Author

[cordless](https://codefights.com/profile/cordless)

https://codefights.com/img/coins_new.png2000

Lisa likes to play sudoku in her free time. However, since she only solves sudokus that are published in the local newspaper, she can check her solutions only by manually checking each row, column and subgrid, which is super boring.

Your job is to help Lisa by writing a program which accepts a 9 × 9 matrix and determines if it follows sudoku rules. Each row, column and a 3 × 3subgrid in the matrix must contain all digits from 1to 9 with no repetitions.

**Example**

For

s = [[5,3,4,6,7,8,9,1,2],

[6,7,2,1,9,5,3,4,8],

[1,9,8,3,4,2,5,6,7],

[8,5,9,7,6,1,4,2,3],

[4,2,6,8,5,3,7,9,1],

[7,1,3,9,2,4,8,5,6],

[9,6,1,5,3,7,2,8,4],

[2,8,7,4,1,9,6,3,5],

[3,4,5,2,8,6,1,7,9]]

the output should be  
Sudoku(s) = true.

The given sudoku is correct.

**Input/Output**

* **[time limit] 3000ms (cs)**
* **[input] array.array.integer s**

A 9 × 9 matrix containing integers from 1 to9.

* **[output] boolean**

true if the given sudoku is correct, falseotherwise.

<https://codefights.com/challenge/KjwiZjbgJ2rHRuWPk/main>

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

namespace ConsoleApplication1

{

class Program

{

static bool Sudoku(List<List<int>> s)

{

//comprobar fila por fila

for (int i = 0; i < 9; i++)

{

HashSet<int> filaActual = new HashSet<int>();

for (int j = 0; j < 9; j++)

{

// if (char.IsNumber(s[i][j]))

{

if (filaActual.Contains( s[i][j]))

{

return false;

}

filaActual.Add(s[i][ j]);

}

}

}

//comprobar columna por columna

for (int j = 0; j < 9; j++)

{

HashSet<int> colActual = new HashSet<int>();

for (int i = 0; i < 9; i++)

{

// if (char.IsNumber(s[i][j]))

{

if (colActual.Contains(s[i][j]))

{

return false;

}

colActual.Add(s[i][ j]);

}

}

}

for (int fila = 0; fila < 9; fila += 3)

{

for (int col = 0; col < 9; col += 3)

{

HashSet<int> cuadrante = new HashSet<int>();

for (int i = fila; i < fila + 3; i++)

{

for (int j = col; j < col + 3; j++)

{

// if (char.IsNumber(s[i][ j]))

{

if (cuadrante.Contains(s[i][j]))

{

return false;

}

cuadrante.Add(s[i][ j]);

}

}

}

}

}

return true;

}

static void Main(string[] args)

{

}

}

}