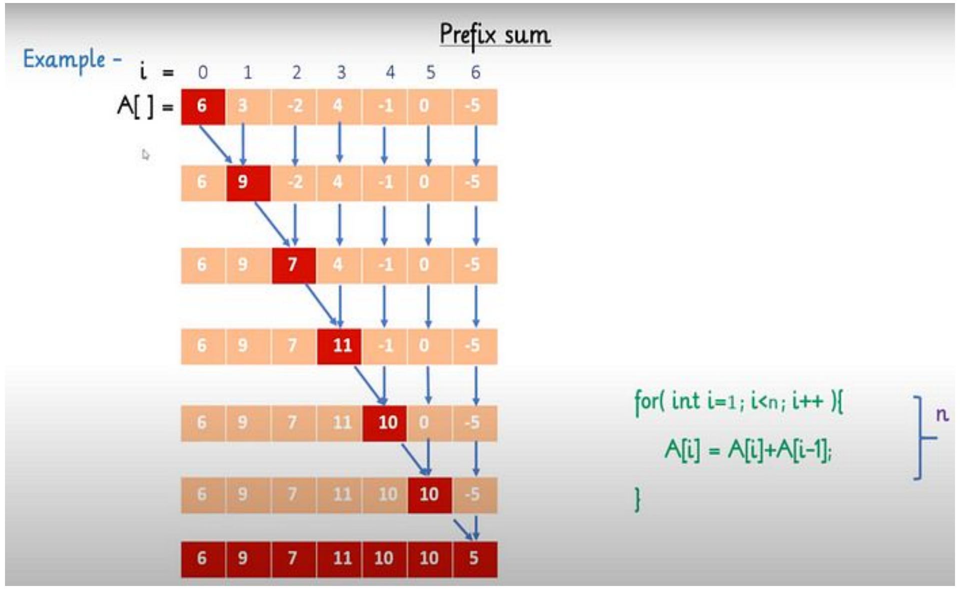
Equilibrium Index

Session No.: 2
Course Name: Advanced Algorithmic Problem Solving
Course Code: R1UC601B
Instructor Name: Dr. Subhash chandra Gupta
Duration: 50 mins
Date of Conduction of Class: 13 Feb 2026

Review

Prefix Sum Array: Find the sum of elements in a given range [L, R] using a prefix sum array.

