You are given a matrix that contains booleans. If an element is true, it means that it is colored black, otherwise it is colored white.

Your task is to find perimeter of the objects colored black.

**Example**

For

matrix = [[false, false, false, true, false],

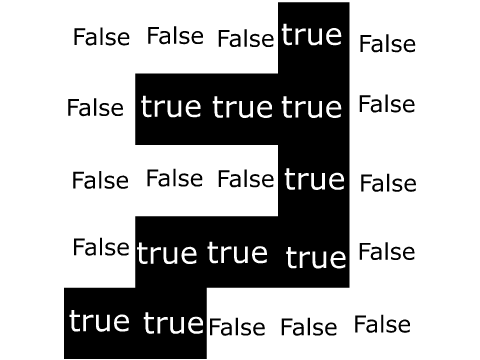
[false, true, true, true, false],

[false, false, false, true, false],

[false, true, true, true, false],

[ true, true, false, false, false]]

the output should be  
MatrixPerimeter(matrix) = 22.



As you can see, perimeter of the area colored black is 22.

**Input/Output**

* **[time limit] 3000ms (cs)**
* **[input] array.array.boolean matrix**

A rectangular matrix.

*Constraints:*  
1 ≤ matrix.length ≤ 100,  
1 ≤ matrix[i].length ≤ 100.

* **[output] integer**

Perimeter of the black figures.

<https://codefights.com/challenge/J6xCK2vBHNn5LB4YM/main>

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

namespace ConsoleApplication1

{

class Program

{

static int MatrixPerimeter(bool[][] matrix)

{

int perimetro = 0;

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

{

for (int j = 0; j < matrix[i].Length; j++)

{

if (matrix[i][j] == true)

{

if ( i-1 >=0 && matrix[i - 1][j] == false)

{

perimetro++;

}

if (i - 1 < 0)

{

perimetro++;

}

if (i + 1 < matrix.Length && matrix[i + 1][j] == false)

{

perimetro++;

}

if (i + 1 >= matrix.Length)

{

perimetro++;

}

if (j - 1 >= 0 && matrix[i][j - 1] == false)

{

perimetro++;

}

if (j - 1 < 0)

{

perimetro++;

}

if (j + 1 < matrix[i].Length && matrix[i][j + 1] == false)

{

perimetro++;

}

if (j + 1 >= matrix[i].Length)

{

perimetro++;

}

}

}

}

return perimetro;

}

static void Main(string[] args)

{

}

}

}