29    }
30 }
31
32 int main() {
33    int n;
34    scanf("%d", &n);
35    int arr[n];
36
37    for (int i = 0; i < n; i++)
38        scanf("%d", &arr[i]);
39
40    quickSort(arr, 0, n - 1);
41
42    for (int i = 0; i < n; i++)
43        printf("%d ", arr[i]);
44
45    return 0;
46 }
```

	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.

[Back to Course](#)



ARIFA H 2024-CSE

A2

Started on Monday, 22 September 2025, 7:46 PM**State** Finished**Completed on** Monday, 22 September 2025, 7:52 PM**Time taken** 6 mins 11 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 $\lfloor n / 2 \rfloor$ 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 * 104`
- `-231 <= nums[i] <= 231 - 1`

For example:

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

Answer: (penalty regime: 0 %)

```

1 #include <stdio.h>
2
3 int majorityElement(int* nums, int numsSize) {
4     int count = 0, candidate = 0;
5     for (int i = 0; i < numsSize; i++) {
6         if (count == 0) {
7             candidate = nums[i];
8             count = 1;
9         } else if (nums[i] == candidate) {
10            count++;
11        } else {
12            count--;
13        }
14    }
15    return candidate;
16 }
17
18 int main() {
19     int n;
20     scanf("%d", &n);
21     int nums[n];
22     for (int i = 0; i < n; i++) {
23         scanf("%d", &nums[i]);
24     }
25     int result = majorityElement(nums, n);
26     printf("%d\n", result);
27 }
```

```
27 }  
28 } return v;  
29 }
```

	Input	Expected	Got	
✓	3	3	3	✓
	3 2 3			

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

[Back to Course](#)





ARIFA H 2024-CSE

A2

Started on Monday, 22 September 2025, 7:46 PM**State** Finished**Completed on** Monday, 22 September 2025, 7:52 PM**Time taken** 6 mins 11 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 $\lfloor n / 2 \rfloor$ 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 * 104`
- `-231 <= nums[i] <= 231 - 1`

For example:

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

Answer: (penalty regime: 0 %)

```

1 #include <stdio.h>
2
3 int majorityElement(int* nums, int numsSize) {
4     int count = 0, candidate = 0;
5     for (int i = 0; i < numsSize; i++) {
6         if (count == 0) {
7             candidate = nums[i];
8             count = 1;
9         } else if (nums[i] == candidate) {
10            count++;
11        } else {
12            count--;
13        }
14    }
15    return candidate;
16 }
17
18 int main() {
19     int n;
20     scanf("%d", &n);
21     int nums[n];
22     for (int i = 0; i < n; i++) {
23         scanf("%d", &nums[i]);
24     }
25     int result = majorityElement(nums, n);
26     printf("%d\n", result);
27 }
```

```
27 }  
28 }  
29 }
```

	Input	Expected	Got	
✓	3	3	3	✓
	3 2 3			

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

[Back to Course](#)



ARIFA H 2024-CSE ▾**A2****Started on** Tuesday, 19 August 2025, 10:07 PM**State** Finished**Completed on** Tuesday, 19 August 2025, 10:09 PM**Time taken** 1 min 58 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 of N integer, we have to maximize the sum of $\text{arr}[i] * i$, where i is the index of the element ($i = 0, 1, 2, \dots, N$). Write an algorithm based on Greedy technique with a Complexity $O(n\log n)$.

Input Format:

First line specifies the number of elements-n

The next n lines contain the array elements.

Output Format:

Maximum Array Sum to be printed.

Sample Input:

5

2 5 3 4 0

Sample output:

40

Answer: (penalty regime: 0 %)