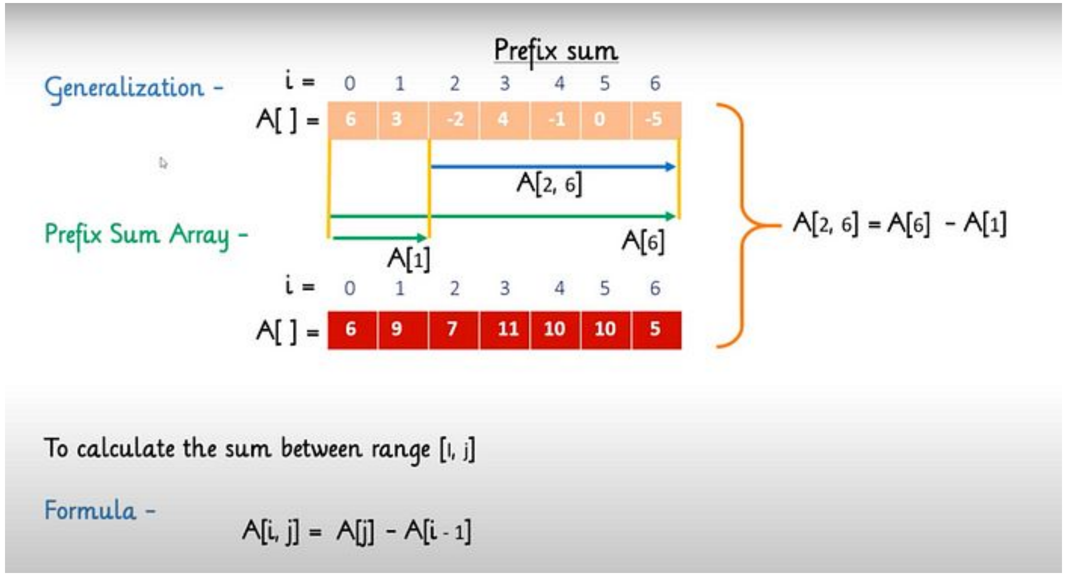


Prefix sum

Example -

i =	0	1	2	3	4	5	6
A[ ] =	6	3	-2	4	-1	0	-5
	6	9	-2	4	-1	0	-5
	6	9	7	4	-1	0	-5
	6	9	7	11	-1	0	-5
	6	9	7	11	10	0	-5
	6	9	7	11	10	10	-5
	6	9	7	11	10	10	5

```
for( int i=1; i<n; i++ ){
    A[i] = A[i]+A[i-1];
}
```



Prefix sum

Generalization -

i =	0	1	2	3	4	5	6
A[ ] =	6	3	-2	4	-1	0	-5

Prefix Sum Array -

i =	0	1	2	3	4	5	6
A[ ] =	6	9	7	11	10	10	5

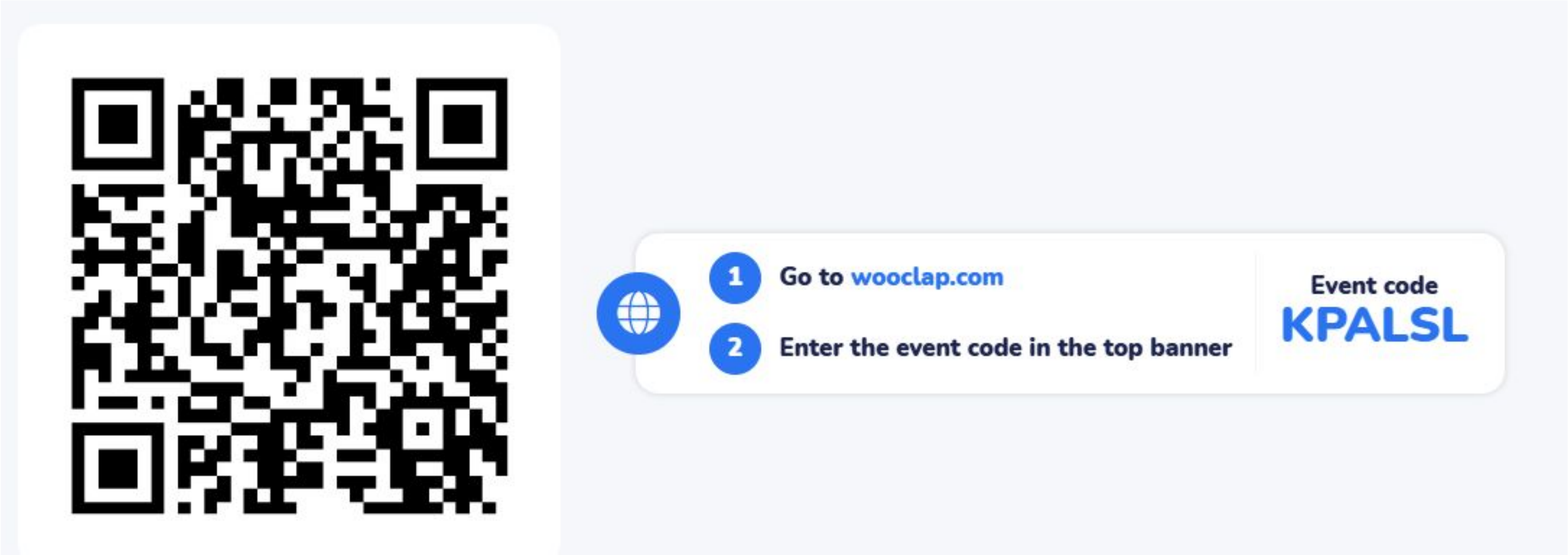
A[2, 6] = A[6] - A[1]

