AARGO

HackerRank - 2022 Sep 19

(4 coding questions + 4 logic questions in 100 min, I completed 4 coding questions in 60 min, all pass all test cases in one go)

Question 1

1. Simple Max Difference

In securities research, an analyst will look at a number of attributes for a stock. One analyst would like to keep a record of the highest positive spread between a closing price and the closing price on any prior day in history. Determine the maximum positive spread for a stock given its price history. If the stock remains flat or declines for the full period, return -1.

Example 0

px = [7, 1, 2, 5]

Calculate the positive difference between each price and its predecessors:

- At the first quote, there is no earlier quote to compare to.
- At the second quote, there was no earlier price that was lower.
- At the third quote, the price is higher than the second quote:
  - 2 - 1 = 1
- For the fourth quote, the price is higher than the third and the second quotes:
  - 5 - 2 = 3
  - 5 - 1 = 4.
- The maximum difference is 4.

Example 1

px = [7, 5, 3, 1]

- The price declines each quote, so there is never a difference greater than 0. In this case, return -1.

Function Description

Complete the function *maxDifference* in the editor below.

maxDifference has the following parameters:
int px[n]: an array of stock prices (quotes)
Returns:
int: the maximum difference between two prices as described above

Constraints

- $1 \le n \le 10^5$
- $-10^5 \le px[i] \le 10^5$

This is identical to max profit problem.

```
int maxDifference(vector<int> px)
{
    if (px.size()<=1) return -1;
    int msp = px[1]-px[0]; // modified subproblem
    int ans = msp;

    for(int n=2; n<px.size(); ++n)
    {
        msp = std::max(msp-px[n-1]+px[n], px[n]-px[n-1]);
        ans = std::max(ans, msp);
    }
    if (ans <= 0) return -1;
    else return ans;
}
```

Question 2

2. Frequency Sort

Given an array of *n* item values, sort the array in ascending order, first by the frequency of each value, then by the values themselves.

Example
n = 6
items = [4, 5, 6, 5, 4, 3]

- There are 2 values that occur once: *[3, 6]*.
- There are 2 values that occur twice: *[4, 4, 5, 5]*.
- The array of items sorted by frequency and then by value in ascending order is *[3, 6, 4, 4, 5, 5]*

Function Description
Complete the function *itemsSort* in the editor below.

itemsSort has the following parameter(s):
int items[n]: the array to sort

Returns
int[n]: the sorted array

Constraints
- $1 \le n \le 2 \times 10^5$
- $1 \le items[i] \le 10^6$

The key is to use stable sort.

```
#include<vector>
#include<map>
#include<algorithm>

vector<int> itemsSort(vector<int> items)
{
    std::map<int,int> hist;
    for(const auto& x:items)
    {
        auto iter = hist.find(x);
        if (iter == hist.end())
        {
            hist[x]=1;
        }
        else
        {
            ++iter->second;
        }
    }

    std::vector<std::pair<int,int>> vec; // default comparator of std::pair is element-wise
    for(const auto& x:hist)
    {
        vec.push_back(std::make_pair(x.second,x.first));
    }

    std::stable_sort(vec.begin(), vec.end());
    std::vector<int> output;
    for(const auto& x:vec)
    {
        for(int n=0; n!=x.first; ++n)
        {
            output.push_back(x.second);
        }
    }
    return output;
}
```

Question 3

3. Balance Parantheses

A balanced sequence of parentheses is one in which every opening bracket has a corresponding closing bracket to it. More formally, a sequence of parantheses is considered balanced if it can be represented in the form s1(s2) where both s1 and s2 are either empty or balanced strings.

Given a sequence of parentheses, find the minimum number of swaps needed to make the sequence balanced. It is not necessary to swap adjacent characters only. If it is impossible to balance the string, return -1.

Example
brackets = "))((()("

Swap the characters at the first and last index to get *"((()(()))"* which is balanced. The string can be balanced with1 swap.

