Maximum Chocolate - Hackerearth

Tuesday, March 3, 2020 8:24 PM

The Problem:

Bharat is chocoholic. He found a chocolate factory of **N** floors ,but the factory has **N*N rooms** where in each room specific number of chocolates are present. Now, Bharat starts collecting chocolates from the ground floor. He can only collect chocolates from one room in a floor. Bharat can only move to the upper, or upper-right or upper-left rooms.

Help him collect the maximum number of chocolates possible.

Input:
First line contains the value of N. Next N lines contains N space separated integers.

Output:
Output a single integer denoting the maximum number of chocolates Bharat can collect.

Constraints:
1 <= N <= 1000
1 <= Number chocolate in 1 room <= 10^5

SAMPLE INPUT

5
1 2 3 4 5
1 2 3 4 5
1 2 3 4 5
1 2 3 4 5
100 2 3 4 5

SAMPLE OUTPUT

114

The Code:

```
/*
 * The goal is to find the maximum number of chocolates per floor in a top-down approach.
 * Start with the topmost floor, maximum chocolates is the room with the max no. of chocolates.
 * From top-second floor, find the maximum obtained for every room in the floor. Then find the maximum amongst all the rooms.
 * Continue this approach.
 * Optimize in terms of space by calculating max values on the fly. No need of an N*N array for storing the values.
 */

#include <stdio.h>
```

```
int main(){
	int N, i, j, currRow, prevRow;
	scanf("%d", &N);
	long int maxChocos[2][N], max, top, top_left, top_right;

	for(i=0; i<N; i++) {
	   currRow = i % 2;
	   prevRow = abs(currRow - 1);
	   max = 0;

	   for(j=0; j<N; j++) {

	      scanf("%ld", &maxChocos[currRow][j]);
	      if(i > 0) {
	         top_left = j > 0 ? maxChocos[prevRow][j-1] : 0;
	         top_right = j < N-1 ? maxChocos[prevRow][j+1] : 0;
	         top = maxChocos[prevRow][j];

	         maxChocos[currRow][j] += (top_left > top_right) ? (top_left > top ? top_left : top) :
(top_right > top ? top_right : top);
	      }
	      if(maxChocos[currRow][j] > max) {
	         max = maxChocos[currRow][j];
	      }
	   }
	}

	printf("%ld\n", max);

}
```

The stats:

Score
30.0

Time (sec)
0.76036

Memory (KiB)
64

Language
C