To calculate the sum between range [i, j]

Formula - $A[i, j] = A[j] - A[i-1]$

Pre-Class Assessment

[2-mins]



1 Go to wooclap.com

2 Enter the event code in the top banner

Event code

KPALSL

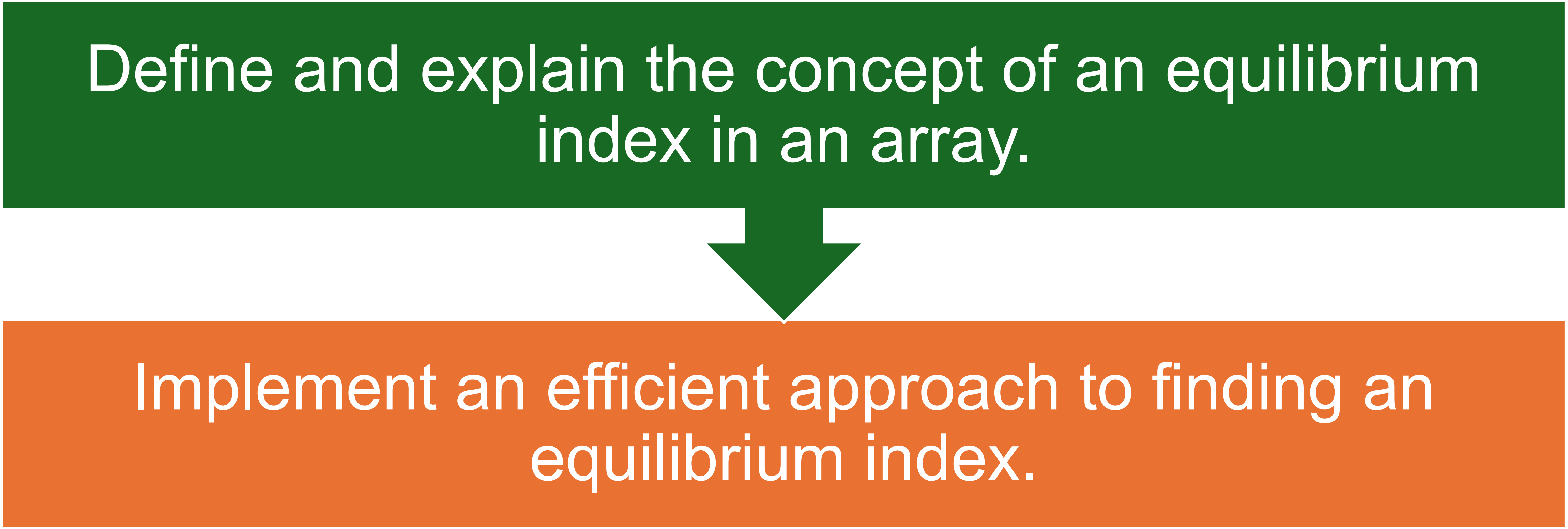
[1-mins]

How to find an index in an array such that the sum of elements to the left is equal to the sum of elements to the right.

Learning Outcomes

[2-mins]

By the end of this session, You will be able to:



Define and explain the concept of an equilibrium index in an array.

Implement an efficient approach to finding an equilibrium index.

Session Outline

- Introduce students to the concept of the equilibrium index in an array.
- Demonstrate efficient approaches to finding the equilibrium index.
- Provide hands-on coding practice and reinforce learning through an interactive Wooclap quiz or poll.

Activity-1 (Think – Pair – Share)

[1-mins]

How to find an index in an array such that the sum of elements to the left is equal to the sum of elements to the right.

For the array [1, 3, 5, 2, 2], the equilibrium index is 2 (since 1+3 = 2+2).

