GeeksforGeeks

A computer science portal for geeks

Practice IDE Q&A GeeksQuiz

Length of the largest subarray with contiguous elements | Set 1

Given an array of distinct integers, find length of the longest subarray which contains numbers that can be arranged in a continuous sequence.

Examples:

```
Input:  arr[] = {10, 12, 11};
Output: Length of the longest contiguous subarray is 3

Input:  arr[] = {14, 12, 11, 20};
Output: Length of the longest contiguous subarray is 2

Input:  arr[] = {1, 56, 58, 57, 90, 92, 94, 93, 91, 45};
Output: Length of the longest contiguous subarray is 5
```

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

The important thing to note in question is, it is given that all elements are distinct. If all elements are distinct, then a subarray has contiguous elements if and only if the difference between maximum and minimum elements in subarray is equal to the difference between last and first indexes of subarray. So the idea is to keep track of minimum and maximum element in every subarray.

The following is C++ implementation of above idea.

```cpp
#include<iostream>
using namespace std;

// Utility functions to find minimum and maximum of
// two elements
int min(int x, int y) { return (x < y)? x : y; }
int max(int x, int y) { return (x > y)? x : y; }

// Returns length of the longest contiguous subarray
int findLength(int arr[], int n)
{
    int max_len = 1;  // Initialize result
    for (int i=0; i<n-1; i++)
    {
        // Initialize min and max for all subarrays starting with i
        int mn = arr[i], mx = arr[i];

        // Consider all subarrays starting with i and ending with j
        for (int j=i+1; j<n; j++)
        {
            // Update min and max in this subarray if needed
            mn = min(mn, arr[j]);
            mx = max(mx, arr[j]);

            // If current subarray has all contiguous elements
            if ((mx - mn) == j-i)
                max_len = max(max_len, mx-mn+1);
        }
    }
    return max_len;  // Return result
}

// Driver program to test above function
int main()
{
    int arr[] = {1, 56, 58, 57, 90, 92, 94, 93, 91, 45};
    int n = sizeof(arr)/sizeof(arr[0]);
    cout << "Length of the longest contiguous subarray is "
         << findLength(arr, n);
    return 0;
}
```

Run on IDE

Output:

```
Length of the longest contiguous subarray is 5
```

Time Complexity of the above solution is O(n^2).

We will soon be covering solution for the problem where duplicate elements are allowed in subarray.

This article is contributed by **Arjun**. Please write comments if you find anything incorrect, or you want to share more information about the topic discussed above

57 Comments Category: Arrays

Related Posts:

- Longest Span with same Sum in two Binary arrays
- Count Inversions of size three in a give array
- Find the subarray with least average
- 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

(Login to Rate and Mark)

2.8 Average Difficulty : **2.8/5.0**
Based on **8** vote(s)

□ Add to TODO List

□ Mark as DONE

Like Share 14 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.