Yu's Coding Garden

My blog for LeetCode Questions and Answers...

Tuesday, October 28, 2014

leetcode Quesion: Maximum Product Subarray

Maximum Product Subarray

Find the contiguous subarray within an array (containing at least one number) which has the largest product.

For example, given the array [2,3,-2,4],
the contiguous subarray [2,3] has the largest product = 6.

Analysis:
Brutal force method will take O(n^2) time complexity, which is not acceptable.
Since it requires the "largest" product in an array, it is common to consider the DP(dynamic programming) method. And also since O(n^2) is NOT acceptable, there might be an O(n) solution.

So, given a whole scan , how to find the maximum product?
Note that, we need to consider two cases:
(1) negative numbers
        Store the minimum product to handle the case that new element < 0. Because if current element < 0, the product of two negative numbers (new element, and minimum product before the new element) become positive.
(2) zeros
        When meets zero, current max and min product become 0, new search is needed from the next element.

Therefore, we can write down to function to store both + and - products:

*max_product = max{A[i]*min_product (when A[i]<0), A[i]*max_product (when A[i]>0), A[i] (when 0 occurs before A[i]) }.*

*min_product = min{A[i]*min_product, A[i]*max_product, A[i] }.*

Because current sequence might start from any element, to get the final result, we also need to store the max product after each iteration "res = max(res, maxp);".

Code(C++):

```cpp
class Solution {
public:
    int maxProduct(int A[], int n) {
        int res = A[0];
        int maxp = A[0];
        int minp = A[0];
        for (int i=1;i<n;i++){
            int tmpmax = maxp;
            int tmpmin = minp;
            maxp = max(max(tmpmax*A[i],tmpmin*A[i]),A[i]);
            minp = min(min(tmpmax*A[i],tmpmin*A[i]),A[i]);
            res = max(maxp,res);
        }
        return res;
    }
};
```

Code(Python):

```python
class Solution:
    # @param A, a list of integers
    # @return an integer
    def maxProduct(self, A):
        res = A[0]
        minp = A[0]
        maxp = A[0]
        for a in A[1:]:
            tmp_maxp = maxp
            tmp_minp = minp
            maxp = max(max(tmp_maxp*a, tmp_minp*a), a)
            minp = min(min(tmp_maxp*a, tmp_minp*a), a)
            res = max(res, maxp)
        return res
```

Who am I?

I have just got my Ph.D. in Computer Science. Now I am a research engineer at a startup company.

My research focuses on deep learning and computer vision.

My homepage: (1)
My Google Scholar Page: here

New!!!
I am starting to add *Python* Code for the problems!!!

Useful link: C++ reference

Basics:

Common Sorting Algorithms

Hash Table Basics

Tree Traversal (Recursive & Non-Recursive)

Search the leetcode solutions here:

Search

Posted by Yu Zhu at 7:16 AM +2 Recommend this on Google

Labels: array, DP, leetcode

2 comments:

Unknown September 21, 2015 at 11:46 PM

Can you please share your thought process? I mean, how did you arrive at this solution step by step?

Reply

Rajan Kasodariya September 29, 2015 at 6:21 AM

Great

Reply

Enter your comment...

Comment as: Select profile...

Publish Preview

Newer Post Home Older Post

Subscribe to: Post Comments (Atom)

I started to post **C++ Solutions for the famous book:**
Cracking The Coding Interview (5th Edition)
Please Look at my gitHub:
here
(Regularly update)

Note:
All the code provided for the ***leetcode questions*** *have been tested*
and have ***passed both small and large*** *tests of the online judge*
system from leetcode.

Pages

Home

Labels

answer (1) array (19) backtracking (1) BFS (13) binary search (10) binary tree (5) bit manipulations (8) BST (2) bubble sort (1) bucket sort (1) c++ (1) collision (1) common prefix (1) compare (1) design (1) DFS (16) dict (2) divide and conquer (2) DP (10) dynamic programming (1) graph (2) hash (5) hash function (1) hash table (6) heap (2) heap sort (3) Index (2) inorder (2) insert sort. (2) iteration (3) iterative (2) iteratively (2) iterator (1) KMP (1) leetcode (202) linked list (8) list (2) loop (7) map (5) mask (1) math (6) merge sort (2) minimum substring (1) non-recursive (1) O(n) (1) object oriented design (1) online judge (1) oop (1) permutations (1) pointers (9) postorder (1) preorder (2) prime number (1) python (1) questions (1) queue (2) quick sort (1) radix sort (1) Re-view (3) recursive (7) selection sort (1) solutions (1) sort (2) sorting algorithm (2) stack (5) starter (4) string (10) stringstream (2) topology (1) trapping rain water (1) traverse (1) Tree (29) Tree Traversal (8) Trie (1) two pointers (1) twosum (1) XOR (1)

Blog Archive

- ► 2017 (1)
- ► 2016 (25)
- ► 2015 (44)
- ▼ 2014 (15)
  - ► Dec 2014 (4)
  - ▼ Oct 2014 (4)
    - leetcode Question: Single Number II
    - leetcode Quesion: Maximum Product Subarray

- leetcode Question: Find Minimum in Rotated Sorted ...
- leetcode Question: Find Minimum in Rotated Sorted ...

- ► Apr 2014 (1)
- ► Mar 2014 (3)
- ► Feb 2014 (1)
- ► Jan 2014 (2)

- ► 2013 (126)
- ► 2012 (29)