

Solution 1Solution 2

1// Copyright © 2020 AlgoExpert, LLC. All rights reserved.  
2  
3package main  
4  
5import "math"  
6  
7// O(nk) time | O(nk) space  
8func MaxProfitWithKTransactions(prices []int, k int) int {  
9 if len(prices) == 0 {  
10 return 0  
11 }  
12 profits := make([][]int, k+1)  
13 for i := range profits {  
14 profits[i] = make([]int, len(prices))  
15 }  
16 for t := 1; t < k+1; t++ {  
17 maxThusFar := math.MinInt32  
18 for d := 1; d < len(prices); d++ {  
19 maxThusFar = max(maxThusFar, profits[t-1][d-1]-prices[d-1])  
20 profits[t][d] = max(profits[t][d-1], maxThusFar+prices[d])  
21 }  
22 }  
23 return profits[k][len(prices)-1]  
24 }  
25  
26func max(arg int, rest ...int) int {  
27 curr := arg  
28 for \_, num := range rest {  
29 if curr < num {  
30 curr = num  
31 }  
32 }  
33 return curr  
34 }  
35

