

ACSL
American Computer Science League

ACSL Kakuro

2005 - 2006

All-Star #8

PROBLEM: Kakuro (see www.kakuro.com/www.kakuro.net) is another fill in the numbers game. In the board shown the X's in some of the squares indicates that those squares are not used. The squares with the diagonals are the direction squares. The 3 at location (1,4) and below the diagonal states that you must fill in the column below, using only the digits 1 – 9, such that the sum of the 2 digits will be 3. The 10 at location (3,1) and above the diagonal states that you must fill in the row, using only the digits 1 - 9, such that the sum of the 4 digits will be 10. No digit in the same row or column can repeat. There is just one combination for 3 numbers whose sum is 6 (1+2+3). There are two combinations for 2 numbers whose sum is 6 (4+2 and 5+1). The puzzle has a unique solution.

X	X	X	3	6
X	4	3	3	
10				
3			X	X

2	1
1	2

Note that this type of arrangement is not unique and is not allowed.

INPUT: There will be 13 lines of input. The first line gives the size of the grid in row/column order. Rather than using ordered pairs to name the individual squares, the squares will be numbered from 1 through 20 for the 4 x 5 grid above. The square with ordered pair (1,1) is numbered 1. The other squares are numbered in sequence across the rows. The second line will provide the number of X's and their numbered locations. The third line will contain pairs that give the location and rule of the direction squares. A denotes above the diagonal and B denotes below the diagonal. The last pair will be 0,0 signifying the end of the data line. The last 10 lines of input will each be a numbered location. Note that our example only has 11 input lines and only 8 outputs. Remember to change your program before entering the test data.

OUTPUT: For the last 10 lines of input, print the number at that location once the grid has been filled according to the rules above.

SAMPLE INPUT

1. 4,5
2. 6, 1, 2, 3, 6, 19, 20
3. 4, 3B, 5, 6B, 7, 4B, 8, 3A, 8, 3B, 11,10A, 16, 3A, 0, 0
4. 9
5. 10
6. 12
7. 13
8. 14
9. 15
10. 17
11. 18

SAMPLE OUTPUT

1. 1
2. 2
3. 3
4. 1
5. 2
6. 4
7. 1
8. 2