GeeksforGeeks
A computer science portal for geeks
Practice IDE Q&A GeeksQuiz

Find the subarray with least average

Given an array arr[] of size n and integer k such that k <= n.

```
Input:  arr[] = {3, 7, 90, 20, 10, 50, 40}, k = 3
Output: Subarray between indexes 3 and 5
The subarray {20, 10, 50} has the least average
among all subarrays of size 3.

Input:  arr[] = {3, 7, 5, 20, -10, 0, 12}, k = 2
Output: Subarray between [4, 5] has minimum average
```

We strongly recommend you to minimize your browser and try this yourself first.

A **Simple Solution** is to consider every element as beginning of subarray of size k and compute sum of subarray starting with this element. Time complexity of this solution is O(nk).

An **Efficient Solution** is to solve the above problem in O(n) time and O(1) extra space. The idea is to use sliding window of size k. Keep track of sum of current k elements. To compute sum of current window, remove first element of previous window and add current element (last element of current window).

```
1) Initialize res_index = 0 // Beginning of result index
2) Find sum of first k elements. Let this sum be 'curr_sum'
3) Initialize min_sum = sum
4) Iterate from (k+1)'th to n'th element, do following
   for every element arr[i]
      a) curr_sum = curr_sum + arr[i] - arr[i-k]
      b) If curr_sum < min_sum
           res_index = (i-k+1)
5) Print res_index and res_index+k-1 as beginning and ending
   indexes of resultant subarray.
```

Below is C++ implementation of above algorithm.

```
// A Simple C++ program to find minimum average subarray
#include<bits/stdc++.h>
using namespace std;

// Prints beginning and ending indexes of subarray
// of size k with minimum average
void findMinAvgSubarray(int arr[], int n, int k)
{
    // k must be smaller than or equal to n
    if (n < k)
       return;

    // Initialize  beginning index of result
    int res_index = 0;

    // Compute sum of first subarray of size k
    int curr_sum = 0;
    for (int i=0; i<k; i++)
       curr_sum += arr[i];

    // Initialize minimum sum as current sum
    int min_sum = curr_sum;

    // Traverse from (k+1)'th element to n'th element
    for (int i = k; i < n; i++)
    {
        // Add current item and remove first item of
        // previous subarray
        curr_sum += arr[i] - arr[i-k];

        // Update result if needed
        if (curr_sum < min_sum)
        {
            min_sum = curr_sum;
            res_index = (i-k+1);
        }
    }

    cout << "Subarray between [" << res_index << ", "
        << res_index + k - 1<< "] has minimum average";
}

// Driver program
int main()
{
    int arr[] = {3, 7, 90, 20, 10, 50, 40};
    int k = 3; // Subarray size
    int n = sizeof arr / sizeof arr[0];
    findMinAvgSubarray(arr, n, k);
    return 0;
}
```

Output:

```
Subarray between [3, 5] has minimum average
```

Time Complexity: O(n)
Auxiliary Space: O(1)

Source: http://qa.geeksforgeeks.org/2221/given-an-array-integers-find-subarray-having-least-average

Please write comments if you find anything incorrect, or you want to share more information about the topic discussed above

8 Comments Category: Arrays

Related Posts:

- Longest Span with same Sum in two Binary arrays
- Count Inversions of size three in a give array
- Count triplets with sum smaller than a given value
- Find zeroes to be flipped so that number of consecutive 1's is maximized
- Reorder an array according to given indexes
- Find maximum value of Sum(i*arr[i]) with only rotations on given array allowed
- Find maximum average subarray of k length
- Convert array into Zig-Zag fashion

(Login to Rate and Mark)

1.8 Average Difficulty : **1.8/5.0**
Based on **15** vote(s)

□ Add to TODO List

□ Mark as DONE

Like Share 26 people like this. Be the first of your friends.

Writing code in comment? Please use code.geeksforgeeks.org, generate link and share the link here.

@geeksforgeeks, Some rights reserved Contact Us! About Us! Advertise with us!