Concept and Definition for (LO-1) Equilibrium index

- "An index i is an equilibrium index if the sum of elements to its left equals the sum of elements to its right."
- Give an example: For the array [1, 3, 5, 2, 2], the equilibrium index is 2 (since 1+3 = 2+2).
- Compare brute force ($O(n^2)$) vs. optimized prefix sum approach ($O(n)$).

[5-mins]

Using Two Loops (Brute Force Approach)

- Consider two variables leftSum and rightSum initialized to zero. Now for every index, i calculate the leftSum till that index and rightSum till that index.
- At any point if (leftSum == rightSum) return that index i.
- If equilibrium index not found return -1.

[5-mins]

```
int findEquilibriumIdx(int nums[], int n) {
    int leftSum, rightSum;
    for (int i = 0; i < n; i++)
    {
        leftSum = 0;
        for (int j = 0; j < i; j++) {                    leftSum
+= nums[j];
        }
        rightSum = 0;
        for (int j = i + 1; j < n; j++) {
            rightSum += nums[j];
        }
```

```
    if (leftSum == rightSum) {
        return i;
    }
    }
    return -1;
}
```

Time Complexity : $O(n^2)$

Optimized Approach:

- **Compute the prefix sum array** so that sum calculations are done efficiently.
- **Iterate through the array** and for each index:
  - Compute leftSum using the prefix sum array.
  - Compute rightSum using the prefix sum of last index minus the prefix sum of the current index.
  - If leftSum == rightSum, it's an equilibrium index.
- **Store and print all equilibrium indices**.

Activity 2: Coding Problem (Pen-Paper)

Problem Statement:

Given an array, find and print all equilibrium indices.

Example Input: [2, 3, -1, 8, 4]

Example Output: Index 3 is an equilibrium index.

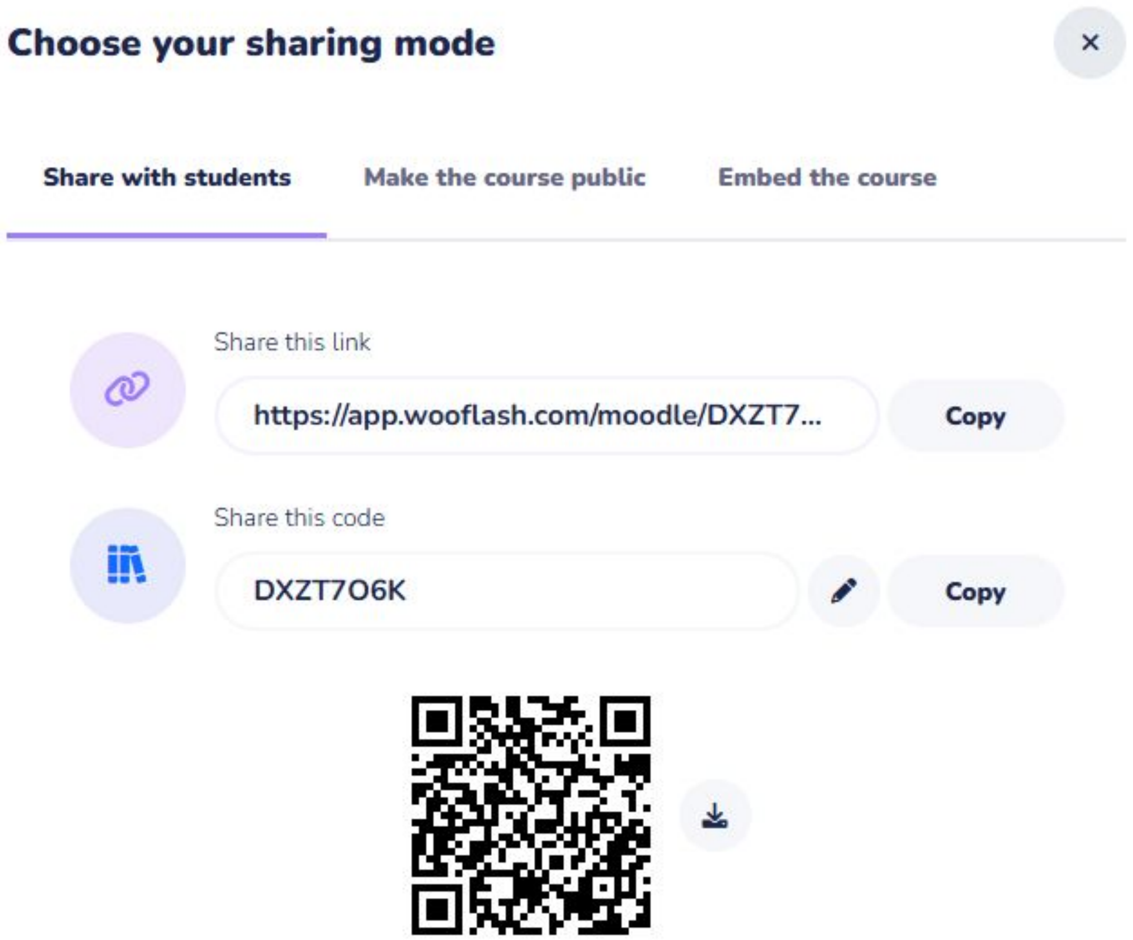
```
int findEquilibriumIndicesPrefixSum(int arr[], int n) {
    int prefixSum[n];
    prefixSum[0] = arr[0];
     for (int i = 1; i < n; i++) {
            prefixSum[i] = prefixSum[i - 1] + arr[i];
     }
    for (int i = 0; i < n; i++) {
            int leftSum = (i == 0) ? 0 : prefixSum[i - 1];
            int rightSum = prefixSum[n - 1] - prefixSum[i];

            if (leftSum == rightSum)
            return i;
    }
    return -1;
}
```