```

1 #include <stdio.h>
2 #include <stdlib.h>
3
4 int compare(const void *a, const void *b) {
5     return (*(int *)a - *(int *)b);
6 }
7
8 int main() {
9     int n;
10    scanf("%d", &n);
11    int arr[n];
12    for (int i = 0; i < n; i++) {
13        scanf("%d", &arr[i]);
14    }
15
16    qsort(arr, n, sizeof(int), compare);
17
18    long long result = 0;
19    for (int i = 0; i < n; i++) {
20        result += (long long)arr[i] * i;
21    }
22
23    printf("%lld\n", result);
24    return 0;
25 }
```

	Input	Expected	Got	
✓	5	40	40	✓
	2			
	5			
	3			
	4			
	0			

	Input	Expected	Got	
✓	10 2 2 2 4 4 3 3 5 5 5	191	191	✓
✓	2 45 3	45	45	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

[Back to Course](#)



ARIFA H 2024-CSE

A2

Started on Tuesday, 19 August 2025, 10:10 PM**State** Finished**Completed on** Tuesday, 19 August 2025, 10:11 PM**Time taken** 1 min 23 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 two arrays array_One[] and array_Two[] of same size N. We need to first rearrange the arrays such that the sum of the product of pairs(1 element from each) is minimum. That is $\text{SUM } (A[i] * B[i])$ for all i is minimum.

For example:

Input	Result
3	28
1	
2	
3	
4	
5	
6	

Answer: (penalty regime: 0 %)

```

1 #include <stdio.h>
2 #include <stdlib.h>
3
4 int asc(const void *a, const void *b) {
5     return (*(int *)a - *(int *)b);
6 }
7
8 int desc(const void *a, const void *b) {
9     return (*(int *)b - *(int *)a);
10}
11
12 int main() {
13     int n;
14     scanf("%d", &n);
15     int A[n], B[n];
16     for (int i = 0; i < n; i++) {
17         scanf("%d", &A[i]);
18     }
19     for (int i = 0; i < n; i++) {
20         scanf("%d", &B[i]);
21     }
22
23     qsort(A, n, sizeof(int), asc);
24     qsort(B, n, sizeof(int), desc);
25
26     long long result = 0;
27     for (int i = 0; i < n; i++) {
28         result += (long long)A[i] * B[i];
29     }
30
31     printf("%lld\n", result);
32     return 0;
33 }
34

```

	Input	Expected	Got	
✓	3 1 2 3 4 5 6	28	28	✓
✓	4 7 5 1 2 1 3 4 1	22	22	✓
✓	5 20 10 30 10 40 8 9 4 3 10	590	590	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

[Back to Course](#)



ARIFA H 2024-CSE

A2

Started on Monday, 22 September 2025, 7:29 PM

State Finished

Completed on Monday, 22 September 2025, 7:45 PM

Time taken 15 mins 49 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 %)

```

1 #include <stdio.h>
2
3 int countZeroes(int arr[], int low, int high, int size) {
4     if (high >= low) {
5         int mid = low + (high - low) / 2;
6         if ((mid == 0 || arr[mid - 1] == 1) && arr[mid] == 0)
7             return size - mid;
8         if (arr[mid] == 1)
9             return countZeroes(arr, mid + 1, high, size);
10        else
11            return countZeroes(arr, low, mid - 1, size);
12    }
13    return 0;
14 }
15
16 int main() {
17     int m;
18     scanf("%d", &m);
19     int arr[m];
20     for (int i = 0; i < m; i++) {
21         scanf("%d", &arr[i]);
22     }
23     int result = countZeroes(arr, 0, m - 1, m);
24     printf("%d\n", result);
25     return 0;
26 }
27

```

	Input	Expected	Got	
✓	5 1 1 1 0 0	2	2	✓

	Input	Expected	Got	
✓	10 1 1 1 1 1 1 1 1 1	0	0	✓
✓	8 0 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.

[Back to Course](#)



ARIFA H 2024-CSE

A2

Started on Wednesday, 8 October 2025, 3:49 PM

State Finished

Completed on Wednesday, 8 October 2025, 3:54 PM

Time taken 4 mins 23 secs

Grade 10.00 out of 10.00 (100%)

Question 1 | Correct Mark 10.00 out of 10.00**Playing with Numbers:**

Ram and Sita are playing with numbers by giving puzzles to each other. Now it was Ram term, so he gave Sita a positive integer 'n' and two numbers 1 and 3. He asked her to find the possible ways by which the number n can be represented using 1 and 3. Write any efficient algorithm to find the possible ways.

Example 1:***Input:*** 6***Output:*** 6***Explanation:*** There are 6 ways to represent number with 1 and 3

1+1+1+1+1+1

3+3

1+1+1+3

1+1+3+1

1+3+1+1

3+1+1+1

Input Format

First Line contains the number n

Output Format**Print:** The number of possible ways 'n' can be represented using 1 and 3**Sample Input**

6

Sample Output

6

Answer: (penalty regime: 0 %)

