Rules for submission

- Do **not cheat or use ready-made solutions.** If any attempt of cheating is discovered, **legal actions will be taken.**
- You have **10 days** to solve the problem.
- You **must present the solution to the professor** during the days from **December 20th to 21st**.
- ***2*** **marks will be deducted** if you fail to present the solution within the given time frame.

Problem Statement: Cumulative Sum for Multiple Queries

Problem Description:

You are given an array of integers arr[] of size *n*. You need to answer multiple range sum queries. For each query, you will be asked to return the sum of elements in the subarray from index l to index r (both inclusive). You need to process these queries efficiently.

Input:

- An array arr[] of integers with size *n*.
- An integer *q* representing the number of queries.
- For each query, you are given two integers *l* and *r*, where you need to return the sum of elements in the subarray arr[l...r].

Output:

- For each query, print the sum of elements from index l to r (inclusive).

Example Test Cases:

Example 1:

Input:

arr = [1, 2, 3, 4, 5]
Number of Queries: 3
0 2
1 4
0 4

Output:

6
14
15

Explanation:

- Query 0 2: Sum of elements from index 0 to 2: 1 + 2 + 3 = 6
- Query 1 4: Sum of elements from index 1 to 4: 2 + 3 + 4 + 5 = 14
- Query 0 4: Sum of elements from index 0 to 4: 1 + 2 + 3 + 4 + 5 = 15

Example 2:

Input:

```
arr = [10, 20, 30, 40, 50]
Number of Queries: 3
0 0
2 4
1 3
```

Output:

```
10
120
90
```

Explanation:

- Query 0 0: Sum of elements from index 0 to 0: 10
- Query 2 4: Sum of elements from index 2 to 4: 30 + 40 + 50 = 120
- Query 1 3: Sum of elements from index 1 to 3: 20 + 30 + 40 = 90

Example 3:

Input:

```
arr = [5, 8, 12, 6]
Enter Number of Queries: 2
1 3
0 3
```

Output:

```
20
31
```

Explanation:

- Query 1 3: Sum of elements from index 1 to 3: 8 + 12 + 6 = 20
- Query 0 3: Sum of elements from index 0 to 3: 5 + 8 + 12 + 6 = 31