Time Complexity

- Computing the prefix sum: **O(n)**
- Checking for equilibrium indices: **O(n)**
- **Overall Complexity: O(n)**

Assessment: WooFlash Quiz



Choose your sharing mode

Share this link

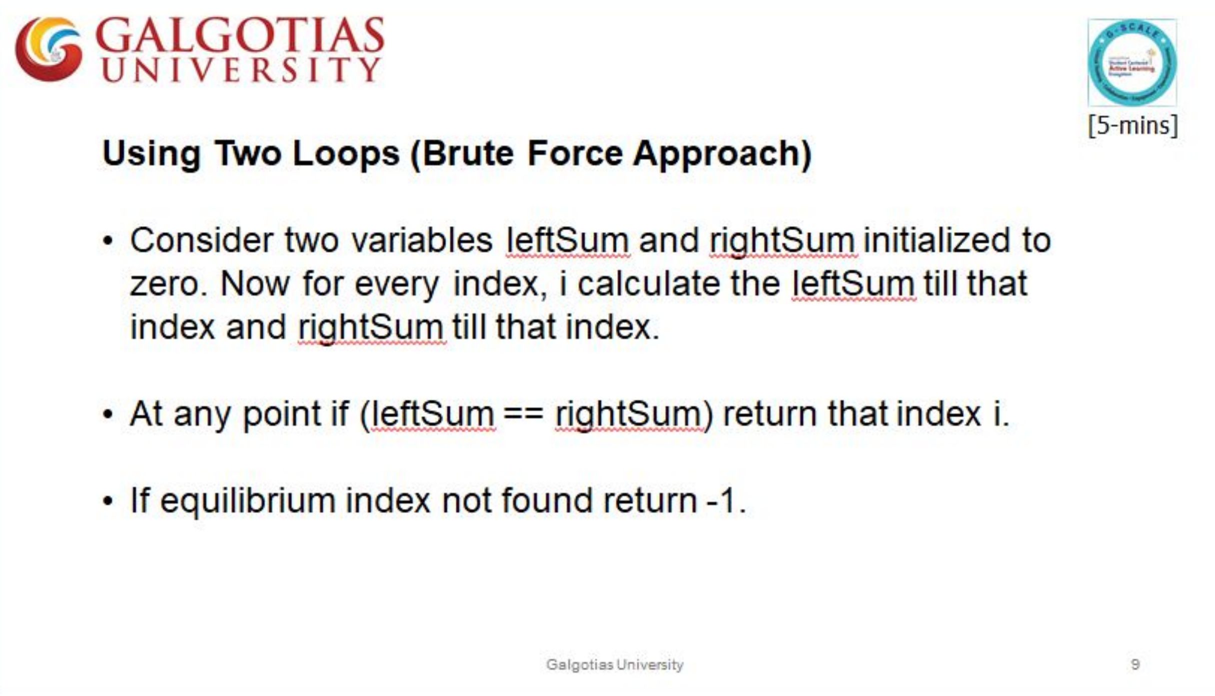
https://app.wooflash.com/moodle/DXZT7...

