ACSL

**American Computer Science League**

#### All-Star #4

**2012 - 2013**

**ACSL Seega**

**PROBLEM:** Seega is an ancient Egyptian game played by 2 players, and for this program, on a 5 x 5 grid. Markers will be X's and O's. All moves are orthogonal - not diagonally. All moves are to an adjacent empty cell. The following rules in priority apply:  
  
 A. A player can capture an opponent's marker by moving his marker to an adjacent  
 empty cell and "sandwiching" the opponent's marker either horizontally or vertically.   
   
 B. A marker is in danger of being captured if it can be sandwiched by the next move of its opponent. To avoid capture, a player will move his marker using the following priorities:  
 1. Move a marker to location 13 if possible, it is safe from capture there  
 2. Move a marker between two opponent's markers. It is not captured. 3. Move the lowest numbered marker in danger of capture to the right or left in that order, if possible.  
 4. Move the lowest numbered marker in danger of capture up or down in that order, if possible.

C. If no markers are in danger of capture, move a marker according to the rules 1 - 4 above minus the "in danger of capture" phrase.   
   
 D. If a player can't move, he loses his turn but his opponent must move so that a move is then available for the player. Use the priorities 1 - 4 above in C.

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 21 | 22 | 23 | 24 | 25 |  |  |  |  |  |  |  |
| 16 | 17 | 18 | 19 | 20 |  | X | O | X |  |  |  |
| 11 | 12 | 13 | 14 | 15 |  |  |  |  |  | O |  |
| 6 | 7 | 8 | 9 | 10 |  |  |  |  |  | X |  |
| 1 | 2 | 3 | 4 | 5 |  |  |  |  |  | O |  |

The grid on the left above shows how the grid squares are numbered. The grid on right shows examples of a marker sandwiched by its opponent.

**INPUT:** There will be 5 lines of input. The first integer on each line will give the number of X's and O's on the grid. There will always be the same number of each. This will be followed by the location numbers of the X's and then the location numbers of the O's.

**OUTPUT:** There will be 10 lines of output. For each line of input print the ending location of the move made by X and then using that same original data print the ending location of the move made by O. The priority of the move is to capture first and then, if capture is not possible, to avoid capture and then to make a not in danger move. If more than one marker can be moved to satisfy the priority, move the lowest numbered of those markers. If an X marker can't be moved print LOSE A TURN. Then an O marker must be moved using priorities 1 - 4 above so that X can move.

The situation where a marker can be captured by 2 different moves is not tested.

SAMPLE INPUT SAMPLE OUTPUT

1. 3, 1, 8, 21, 2, 5, 9 1. 3  
2. 3, 1, 2, 6, 3, 7, 11 2. 7  
3. 2, 7, 10, 8, 11 3. LOSE A TURN  
4. 2, 8, 18, 7, 9 4. 4  
5. 2, 7, 14, 1, 8 5. 9  
 6. 6  
 7. 13  
 8. 6  
 9. 9  
 10. 6

ACSL

**American Computer Science League**

#### All-Star #4

**2012 - 2013**

ACSL Seega

TEST DATA

TEST INPUT TEST OUTPUT

1. 2, 16, 23, 17, 14 1. 18  
2. 2, 7, 10, 8, 13 2. 18  
3. 3, 20, 24, 25, 23, 19, 15 3. 9  
4. 3, 7, 14, 22, 2, 13, 18 4. 9  
5. 3, 3, 7, 10, 8, 1, 5 5. LOSE A TURN  
 6. 14  
 7. 12  
 8. 12  
 9. 9  
 10. 6