AlgoExpert

Quad Layout C++ 12px Sublime Monokai 00:00:00

Prompt Scratchpad Our Solution(s) Video Explanation Run Code

Solution 1 Solution 2

```cpp
// Copyright © 2020 AlgoExpert, LLC. All rights reserved.

#include <vector>
#include <climits>
using namespace std;

// O(nk) time | O(n) space
int maxProfitWithKTransactions(vector<int> prices, int k) {
  if (prices.size() == 0) {
    return 0;
  }
  int *evenProfits = new int[prices.size()];
  int *oddProfits = new int[prices.size()];
  for (int i = 0; i < prices.size(); i++) {
    evenProfits[i] = 0;
    oddProfits[i] = 0;
  }
  for (int t = 1; t < k + 1; t++) {
    int maxThusFar = INT_MIN;
    int *currentProfits = new int[prices.size()];
    int *previousProfits = new int[prices.size()];
    if (t % 2 == 1) {
      currentProfits = oddProfits;
      previousProfits = evenProfits;
    } else {
      currentProfits = evenProfits;
      previousProfits = oddProfits;
    }
    for (int d = 1; d < prices.size(); d++) {
      maxThusFar = max(maxThusFar, previousProfits[d - 1] - prices[d - 1]);
      currentProfits[d] = max(currentProfits[d - 1], maxThusFar + prices[d]);
    }
  }
  return k % 2 == 0 ? evenProfits[prices.size() - 1]
                    : oddProfits[prices.size() - 1];
}
```