Share this code

DXZT7O6K

Summary

[1-mins]

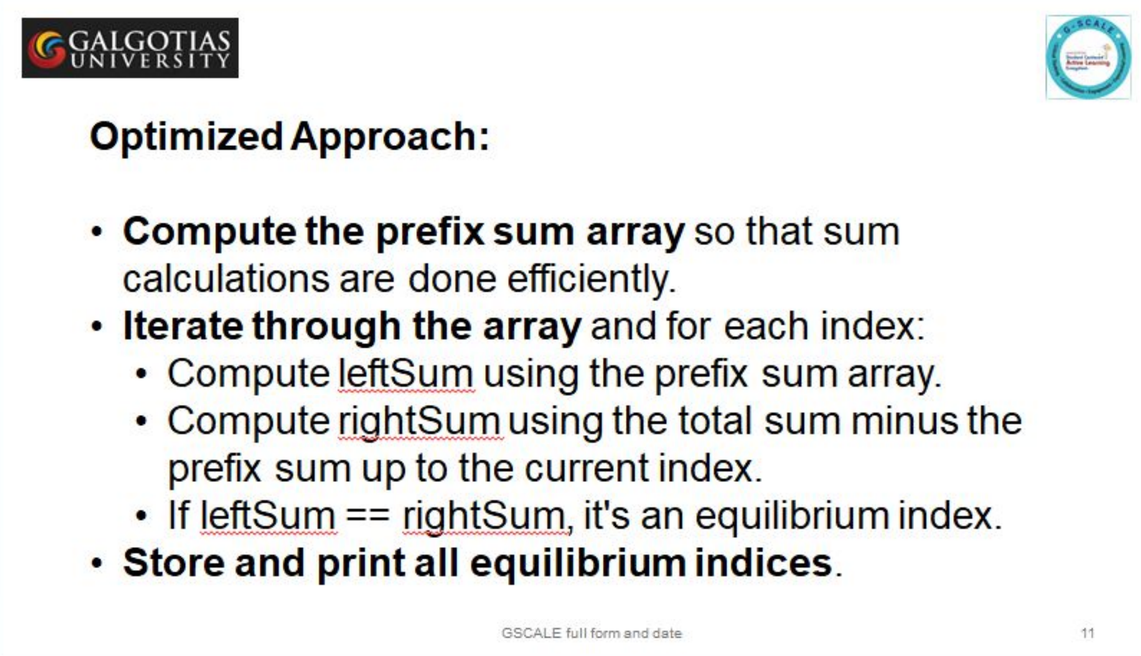


[5-mins]

Using Two Loops (Brute Force Approach)

- Consider two variables leftSum and rightSum initialized to zero. Now for every index, i calculate the leftSum till that index and rightSum till that index.
- At any point if (leftSum == rightSum) return that index i.
- If equilibrium index not found return -1.

