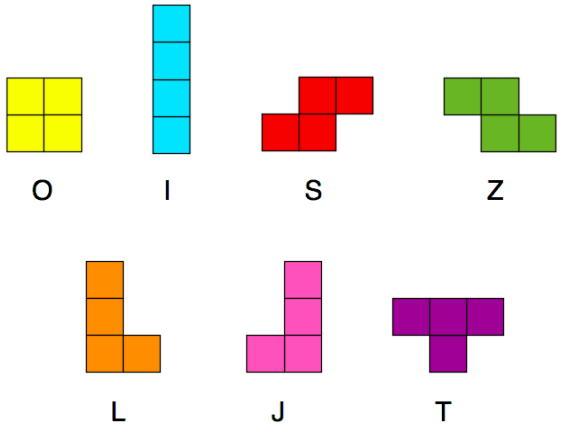
Requirements

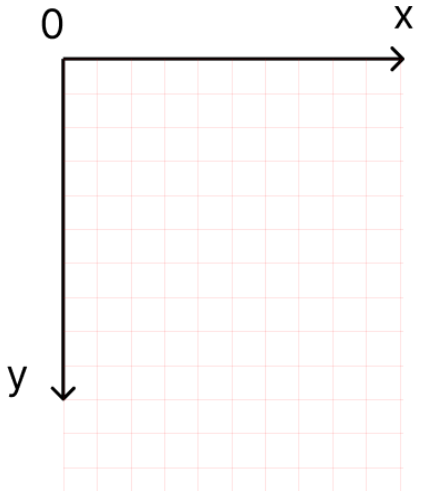
Your task is to make a Tetris game resolver. This Tetris game works like the classic game Tetris, but for simplicity, user inputs (moving pieces) are not possible. So the game rules are:

- The playground is a grid of 10 squares wide and 100 squares tall.
- Each piece enters the grid from the top and falls down to the bottom, one after the other.
- When any row in the grid is fully filled, then that row is removed from the grid, and all cells above it shift down.
- Unlike the real Tetris game, a falling piece cannot be rotated or moved on the horizontal axis. It just falls down from top, until it encounters an obstacle, and stops.

The shapes we'll work with are outlined below:



The grid coordinate system is like this:



Your resolver program will take a text file as input (like the “games.txt” file included). One line in this file represents 1 full game. As an example, a line looks like this: “L2,J4,O1,L6,J8”. Each letter represents a piece, and the digit represents its corresponding starting X coordinate.

Your goal: for each game (each line in the file), your resolver program needs to output the maximum height of the remaining blocks in the grid.

Your program should be invokable from the command line, taking input from STDIN and writing to STDOUT, eg:

$ ./tetris < games.txt > output.txt

You don’t need to validate the input file, there will be no illegal inputs (and no games will create a height beyond 100). For each line in STDIN, only the game height (followed by a new line) should be sent to STDOUT.

Examples Inputs/Outputs:

1. “O0,O2,O4,O6,O8” would output 0 - two rows removed.
2. “L0,J2,O4,L6,J8” would output 2 - one row removed.
3. “T0,Z3,T5,J8” would output 3 - no rows removed.
4. “T0,Z3,T5,O2,O8” would output 3 - one row removed, O2 piece does not continue to fall although empty space is below it.