

# GeeksforGeeks

A computer science portal for geeks

Practice

IDE

Q&A

GeeksQuiz

## Stock Buy Sell to Maximize Profit

The cost of a stock on each day is given in an array, find the max profit that you can make by buying and selling in those days. For example, if the given array is {100, 180, 260, 310, 40, 535, 695}, the maximum profit can be earned by buying on day 0, selling on day 3. Again buy on day 4 and sell on day 6. If the given array of prices is sorted in decreasing order, then profit cannot be earned at all.

If we are allowed to buy and sell only once, then we can use following algorithm. **Maximum difference between two elements**. Here we are allowed to buy and sell multiple times. Following is algorithm for this problem.

1. Find the local minima and store it as starting index. If not exists, return.
2. Find the local maxima. and store it as ending index. If we reach the end, set the end as ending index.
3. Update the solution (Increment count of buy sell pairs)
4. Repeat the above steps if end is not reached.

```
// Program to find best buying and selling days
#include <stdio.h>

// solution structure
struct Interval
{
    int buy;
    int sell;
};

// This function finds the buy sell schedule for maximum profit
void stockBuySell(int price[], int n)
{
    // Prices must be given for at least two days
    if (n == 1)
        return;

    int count = 0; // count of solution pairs

    // solution vector
    Interval sol[n/2 + 1];

    // Traverse through given price array
    int i = 0;
    while (i < n-1)
    {
        // Find Local Minima. Note that the limit is (n-2) as we are
        // comparing present element to the next element.
        while ((i < n-1) && (price[i+1] <= price[i]))
            i++;
    }
```

```
// If we reached the end, break as no further solution possible
if (i == n-1)
    break;

// Store the index of minima
sol[count].buy = i++;

// Find Local Maxima. Note that the limit is (n-1) as we are
// comparing to previous element
while ((i < n) && (price[i] >= price[i-1]))
    i++;

// Store the index of maxima
sol[count].sell = i-1;

// Increment count of buy/sell pairs
count++;
}

// print solution
if (count == 0)
    printf("There is no day when buying the stock will make profit\n");
else
{
    for (int i = 0; i < count; i++)
        printf("Buy on day: %d\t Sell on day: %d\n", sol[i].buy, sol[i].sell);
}

return;
}

// Driver program to test above functions
int main()
{
    // stock prices on consecutive days
    int price[] = {100, 180, 260, 310, 40, 535, 695};
    int n = sizeof(price)/sizeof(price[0]);

    // function call
    stockBuySell(price, n);

    return 0;
}
```

[Run on IDE](#)

Output:

```
Buy on day : 0    Sell on day: 3
Buy on day : 4    Sell on day: 6
```

**Time Complexity:** The outer loop runs till  $i$  becomes  $n-1$ . The inner two loops increment value of  $i$  in every iteration. So overall time complexity is  $O(n)$

This article is compiled by [Ashish Anand](#) and reviewed by GeeksforGeeks team. Please write comments if you find anything incorrect, or you want to share more information about the topic discussed above.



# Top SIP Investment plans

Compare & invest in best  
Funds. 0 paper work. Start  
today.



90 Comments Category: Arrays

## Related Posts:

- Longest Span with same Sum in two Binary arrays
- Count Inversions of size three in a give array
- Find the subarray with least average
- Count triplets with sum smaller than a given value
- Find zeroes to be flipped so that number of consecutive 1's is maximized
- Reorder an array according to given indexes
- Find maximum value of  $\text{Sum}(i * \text{arr}[i])$  with only rotations on given array allowed
- Find maximum average subarray of k length

(Login to Rate and Mark)

3.5

Average Difficulty : 3.5/5.0  
Based on 8 vote(s)

☐

Add to TODO List

☐

Mark as DONE

Like Share 24 people like this. Be the first of your friends.

Writing code in comment? Please use [code.geeksforgeeks.org](http://code.geeksforgeeks.org), generate link and share the link here.

@geeksforgeeks, Some rights reserved

Contact Us!

About Us!

Advertise with us!