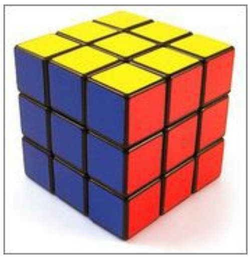
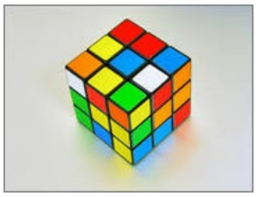
The Rubix Cube: Is it solved?

You are provided with an image of a Rubix cube, which could have been clicked from a number of different angles. Your task is to analyze the provided image of the cube and identify whether **all the visible sides** have been 'solved', i.e. all the squares, on each of the visible faces, are of the same color. Among the cubes below, Cube 1, 4 and 6 are considered 'solved' based on the faces which are visible in the image. The remaining cubes need to be identified as 'unsolved'.

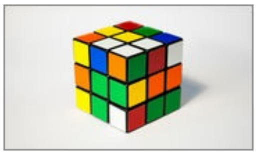
Cube 1



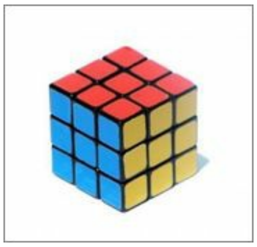
Cube 2



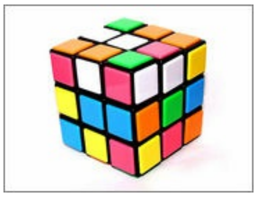
Cube 3



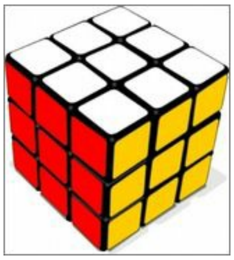
Cube 4



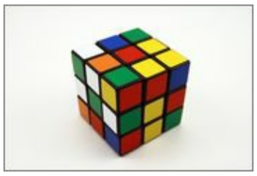
Cube 5



Cube 6



Cube 7



Input Format

A 2D Grid of pixel values will be provided (in regular text format through STDIN), which represent the pixel wise values from the images (which were originally in JPG or PNG formats).
Each pixel will be represented by three comma separated values in the range 0 to 255 representing the **Blue, Green and Red** components respectively. The will be a space between successive pixels in the same row.

Input Constraints

None of the original JPG or PNG images exceeded 150kB in size. The 2D grids of pixels representing these images will not exceed 5MB.

Sample Input

This is for the purpose of explanation only. The real inputs will be much larger than this.

```
0,0,200 0,0,10 10,0,0
90,90,50 90,90,10 255,255,255
100,100,88 80,80,80 15,75,255
```

The above is an image represented by 3x3 pixels. For each pixel the Blue, Green and Red values are provided, separated by commas. The top left pixel has (Blue=0,Green=0,Red=200). The top-right pixel has (Blue=10,Green=0,Red=0). The bottom-right pixel has (Blue=15,Green=75,Red=255). The bottom-left pixel has (Blue=100,Green=100, Red=88).

Output Format

Just one word: 'solved' or 'unsolved'. Do NOT include the single quote marks.

Sample Output

(Please note that the sample input shown above does not actually contain a face!)

```
solved
```

A Note on the Test Cases and Sample Tests

The test cases have been generated from the 20 images out of which the first 7 have been shown in the picture at the top. These seven test cases, are also available as visible, sample test cases when you "Compile and Test" your solution. After the contest is over, the submissions will be re-run after adding the remaing thirteen test cases (hidden test cases which will not be available during the duration of the contest).

Libraries

Libraries available in our Machine Learning/Real Data challenges will be enabled for this contest and are listed here. Please note, that occasionally, a few functions or modules might not work in the constraints of our infrastructure. For instance, some modules try to run multiple threads (and fail). So please try importing the library and functions and cross checking if they work in our online editor in case you plan to develop a solution locally, and then upload to our site.