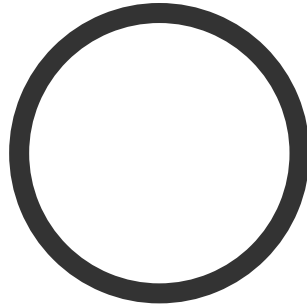


ilovealgorithms

A must look before coding interview

[Home](#) [A Few Lines About the Purpose of this Blog !!](#)



The Two pointer Algorithm !!

Posted on **November 5, 2012** by **neil1422**

A very easy but really effective algorithm..

I will try to describe this method on example of a problem.

Given two arrays (A and B) sorted in ascending order, and an integer x. we need to find i and j, such that $a[i] + b[j]$ is equal to X.

i and j our pointers, at first step i points to the first element of a, and j points to the last element of b.

$i = 0; j = b.size() - 1;$

move first pointer one by one through the array a, and decrease position of the second pointer if needed.

Here we are checking if $a[i] + b[j] = X$. If so, then write the answer. When this is not true, then if $a[i] + b[j] > X$, then decrease j (if $j > 0$), because what we need must be at a position less than current position of j or more than current position of i.

```

while (i < a.size())
{
    while(a[i] + b[j] > X && j > 0) j--;
    if (a[i] + b[j] == X) writeAnswer(i, j);
    i++;
}

```

Now Take the example of this problem.

Consider **problem 190D – Non-Secret Cypher**. The stupid solution goes over all subarrays and checks whether it's good or not. Now we notice that if subarray is 'good', then all its superarrays is 'good' too. Let's go over all left borders x ($1 \leq x \leq N$) and for each of them find rx . rx is a minimal number such that subarray $x... rx$ is good. Obviously, all subarrays which starts at x and ends after rx is good.

If you notice that $r_1 \leq r_2 \leq \dots \leq r_N$, you can use 'two pointers' method. The first pointer is x and the second one is rx . When you move the first one right, the second one can only move right too. No matter how many operations you perform in one step, algo's running time is $O(n)$, because each of pointers makes $\leq N$ steps.

Now, try this problem by your own:-

Q> Given n numbers, find pairs with difference x .

Hint: First sort then apply the algo you have just learnt.

If you still can't understand, then comment below..

Advertisements

Share this:



Be the first to like this.

Related

Max element in current element.
Implementation using the concept of
doubly ended queued but with
simple arrays and two pointers.

Reverse a stack without any other
data structure !!

The sieve of Eratosthenes !!

← A Summary of Time Complexity of Most Common
Sorting Algos !!

The sieve of Eratosthenes !! →

2 thoughts on “The Two pointer Algorithm !!”



neeraj says:

May 26, 2014 at 10:55 AM

How does this method help achieving the better time or space complexity? What I mean is the algorithm implementation is not facilitating the $O(n^2)$ complexity.

Reply



neil1422 says:

May 31, 2014 at 12:36 AM

You use at most 2 pointers. Each of them takes at most $O(n)$ time. So, total time is $O(n + n) = O(n)$.
So this is facilitating the $O(n^2)$ complexity.
Space complexity is not reduced though.

Reply

Leave a Reply

Enter your comment here...

Search ...

Recent Posts

- [Max element in current element. Implementation using the concept of doubly ended queued but with simple arrays and two pointers.](#)
- [Swap the nth and the mth node in a linked list](#)
- [Reverse every K nodes in a linked list.](#)
- [Reverse a linked list in pairs](#)
- [Reverse a linked list](#)

Archives

- [May 2014](#)
- [November 2012](#)

Categories

- [Uncategorized](#)

Meta

- [Register](#)
- [Log in](#)
- [Entries RSS](#)

- [Comments RSS](#)
- [WordPress.com](#)

Blog at WordPress.com.