Forming a Magic Square ☆

We define a magic square to be an $n \times n$ matrix of distinct positive integers from 1 to n^2 where the sum of any row, column, or diagonal of length n is always equal to the same number: the *magic constant*.

You will be given a 3×3 matrix s of integers in the inclusive range $[1, 9]$. We can convert any digit a to any other digit b in the range $[1, 9]$ at cost of $|a - b|$. Given s, convert it into a magic square at *minimal* cost. Print this cost on a new line.

Note: The resulting magic square must contain distinct integers in the inclusive range $[1, 9]$.

For example, we start with the following matrix s:

```
5 3 4
1 5 8
6 4 2
```

We can convert it to the following magic square:

```
8 3 4
1 5 9
6 7 2
```

This took three replacements at a cost of $|5 - 8| + |8 - 9| + |4 - 7| = 7$.

Function Description

Complete the *formingMagicSquare* function in the editor below. It should return an integer that represents the minimal total cost of converting the input square to a magic square.

formingMagicSquare has the following parameter(s):

- *s*: a 3×3 array of integers

Input Format

Each of the lines contains three space-separated integers of row $s[i]$.

Constraints

- $s[i][j] \in [1, 9]$

Output Format

Print an integer denoting the minimum cost of turning matrix s into a magic square.

Sample Input 0

```
4 9 2
3 5 7
8 1 5
```

Sample Output 0

```
1
```

Explanation 0

If we change the bottom right value, $s[2][2]$, from 5 to 6 at a cost of $|6-5|=1$, s becomes a magic square at the minimum possible cost.

Sample Input 1

```
4 8 2
4 5 7
6 1 6
```

Sample Output 1

```
4
```

Explanation 1

Using 0-based indexing, if we make

- $s[0][1]$->9 at a cost of $|9-8|=1$
- $s[1][0]$->3 at a cost of $|3-4|=1$
- $s[2][0]$->8 at a cost of $|8-6|=2$,

then the total cost will be $1+1+2=4$.

C#

```
16
17      // Complete the formingMagicSquare function below.
18      static int formingMagicSquare(int[][] s) {
19
20          var square = new List<int>();
21
22          for (int i = 0; i < 3; i++)
23          {
24              foreach (var item in s[i])
25              square.Add(Convert.ToInt32(item));
26          }
27
28          var allMagics = new int[][]
29          {new int[] { 8, 1, 6, 3, 5, 7, 4, 9, 2},
30           new int[] { 6, 1, 8, 7, 5, 3, 2, 9, 4},
31           new int[] { 4, 9, 2, 3, 5, 7, 8, 1, 6},
32           new int[] { 2, 9, 4, 7, 5, 3, 6, 1, 8},
33           new int[] { 8, 3, 4, 1, 5, 9, 6, 7, 2},
34           new int[] { 4, 3, 8, 9, 5, 1, 2, 7, 6},
35           new int[] { 6, 7, 2, 1, 5, 9, 8, 3, 4},
36           new int[] { 2, 7, 6, 9, 5, 1, 4, 3, 8}};
37
38           var sums = from item in allMagics
39                      select item.Zip(square, (a, b) => Math.Abs(a - b)).Sum();
40
41           //Console.WriteLine(sums.Min());
42
```