```

1 #include <stdio.h>
2
3 int main() {
4     int n;
5     scanf("%d", &n);
6     long long dp[n + 1];
7     dp[0] = 1;
8     for (int i = 1; i <= n; i++) {
9         dp[i] = dp[i - 1];
10        if (i >= 3) dp[i] += dp[i - 3];
11    }
12    printf("%lld", dp[n]);
13    return 0;
14 }
15

```

	Input	Expected	Got	
✓	6	6	6	✓
✓	25	8641	8641	✓
✓	100	24382819596721629	24382819596721629	✓

Passed all tests! ✓

Correct

Marks for this submission: 10.00/10.00.

[Back to Course](#)



ARIFA H 2024-CSE

A2

Started on Wednesday, 8 October 2025, 3:58 PM

State Finished

Completed on Wednesday, 8 October 2025, 4:00 PM

Time taken 1 min 44 secs

Grade 10.00 out of 10.00 (100%)

Question 1 | Correct Mark 10.00 out of 10.00**Playing with Chessboard:**

Ram is given with an $n \times n$ chessboard with each cell with a monetary value. Ram stands at the $(0,0)$, that the position of the top left white rook. He is been given a task to reach the bottom right black rook position $(n-1, n-1)$ constrained that he needs to reach the position by traveling the maximum monetary path under the condition that he can only travel one step right or one step down the board. Help ram to achieve it by providing an efficient DP algorithm.

Example:**Input**

3

1 2 4

2 3 4

8 7 1

Output:

19

Explanation:

Totally there will be 6 paths among that the optimal is

Optimal path value: $1+2+8+7+1=19$

Input Format

First Line contains the integer n

The next n lines contain the $n \times n$ chessboard values

Output Format

Print Maximum monetary value of the path

Answer: (penalty regime: 0 %)

```

1 #include <stdio.h>
2
3 int main() {
4     int n;
5     scanf("%d", &n);
6     int a[n][n], dp[n][n];
7     for(int i=0;i<n;i++)
8         for(int j=0;j<n;j++)
9             scanf("%d", &a[i][j]);
10    dp[0][0]=a[0][0];
11    for(int i=1;i<n;i++)
12        dp[i][0]=dp[i-1][0]+a[i][0];
13    for(int j=1;j<n;j++)
14        dp[0][j]=dp[0][j-1]+a[0][j];
15    for(int i=1;i<n;i++)
16        for(int j=1;j<n;j++)
17            dp[i][j]=a[i][j]+(dp[i-1][j]>dp[i][j-1]?dp[i-1][j]:dp[i][j-1]);
18    printf("%d", dp[n-1][n-1]);
19    return 0;
20 }
21

```

	Input	Expected	Got	
✓	3 1 2 4 2 3 4 8 7 1	19	19	✓
✓	3 1 3 1 1 5 1 4 2 1	12	12	✓
✓	4 1 1 3 4 1 5 7 8 2 3 4 6 1 6 9 0	28	28	✓

Passed all tests! ✓

Correct

Marks for this submission: 10.00/10.00.

[Back to Course](#)



ARIFA H 2024-CSE

A2

Started on Wednesday, 8 October 2025, 4:00 PM**State** Finished**Completed on** Wednesday, 8 October 2025, 4:01 PM**Time taken** 42 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 two strings find the length of the common longest subsequence(need not be contiguous) between the two.

Example:

s1: ggtabe

s2: tgatasb

s1	a	g	g	t	a	b	
s2	g	x	t	x	a	y	b

The length is 4

Solving it using Dynamic Programming

For example:

Input	Result
aab	2
azb	

Answer: (penalty regime: 0 %)

