

# 2048

**Time limit:** 2.5s    **Memory limit:** 16M

## ECOO Practice 2014

2048 is a popular single-player online game where the player is given a  $4 \times 4$  grid where they must slide number tiles around to combine them into the 2048 tile. When an arrow key (up, down, left, or right) is pressed, all the tiles on the grid slide in the specified direction. Tiles stop sliding when they hit the edge of the board or a different numbered tile. When two tiles of the same value collide with each other, they fuse into one tile whose value is the sum of the values of the two initial tiles. The following is an example of making a move (pressing the UP arrow key) in an in-progress game of 2048.

2	64	4	32
4	16	8	2
4	32	4	
4	2		2

2	64	4	32
8	16	8	4
4	32	4	
2	2		

Note that there are three 4-tiles in the leftmost column. By pressing UP, all of the tiles try to slide upward. The two topmost 4-tiles in the leftmost column get fused into an 8-tile. The bottommost 4-tile slides up, but cannot get fused anymore. The 2-tile in the bottom-right corner also slides upward and gets fused with the 2-tile above it. It just so happens that a new 2-tile pops into the grid at the bottom-left corner after making this move.

In the real 2048 game, a new tile always pops into somewhere on the grid for every move you make. For the purposes of this problem, no new tiles will appear. Your task is, given the setup of an in-progress game of 2048, determine the value of the largest possible tile you can create by any series of up, down, left, or right slides if no new tiles are introduced to the grid.

## Input Specification

The input will consist of 5 test cases. Each test case will consist of 4 lines, each with 4 columns — a  $4 \times 4$  grid of values. Numbers in the input will be from the set  $\{0, 2, 4, 8, 16, 32, 64, 128, 256, 512, 1024, 2048\}$ . A value of 0 indicates that the position in the grid is unoccupied. The input positions will be such that 2048 is the largest possible value you can attain.

## Output Specification

For each test case, output one integer — the largest tile you can produce by any (possible length-zero) sequence of moves from the given game setup, if no new tiles were to be introduced to the grid.

## Sample Input

---

```
2 64 4 32
8 16 8 4
4 32 4 0
2 2 0 0
0 0 0 2
0 0 0 2
0 0 0 0
0 0 0 2
2 4 8 2
0 2 0 8
0 0 0 0
0 0 0 0
4 32 8 0
2 64 4 2
8 4 2 0
2 0 0 0
256 256 0 0
256 256 0 0
256 256 0 0
256 256 0 0
```

## Sample Output

---

```
128
4
16
128
2048
```

## Explanation

---

For the first case in the sample input (depicted by the second diagram above), the largest obtainable tile assuming no new tiles pop in is the 128-tile. One way to do is through the sequence of moves: right (shift the bottom-leftmost 4-tile under the other 4-tile), up (merge the 4-tiles), left (merge the 8-tiles), left (merge the 16-tiles), up (merge the 32-tiles), up (merge the 64-tiles).

2	64	4	32
8	16	8	4
4	32	4	
2	2		



2	64	4	32
8	16	8	4
	4	32	4
			4

2	64	4	32
8	16	8	8
	4	32	4

2	64	4	32
8	16	16	
4	32	4	

2	64	4	32
8	32		
4	32	4	

2	64	8	32
8	64		
4			

2	128	8	32
8			
4			