

Sliding Window Technique

Problem

You are given an array $a[]$ with n elements. Your task is to calculate the minimum sum of k consecutive elements.

Example

-2	10	1	3	2	-1	4	5
----	----	---	---	---	----	---	---

$K = 3$

Sums possible of 3 consecutive elements: 9, 14, 6, 4, 5, 8.

Minimum of them = 4 (answer).

Brute force approach

1. Iterate from $i=0$ to $i=n-k-1$ in the outer loop.
2. Starting from $j=i$, compute the sum of k elements and maintain the minimum.

Time complexity: $O(n*k)$

Optimal Approach (Sliding Window Technique)

1. Compute sum of first k elements by iterating from $i=0$ to $i=k-1$ and store it in the variable sum .
2. While increasing i , subtract $a[i-1]$ and add $a[i+k-1]$ in the previous sum, which will become the current sum. This step is known as moving window forward.
3. Keep a variable mn , which stores the minimum of the sum. Operation used is

$$mn = \min(mn, sum)$$

Time Complexity: $O(n)$

Code

```
void solve()
{
    int n,k;
    cin >> n >> k;

    vector<int> a(n);

    for(int i=0; i<n; i++)
        cin >> a[i];

    int sum=0;
    for(int i=0; i<k; i++)
        sum += a[i];

    int i=0, mn = INT_MAX;
    mn = min(mn,sum);
    while(i+k<n)
    {
        sum += a[i+k];
        sum -= a[i];
        mn = min(sum, mn);
        i++;
    }

    cout << mn << endl;
}
```