

Our Solution(s)

Run Code

Your Solutions

Run Code

Solution 1Solution 2

```
1 // Copyright © 2020 AlgoExpert, LLC. All rights reserved.
2
3 #include <vector>
4 #include <climits>
5 using namespace std;
6
7 // O(nk) time | O(nk) space
8 int maxProfitWithKTransactions(vector<int> prices, int k) {
9     if (prices.size() == 0) {
10         return 0;
11     }
12     vector<vector<int>> profits(k + 1, vector<int>(prices.size(), 0));
13     for (int t = 1; t < k + 1; t++) {
14         int maxThusFar = INT_MIN;
15         for (int d = 1; d < prices.size(); d++) {
16             maxThusFar = max(maxThusFar, profits[t - 1][d - 1] - prices[d - 1]);
17             profits[t][d] = max(profits[t][d - 1], maxThusFar + prices[d]);
18         }
19     }
20     return profits[k][prices.size() - 1];
21 }
22
```

Solution 1Solution 2Solution 3

```
1 #include <vector>
2 using namespace std;
3
4 int maxProfitWithKTransactions(vector<int> prices, int k) {
5     // Write your code here.
6     return -1;
7 }
8
```

Our Tests

Custom Output

Submit Code

