

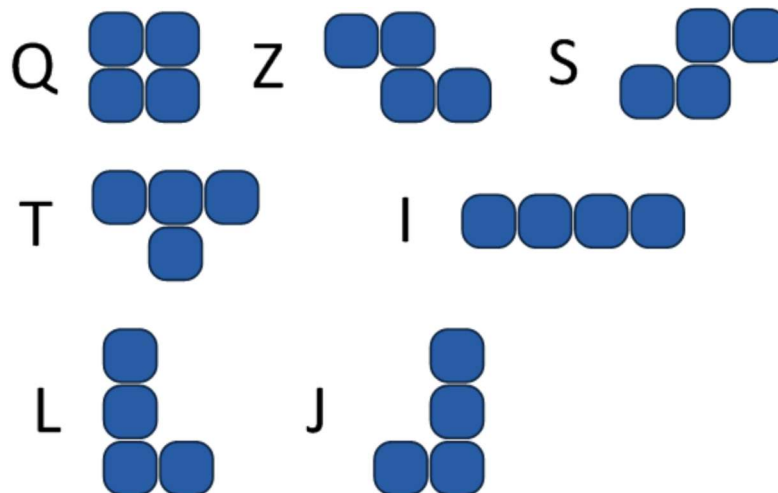
## Coding Challenge

You are to write a simplified Tetris engine.

The engine should model a grid that pieces enter from top and come to rest at the bottom, as if pulled down by gravity. Each piece is made up of four unit squares. No two unit squares can occupy the same space in the grid at the same time. The pieces are rigid, and come to rest as soon as any part of a piece contacts the bottom of the grid or any resting block. As in Tetris, whenever an entire row of the grid is filled, it disappears, and any higher rows drop into the vacated space without any change to the internal pattern of blocks in any row.

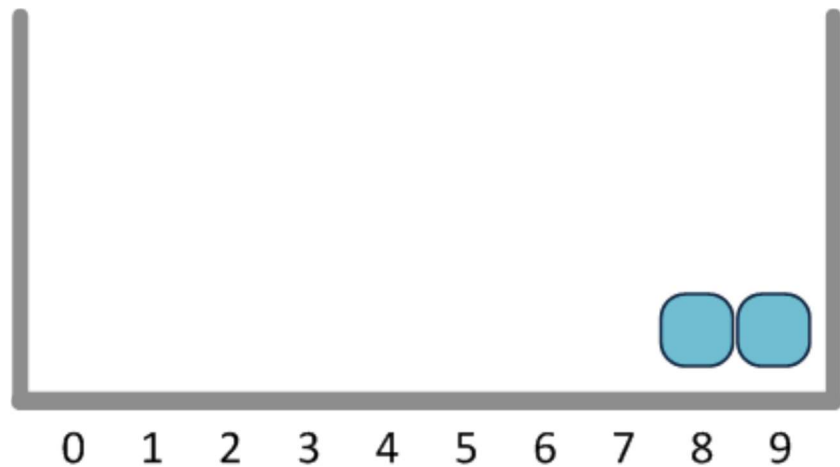
Your program must process a text file of lines each representing a sequence of pieces entering the grid. For each line of the input file, your program should output the resulting height of the remaining blocks within the grid.

The file denotes the different possible shapes by letter. The letters used are Q, Z, S, T, I, L, and J. The shapes of the pieces they represent are shown in the diagram below.

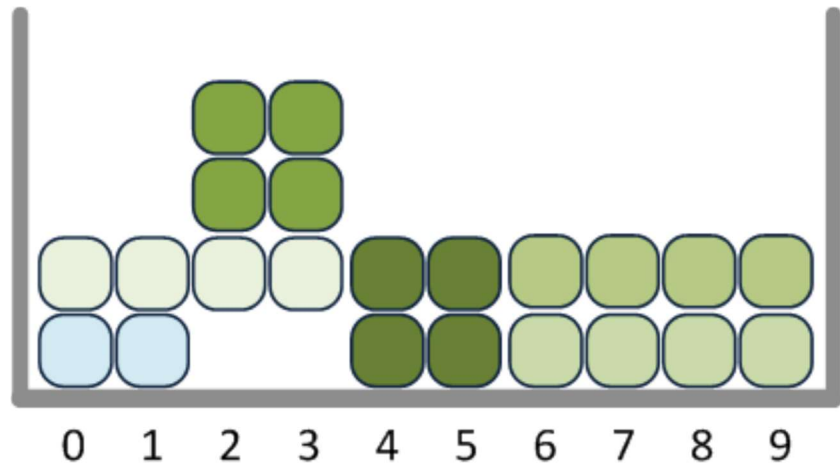


You do not have to account for shape rotation in your model. The pieces will always have the orientations shown above.

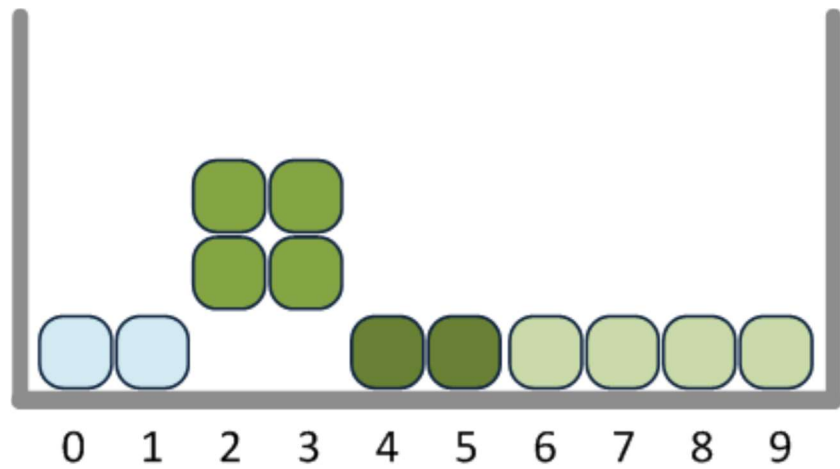
Each line of the input file is a comma-separated list. Each entry in the list is a single letter (from the set above) and a single-digit integer. The integer represents the left-most column of the grid that the shape occupies, starting from zero. The grid of the game space is 10 units wide. Your program need not detect whether any sequence of pieces will exceed any particular height, but you may assume that no test case will result in a height of greater than 100. For each line of the file, the grid's initial state is empty.







The second line clears, and the final result is as follows:



Note that the rows drop as rows, and do not fill gaps in the rows below. So the final output for this test case is "3".