Galgotias University 9



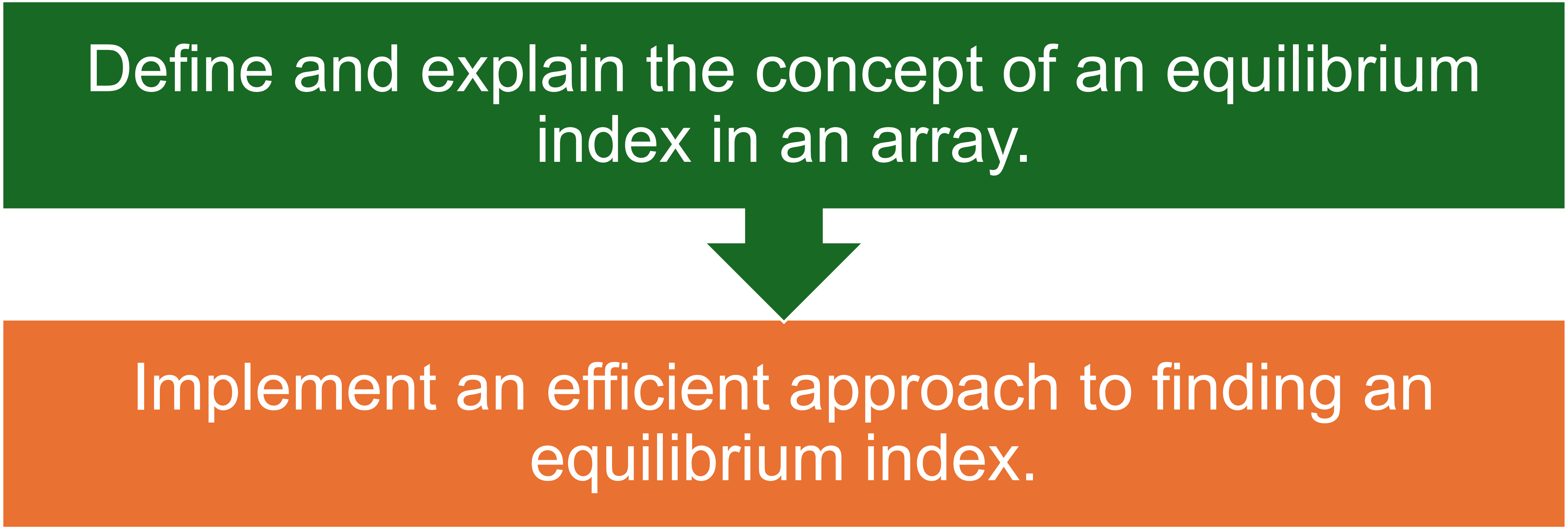
Optimized Approach:

- **Compute the prefix sum array** so that sum calculations are done efficiently.
- **Iterate through the array** and for each index:
  - Compute leftSum using the prefix sum array.
  - Compute rightSum using the total sum minus the prefix sum up to the current index.
  - If leftSum == rightSum, it's an equilibrium index.
- **Store and print all equilibrium indices**.

GSCALE full form and date 11

Ensure attainment of LO’s in alignment to the learning activities:

[2 mins]



Define and explain the concept of an equilibrium index in an array.

Implement an efficient approach to finding an equilibrium index.

Discussion on the post session activities

Key points:

1. **Brute-Force Approach**:
   - Loops through each index, computing the left and right sums separately.
   - Time Complexity: **O(n^2)**
   - Space Complexity: **O(1)**
2. **Prefix Sum Approach**:
   - First computes a prefix sum array to store cumulative sums.
   - Uses this array to efficiently calculate left and right sums.
   - Time Complexity: **O(n)**
   - Space Complexity: **O(n)**

Homework Problem:

