All Competitions > Week of Code 30 > Candy Replenishing Robot

Candy Replenishing Robot

by bishop15

| Problem	Submissions	Leaderboard	Discussions

Your submission will run against only preliminary test cases. Full test cases will run at the end of the day.

Alice is hosting a party! The party lasts for t minutes, and she puts out a bowl of n candies at the beginning of the party. During each minute i, a person comes to the bowl and removes c_i candies.

Alice programs the following algorithm into her robot, Bob, to replenish the candy throughout the party:

- If the party is ending (i.e., it's time t), do not refill the bowl.

- If the bowl contains $b_{i_{end}}$ candies at the end of minute i and $b_{i_{end}} < 5$, add $n - b_{i_{end}}$ candies to the bowl.

For example, if $n = 8$, $t = 3$, and $c = [5, 2, 4]$, then the candy bowl looks like this throughout the party:
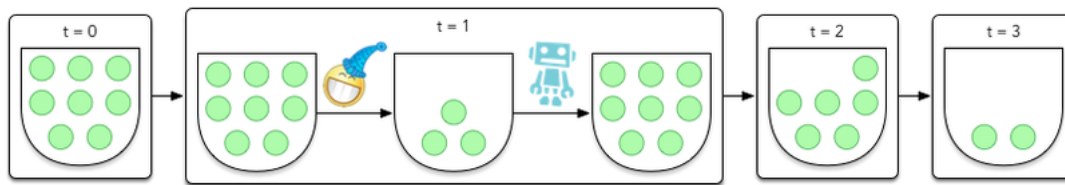
Note that Bob doesn't replenish the bowl at the party's end, so a total of 5 candies were added during the party.

Given n, t, and the number of candies removed from the bowl during each minute, print the total number of new candies Bob adds to the bowl during the party.

Input Format

The first line contains two space-separated integers describing the respective values of n and t.
The second line contains t space-separated integers describing the respective values of $c_0, c_1, \ldots, c_{t-1}$.

Constraints

- $5 \leq n \leq 100$

- $1 \leq t \leq 100$

- $1 \leq c_i \leq n$

- $c_i \leq b_{i_{start}}$, where $b_{i_{start}}$ is the number of candies in the bowl at the start of minute i.

Output Format

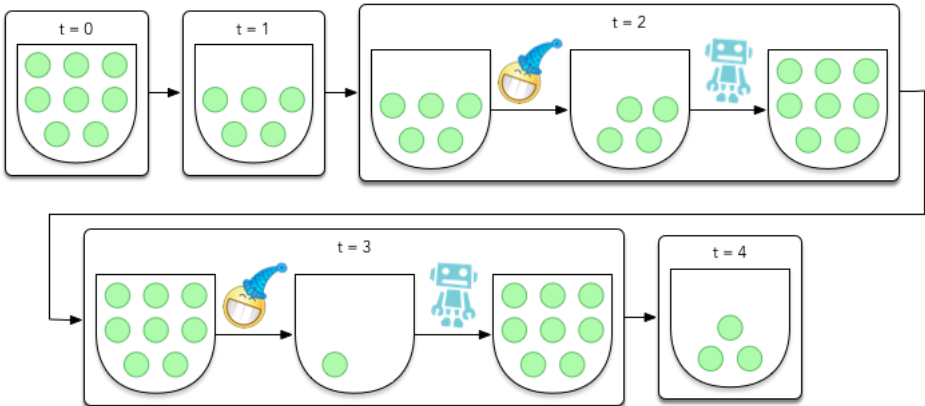Print the total number of new candies Bob adds to the bowl during the party.

Sample Input 0

```
8 4
3 1 7 5
```

Sample Output 0

```
11
```

Explanation 0

The party starts out with $n = 8$ candies in the bowl and the candies removed during each second are denoted by $c = [3, 1, 7, 5]$. We break down each minute of the party like so:

1. Remove $c_0 = 3$ candies, so $8 - 3 = 5$ candies remain.

2. Remove $c_1 = 1$ candies, so $5 - 1 = 4$ candies remain. Because the party is still going on, Bob refills the bowl by adding 4 new candies so it again contains $4 + 4 = 8$ candies.

3. Remove $c_2 = 7$ candies, so $8 - 7 = 1$ candy remains. Because the party is still going on, Bob refills the bowl by adding 7 new candies so it again contains $1 + 7 = 8$ candies.

4. Remove $c_3 = 5$ candies, so $8 - 5 = 3$ candies remain. Because the party is ending during this minute, Bob does not refill the bowl.

We then print the total number of candies added during the party, which is $4 + 7 = 11$.

f　　y　　in

Contest ends in 7 days

Submissions: 3756
Max Score: 10
Difficulty: Easy

Rate This Challenge:
★★★★★　Thanks!

More

| Current Buffer (saved locally, editable) ⅄ 🕑 | | C# ⌄ | ⤢ ⚙ |

```csharp
1  using System;
2  using System.Collections.Generic;
3  using System.IO;
4  using System.Linq;
5  class Solution {
6
7      static void Main(String[] args) {
8          string[] tokens_n = Console.ReadLine().Split(' ');
9          int n = Convert.ToInt32(tokens_n[0]);
10         int t = Convert.ToInt32(tokens_n[1]);
11         string[] c_temp = Console.ReadLine().Split(' ');
12         int[] c = Array.ConvertAll(c_temp, Int32.Parse);
13         // your code goes here
14
15         int n_inicial =n;
16         int agrego = 0;
17         for (int tiempo= 1; tiempo <= t; tiempo++)
```

```
18 ▾          {
19                n -= c[tiempo-1];
20                if (tiempo < t && n < 5)
21 ▾            {
22                    agrego += (n_inicial - n);
23                    n += (n_inicial - n);
24                }
25            }
26
27            Console.WriteLine(agrego);
28        }
29 }
30
```

Line: 19 Col: 30

⬆ Upload Code as File   ☐ Test against custom input   Run Code   Submit Code

```
18 ▾          {
19                n -= c[tiempo-1];
20                if (tiempo < t && n < 5)
```