

# Combo Meal

by kevinsogo

locked

Problem	Submissions	Leaderboard	Discussions	Editorial
---------	-------------	-------------	-------------	-----------

Submitted a few seconds ago • Score: 15.00

Status: Accepted

✓	Test Case #0	✓	Test Case #1	✓	Test Case #2
✓	Test Case #3	✓	Test Case #4	✓	Test Case #5
✓	Test Case #6	✓	Test Case #7	✓	Test Case #8
✓	Test Case #9	✓	Test Case #10	✓	Test Case #11
✓	Test Case #12	✓	Test Case #13	✓	Test Case #14
✓	Test Case #15	✓	Test Case #16	✓	Test Case #17
✓	Test Case #18	✓	Test Case #19	✓	Test Case #20

## Submitted Code

Language: C++14

Open in editor

```
1 #include <bits/stdc++.h>
2
3 using namespace std;
4
5 vector<string> split_string(string);
6
7 int profit(int b, int s, int c)
8 {
9     return b + s - c;
10 }
11
12 int main()
13 {
14     ofstream fout(getenv("OUTPUT_PATH"));
15
16     int t;
17     cin >> t;
18     cin.ignore(numeric_limits<streamsize>::max(), '\n');
19
20     for (int t_itr = 0; t_itr < t; t_itr++)
21     {
22         string bsc_temp;
23         getline(cin, bsc_temp);
24
25         vector<string> bsc = split_string(bsc_temp);
26
27         int b = stoi(bsc[0]);
28         int s = stoi(bsc[1]);
29         int c = stoi(bsc[2]);
30
31         int result = profit(b, s, c);
32
33         fout << result << "\n";
34     }
35
36     fout.close();
37
38     return 0;
39 }
40
41 vector<string> split_string(string input_string)
42 {
43     string::iterator new_end = unique(input_string.begin(), input_string.end(), [] (const char &x, const char &y)
44     {
45         return x == y and x == ' ';
46     });
47
48     input_string.erase(new_end, input_string.end());
49
50     while (input_string[input_string.length() - 1] == ' ')
51     {
52         input_string.pop_back();
53     }
54
55     vector<string> splits;
56     char delimiter = ' ';
57
58     size_t i = 0;
59     size_t pos = input_string.find(delimiter);
60
61     while (pos != string::npos)
62     {
63         splits.push_back(input_string.substr(i, pos - i));
64         i = pos + 1;
65         pos = input_string.find(delimiter, i);
66     }
67
68     splits.push_back(input_string.substr(i, min(pos, input_string.length()) - i + 1));
69
70     return splits;
71 }
72
```