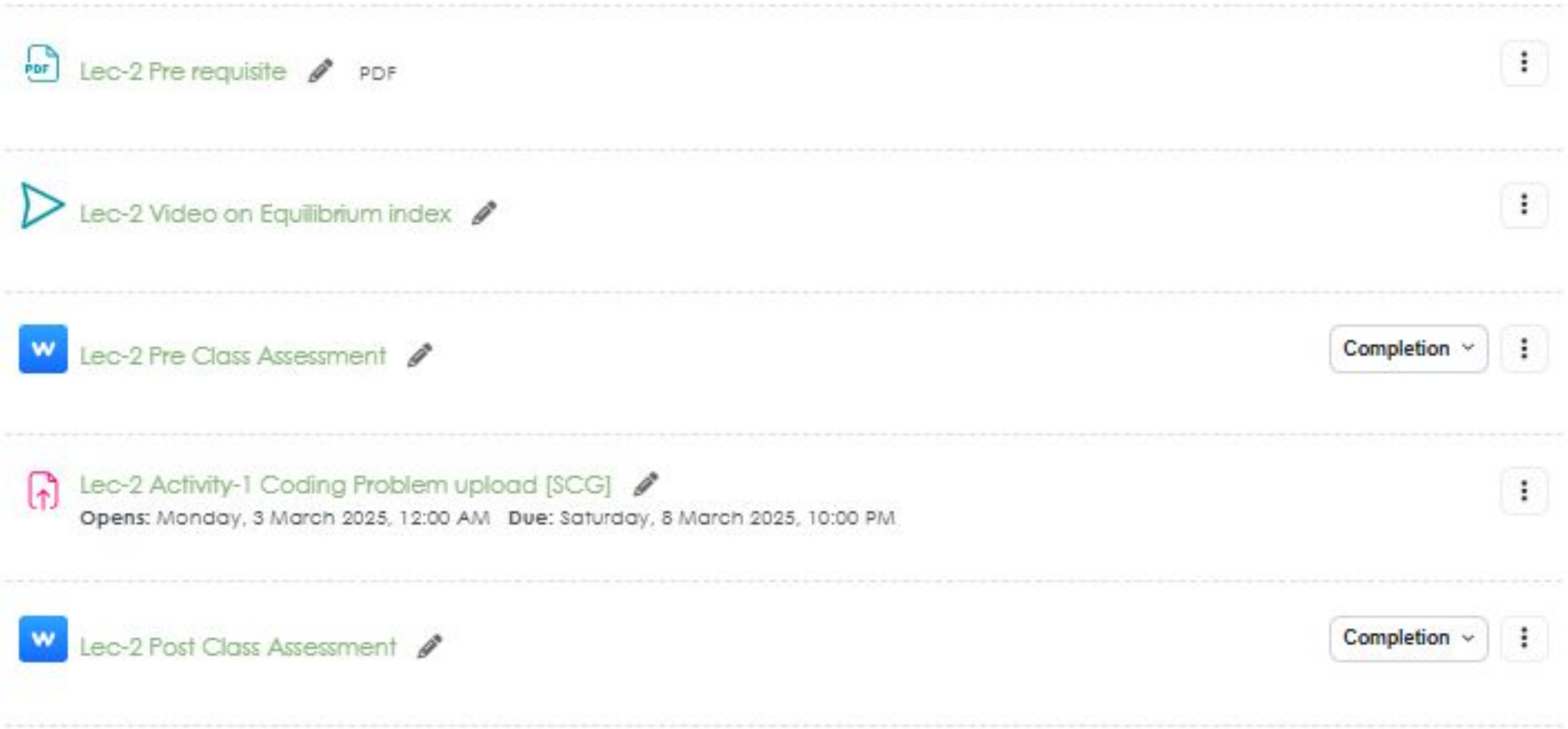
"Given an array [4, -1, 0, 3, -2, 2], find all equilibrium indices."

[1-min]

Next Session:

Split Array into Equal Sum Prefix and Suffix: Check if an array can be split into two parts such that the sum of the prefix equals the sum of the suffix.

Review and Reflection from students



Lec-2 Pre requisite

Lec-2 Video on Equilibrium index

Lec-2 Pre Class Assessment

Lec-2 Activity-1 Coding Problem upload [SCG]

Opens: Monday, 3 March 2025, 12:00 AM **Due:** Saturday, 8 March 2025, 10:00 PM

Lec-2 Post Class Assessment