```

1 #include <stdio.h>
2 #include <string.h>
3
4 int main() {
5     char s1[100], s2[100];
6     scanf("%s", s1);
7     scanf("%s", s2);
8     int n = strlen(s1), m = strlen(s2);
9     int dp[n+1][m+1];
10    for(int i=0;i<=n;i++)
11        for(int j=0;j<=m;j++) {
12            if(i==0 || j==0)
13                dp[i][j]=0;
14            else if(s1[i-1]==s2[j-1])
15                dp[i][j]=dp[i-1][j-1]+1;
16            else
17                dp[i][j]=dp[i-1][j]>dp[i][j-1]?dp[i-1][j]:dp[i][j-1];
18        }
19    printf("%d", dp[n][m]);
20    return 0;
21 }
22

```

	Input	Expected	Got	
✓	aab	2	2	✓
	azb			

	Input	Expected	Got	
✓	ABCD	4	4	✓
	ABCD			

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

[Back to Course](#)



ARIFA H 2024-CSE

A2

Started on Wednesday, 8 October 2025, 4:01 PM**State** Finished**Completed on** Wednesday, 8 October 2025, 4:02 PM**Time taken** 48 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:

Find the length of the Longest Non-decreasing Subsequence in a given Sequence.

Eg:

Input:9

Sequence:[-1,3,4,5,2,2,2,2,3]

the subsequence is [-1,2,2,2,3]

Output:6

Answer: (penalty regime: 0 %)

```

1 #include <stdio.h>
2
3 int main() {
4     int n;
5     scanf("%d", &n);
6     int a[n], dp[n], max = 0;
7     for(int i=0;i<n;i++)
8         scanf("%d", &a[i]);
9     for(int i=0;i<n;i++) {
10         dp[i]=1;
11         for(int j=0;j<i;j++)
12             if(a[i]>=a[j] && dp[i]<dp[j]+1)
13                 dp[i]=dp[j]+1;
14         if(dp[i]>max)
15             max=dp[i];
16     }
17     printf("%d", max);
18     return 0;
19 }
20

```

	Input	Expected	Got	
✓	9 -1 3 4 5 2 2 2 2 3	6	6	✓
✓	7 1 2 2 4 5 7 6	6	6	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

[Back to Course](#)



ARIFA H 2024-CSE

A2

Started on Wednesday, 8 October 2025, 4:02 PM**State** Finished**Completed on** Wednesday, 8 October 2025, 4:03 PM**Time taken** 49 secs**Marks** 1.00/1.00**Grade** **4.00** out of 4.00 (100%)

Question 1 | Correct Mark 1.00 out of 1.00

Find Duplicate in Array.

Given a read only array of n integers between 1 and n, find one number that repeats.

Input Format:

First Line - Number of elements

n Lines - n Elements

Output Format:

Element x - That is repeated

For example:

Input	Result
5	1
1 1 2 3 4	

Answer: (penalty regime: 0 %)

```

1 #include <stdio.h>
2
3 int main() {
4     int n;
5     scanf("%d", &n);
6     int a[n], count[n+1];
7     for(int i=0;i<=n;i++)
8         count[i]=0;
9     for(int i=0;i<n;i++) {
10        scanf("%d", &a[i]);
11        if(count[a[i]]==1) {
12            printf("%d", a[i]);
13            return 0;
14        }
15        count[a[i]]++;
16    }
17    return 0;
18 }
```

	Input	Expected	Got	
✓	11 10 9 7 6 5 1 2 3 8 4 7	7	7	✓
✓	5 1 2 3 4 4	4	4	✓
✓	5 1 1 2 3 4	1	1	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

[Back to Course](#)



ARIFA H 2024-CSE

A2

Started on Wednesday, 8 October 2025, 4:03 PM

State Finished

Completed on Wednesday, 8 October 2025, 4:09 PM

Time taken 5 mins 57 secs

Marks 1.00/1.00

Grade **4.00** out of 4.00 (100%)

Question 1 | Correct Mark 1.00 out of 1.00

Find Duplicate in Array.

Given a read only array of n integers between 1 and n, find one number that repeats.

Input Format:

First Line - Number of elements

n Lines - n Elements

Output Format:

Element x - That is repeated

For example:

Input	Result
5	1
1 1 2 3 4	

Answer: (penalty regime: 0 %)