Function Description
Complete the function *minimumSwaps* in the editor below.

minimumSwaps has the following parameter(s):
string brackets: the string to analyze

Constraints

int: the minimum number of swaps or -1

Constraints

- $1 \le$ length of the string *brackets* $\le 10^5$
- *brackets* consists of *')'* and *'('* only.

The idea is that when former (is paired with latter), then it is balanced. Therefore imbalance occurs only when) occurs before (. In short we can consider (as +1 and) as -1 and find the cumulative sum, the negative sum means the number of inversed brackets. To fix them, we need to swap, each swap and solve two pairs of inverted brackets.

```
int minimumSwaps(string brackets)
{

    int sum = 0;
    int min = 0;

    for(const char& c:brackets)
    {
        if (c == '(') ++sum;
        else
        {
            --sum;
            if (sum < min) min = sum;
        }
    }
    if (sum != 0) return -1;
    else return (-min+1)/2; // one swap solves two inversed pairs
}
```

Question 4

4. Condensed List

Given a list of integers, remove any nodes that have values that have previously occurred in the list and return a reference to the head of the list. For example, the following list has a recurrence of the value *3* initially:

Redundant nodes are colored with the same color

Redundant nodes are removed after calling *condense*

Remove the node at position *2* in the list above, *0* based indexing.

Function Description

Complete the function *condense* in the editor below.

condense has the following parameter(s):

head: the head of a singly-linked list of integers, a LinkedListNode

Note:. A *LinkedListNode* has two attributes: *data*, an integer, and *next*, a reference to the next item in the list or the language equivalent of *null* at the tail.

Returns

reference to LinkedListNode: the head of the list of distinct values

Constraints

- $1 \le n \le 10^5$
- $0 \le LinkedListNode[i].val \le 1000$

```
#include<unordered_set>
SinglyLinkedListNode* condense(SinglyLinkedListNode* head)
{
    std::unordered_set<int> set;
    set.insert(head->data);

    auto* prev_node = head;
    auto* this_node = head->next;
    while(this_node)
    {
        auto iter = set.find(this_node->data);
        if (iter == set.end())
        {
            set.insert(this_node->data);
            prev_node = this_node;
            this_node = this_node->next;
        }
        else // remove this_node
        {
            auto*  del_node = this_node;
            prev_node->next = this_node->next;
            this_node = this_node->next;

            // question : is this function responsible for deleting this_node ?
            delete del_node;
        }
    }
    return head;
}
```

5. Standing Order 1

There are five people - Peter, Tyson, Richard, Steve and Quinn - standing in a row.
Given the following statements, who is on the rightmost position?

1. Peter is next to Quinn and Steve is next to Richard.
2. Steve is not next to Tyson.
3. Tyson is on leftmost position.
4. Richard is on the second position from the right.
5. Peter stands somewhere to the right of both Quinn and Tyson.
6. Peter and Richard are next to each other.

6. Even Products Count

Three integers are chosen at random from the first thirty terms of the sequence an=n. the probability that their product is even is?

Pick **ONE** option

- ◯ (30C3-15C3)/30C3
- ◯ (30C3+15C3)/30C3

7. Cryptogram 2

If "PLATINUM" is coded as "AIUPLTNM", how would "ADVENTURE" be coded?

Pick **ONE** option

- ◯ AEEUDNRTV
- ◯ AEUEDNTVR
- ◯ ADEENRTUV
- ◯ AEUEDVNTR

8. Louis College Puzzle 2

Louis College London offers six subjects: math, chemistry, physics, botany, zoology and anatomy.
Each subject has its own teacher, and each teacher teaches a different number of sections from 1 - 6.
Given the following information, how many sections does Dr. George teach?

1. Dr. Mason teaches 4 sections.
2. Anatomy has 5 sections.
3. Dr. Smith teaches 2 more sections than Dr. Noah. Neither teaches botany nor zoology.
4. Dr. Thomas teaches less than 6 sections of physics.
5. There are more sections of chemistry than math. Dr. Lucas does not teach chemistry.
6. There are 3 more sections of zoology than botany.