**Biased Chandan**

Attempted by: **4330**

/

Accuracy: **79%**

/

Maximum Score: **20**

/

262 Votes

Tag(s):

Ad-Hoc, Data Structures, Easy, Stack

**PROBLEM**

**EDITORIAL**

**MY SUBMISSIONS**

**ANALYTICS**

Chandan is an extremely biased person, and he dislikes people who fail to solve all the problems in the interview he takes for hiring people. There are **n** people on a day who came to be interviewed by Chandan.

Chandan rates every candidate from **0 to 10.** He has to output the total ratings of all the people who came in a day. But, here's the problem: Chandan gets extremely frustrated when someone ends up scoring a 0 in the interview. So in frustration he ends up removing the candidate who scored that 0, and also removes the candidate who came before him. If there is no candidate before the one who scores a 0, he does nothing.

You've to find the summation of all the ratings in a day for Chandan.

**Input constraints:**  
The first line of input will contain an integer — **n**. The next *n* lines will contain an integer, where the ith integer represents the rating of the ithperson.

**Output constraints:**  
Print the required sum.

**Constraints:**  
1 ≤ n ≤5 \* 103  
0 ≤ Value of ratings ≤10

**SAMPLE INPUT**

5

2

3

0

7

0

**SAMPLE OUTPUT**

2

**Time Limit:**2.0 sec(s) for each input file.

**Memory Limit:**256 MB

**Source Limit:**1024 KB

**Marking Scheme:**Marks are awarded when all the testcases pass.

**Allowed Languages:**C, C++, Clojure, C#, D, Erlang, F#, Go, Groovy, Haskell, Java, Java 8, JavaScript(Rhino), JavaScript(Node.js), Lisp, Lisp (SBCL), Lua, Objective-C, OCaml, Octave, Pascal, Perl, PHP, Python, Python 3, R(RScript), Racket, Ruby, Rust, Scala, Scala 2.11.8, Swift, Visual Basic

<https://www.hackerearth.com/practice/data-structures/arrays/1-d/practice-problems/algorithm/biased-chandan/>

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

class MyClass

{

static void Main(string[] args)

{

int n = int.Parse(Console.ReadLine());

int[] a = new int[n];

for (int i = 0; i < n; i++)

{

a[i] = int.Parse(Console.ReadLine());

}

bool[] marcas = new bool[a.Length];

for (int i = 0; i < a.Length; i++)

{

if (a[i] == 0)

{

// marcas[i+1] = true;

marcas[i] = true;

int k = i;

while (k >= 0 && marcas[k])

{

k--;

}

marcas[k] = true;

}

}

int sum = 0;

for (int i = 0; i < marcas.Length; i++)

{

//Console.Write(marcas[i] + " ");

if (!marcas[i])

{

sum += a[i];

}

}

Console.WriteLine(sum);

}

}