```

1 #include <stdio.h>
2 #include <stdlib.h>
3
4 int main() {
5     int n, i;
6     scanf("%d", &n);
7     int arr[n];
8     for(i = 0; i < n; i++)
9         scanf("%d", &arr[i]);
10    for(i = 0; i < n; i++) {
11        int index = abs(arr[i]) - 1;
12        if(arr[index] < 0) {
13            printf("%d", abs(arr[i]));
14            return 0;
15        }
16        arr[index] = -arr[index];
17    }
18    return 0;
19 }
20

```

	Input	Expected	Got	
✓	11 10 9 7 6 5 1 2 3 8 4 7	7	7	✓
✓	5 1 2 3 4 4	4	4	✓
✓	5 1 1 2 3 4	1	1	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

[Back to Course](#)



ARIFA H 2024-CSE

A2

Started on Wednesday, 8 October 2025, 4:09 PM**State** Finished**Completed on** Wednesday, 8 October 2025, 4:11 PM**Time taken** 1 min 36 secs**Marks** 1.00/1.00**Grade** **30.00** out of 30.00 (**100%**)

Question 1 | Correct Mark 1.00 out of 1.00

Find the intersection of two sorted arrays.

OR in other words,

Given 2 sorted arrays, find all the elements which occur in both the arrays.

Input Format

- The first line contains T, the number of test cases. Following T lines contain:
 1. Line 1 contains N1, followed by N1 integers of the first array
 2. Line 2 contains N2, followed by N2 integers of the second array

Output Format

The intersection of the arrays in a single line

Example

Input:

1

3 10 17 57

6 2 7 10 15 57 246

Output:

10 57

Input:

1

6 1 2 3 4 5 6

2 1 6

Output:

1 6

For example:

Input	Result
1	10 57
3 10 17 57	
6	
2 7 10 15 57 246	

Answer: (penalty regime: 0 %)

```

1 #include <stdio.h>
2
3 int main() {
4     int T;
5     scanf("%d", &T);
6     while (T--) {
7         int n1, n2, i, j;
8         scanf("%d", &n1);
9         int a[n1];
10        for (i = 0; i < n1; i++)
11            scanf("%d", &a[i]);
12        scanf("%d", &n2);
13        int b[n2];
14        for (i = 0; i < n2; i++)
15            .....;
```

```

15     scanf("%d", &b[1]);
16     i = 0;
17     j = 0;
18     while (i < n1 && j < n2) {
19         if (a[i] == b[j]) {
20             printf("%d ", a[i]);
21             i++;
22             j++;
23         } else if (a[i] < b[j]) {
24             i++;
25         } else {
26             j++;
27         }
28     }
29     printf("\n");
30 }
31     return 0;
32 }
33

```

	Input	Expected	Got	
✓	1 3 10 17 57 6 2 7 10 15 57 246	10 57	10 57	✓
✓	1 6 1 2 3 4 5 6 2 1 6	1 6	1 6	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

[Back to Course](#)



ARIFA H 2024-CSE

A2

Started on Wednesday, 8 October 2025, 4:11 PM

State Finished

Completed on Wednesday, 8 October 2025, 4:12 PM

Time taken 1 min 25 secs

Marks 1.00/1.00

Grade **30.00** out of 30.00 (**100%**)

Question 1 | Correct Mark 1.00 out of 1.00

Find the intersection of two sorted arrays.

OR in other words,

Given 2 sorted arrays, find all the elements which occur in both the arrays.

Input Format

- The first line contains T, the number of test cases. Following T lines contain:
 - Line 1 contains N1, followed by N1 integers of the first array
 - Line 2 contains N2, followed by N2 integers of the second array

Output Format

The intersection of the arrays in a single line

Example

Input:

1

3 10 17 57

6 2 7 10 15 57 246

Output:

10 57

Input:

1

6 1 2 3 4 5 6

2 1 6

Output:

1 6

For example:

Input	Result
1	10 57
3 10 17 57	
6	
2 7 10 15 57 246	

Answer: (penalty regime: 0 %)

```

1 #include <stdio.h>
2
3 int main() {
4     int T;
5     scanf("%d", &T);
6     while (T--) {
7         int n1, n2, i, j;
8         scanf("%d", &n1);
9         int a[n1];
10        for (i = 0; i < n1; i++)
11            scanf("%d", &a[i]);
12        scanf("%d", &n2);
13        int b[n2];
14        for (i = 0; i < n2; i++)
15            .....;
```

