

CSI3108-01

2016. 10. 24

Programming HW#4

Max 30 points

Due on Nov. 1 (Tue) by 5pm

You are given a game board which has 4 rows and N columns, and has a non-zero integer written in each square. You are also given a set of $2N$ white stones and $2N$ black stones, and you want to place some or all of these stones on the game board so as to maximize the sum of the integers in the squares that are covered by the stones under the condition that a negative integer under a black stone becomes its corresponding positive number.

There are two rules in placing the stones to be legal;

- (1) No two of them can be on horizontally or vertically adjacent squares, but diagonal adjacency is acceptable.
- (2) Each stone can be placed on exactly one square.

3	1	-3	2	-2
1	9	2	-4	1
-2	2	4	1	-9
-8	30	3	1	20

Fig. 1

(3)	1	(-3)	2	(-2)
1	(9)	2	(-4)	1
(-2)	2	(4)	1	-9
-8	(30)	3	1	(20)

Fig. 2

Fig. 1 shows a sample game board. The optimal placement of the white and black stones is shown in Fig. 2; the value of the placement is 77. Write a Java program to find the maximum sum of the numbers under the black and white stones for a given game board.

[Input]

The test cases consist of the following format. In the first line, the number of test cases is given. From the next line, each test case is given in 5 lines. In the first line, the number N of columns of the game board is given. Assume that N is at most 100. In the next 4 lines, the numbers in each row of the game board are given on a separate line. (Number of stones are always even number)

[Output]

Print out the maximum value that can be obtained from the board for each of the test cases on a separate line.

[Sample Input and Output]

Input

2 ←the number of test cases

5 ←test case 1 (Column)

3 1 -3 2 -2

1 9 2 -4 1

-2 2 4 1 -9

-8 30 3 1 20

7 ←test case 2

-14 -11 40 3 -9 42 6

32 -30 39 -28 -1 -36 -32

-16 43 46 -45 -44 41 47

39 -38 -25 -23 -29 29 -29

Output

77

411