```

15     scanf("%d", &b[1]);
16     i = 0;
17     j = 0;
18     while (i < n1 && j < n2) {
19         if (a[i] == b[j]) {
20             printf("%d ", a[i]);
21             i++;
22             j++;
23         } else if (a[i] < b[j]) {
24             i++;
25         } else {
26             j++;
27         }
28     }
29     printf("\n");
30 }
31     return 0;
32 }
33

```

	Input	Expected	Got	
✓	1 3 10 17 57 6 2 7 10 15 57 246	10 57	10 57	✓
✓	1 6 1 2 3 4 5 6 2 1 6	1 6	1 6	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

[Back to Course](#)



ARIFA H 2024-CSE

A2

Started on Wednesday, 8 October 2025, 4:13 PM**State** Finished**Completed on** Wednesday, 8 October 2025, 4:13 PM**Time taken** 41 secs**Marks** 1.00/1.00**Grade** **4.00** out of 4.00 (100%)

Question 1 | Correct Mark 1.00 out of 1.00

Given an array A of sorted integers and another non negative integer k, find if there exists 2 indices i and j such that $A[j] - A[i] = k$, $i \neq j$.

Input Format:

First Line n - Number of elements in an array

Next n Lines - N elements in the array

k - Non - Negative Integer

Output Format:

1 - If pair exists

0 - If no pair exists

Explanation for the given Sample Testcase:

YES as $5 - 1 = 4$

So Return 1.

For example:

Input	Result
3	1
1 3 5	
4	

Answer: (penalty regime: 0 %)

```

1 #include <stdio.h>
2
3 int main() {
4     int n, k;
5     scanf("%d", &n);
6     int arr[n];
7     for(int i = 0; i < n; i++)
8         scanf("%d", &arr[i]);
9     scanf("%d", &k);
10
11     int i = 0, j = 1;
12     while (i < n && j < n) {
13         if (i != j && arr[j] - arr[i] == k) {
14             printf("1\n");
15             return 0;
16         } else if (arr[j] - arr[i] < k) {
17             j++;
18         } else {
19             i++;
20         }
21     }
22     printf("0\n");
23     return 0;
24 }
```

	Input	Expected	Got	
✓	3 1 3 5 4	1	1	✓
✓	10 1 4 6 8 12 14 15 20 21 25 1	1	1	✓
✓	10 1 2 3 5 11 14 16 24 28 29 0	0	0	✓
✓	10 0 2 3 7 13 14 15 20 24 25 10	1	1	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

[Back to Course](#)



ARIFA H 2024-CSE

A2

Started on Wednesday, 8 October 2025, 4:14 PM**State** Finished**Completed on** Wednesday, 8 October 2025, 4:15 PM**Time taken** 1 min 7 secs**Marks** 1.00/1.00**Grade** **4.00** out of 4.00 (100%)

Question 1 | Correct Mark 1.00 out of 1.00

Given an array A of sorted integers and another non negative integer k, find if there exists 2 indices i and j such that $A[j] - A[i] = k$, $i \neq j$.

Input Format:

First Line n - Number of elements in an array

Next n Lines - N elements in the array

k - Non - Negative Integer

Output Format:

1 - If pair exists

0 - If no pair exists

Explanation for the given Sample Testcase:

YES as $5 - 1 = 4$

So Return 1.

For example:

Input	Result
3	1
1 3 5	
4	

Answer: (penalty regime: 0 %)

```

1 #include <stdio.h>
2
3 int main() {
4     int n, k;
5     scanf("%d", &n);
6     int arr[n];
7     for(int i = 0; i < n; i++)
8         scanf("%d", &arr[i]);
9     scanf("%d", &k);
10
11    int i = 0, j = 1;
12    while (i < n && j < n) {
13        int diff = arr[j] - arr[i];
14        if (i != j && diff == k) {
15            printf("1\n");
16            return 0;
17        } else if (diff < k) {
18            j++;
19        } else {
20            i++;
21        }
22    }
23    printf("0\n");
24    return 0;
25}
26

```

	Input	Expected	Got	
✓	3 1 3 5 4	1	1	✓
✓	10 1 4 6 8 12 14 15 20 21 25 1	1	1	✓
✓	10 1 2 3 5 11 14 16 24 28 29 0	0	0	✓
✓	10 0 2 3 7 13 14 15 20 24 25 10	1	1	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

[Back to Course](#)