📖 **Rule.en.md - Grip**

JavaChallenge2015

Online Execution Server

http://arena.ai-comp.net:3333/

Overview

Drop the block, and drop the enemies!

Rules

Move on a board composed of 6 x 6 blocks, and drop the blocks in the direction you are facing. Players can drop the enemy players with a block. The last remaining player is a winner.

Initial Settings

There are four players. Players get their each id (0, 1, 2, 3). Their initial position and direction are decided randomly. They are placed so that the Manhattan distance between each player are more than 3.

Board

The board is composed of 18 x 18 squares. To be exact, there are 6 x 6 blocks on the board, and each block is composed of 3 x 3 squares.

Player

A player exists on 1 x 1 square, facing either up, down, left, or right.
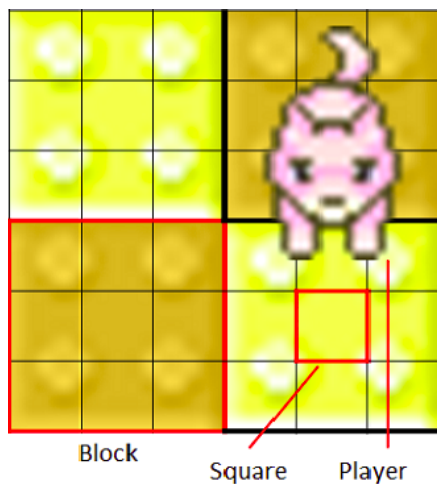
Flow of Game

Game starts from the player whose id is 0, and continues by turn as ID0->ID1->ID2->ID3->ID0->...

At the beginning of each of your turn, you get current information of each player and the board. You choose one action from the following.

- Move to the upper/lower/right/left square
- Attack toward your facing direction
- Do nothing

Move

Move to the upper/lower/right/left **square** of your current position. Then, you face the direction you have moved to.

Cancel of Move

In the following cases, move is canceled and only the direction is changed.

- Move to the outside of the board
- Move to a block already dropped
- Move to a square whose Manhattan distance is 3 or less from another player.

Attack

When you attack, blocks in the direction you are facing drop some turns later. A block at n blocks away drops 4n turns later. Therefore, the neighbor block drops in your next turn (4 turns later), and the neighbor of the neighbor block drops in your second next turn (8 turns later), and so on.

Even if a player attacks fallen blocks or blocks which will be fallen, the leftover turns which are required for blocks to fall or restore are never changed.

Additionally, if you attack, you cannot take an action until your second next turn (8 turns later) ends.

Drop of a Player

If there is a player on a block which has dropped, that player drops with the block and is removed from the board.

Restoration of a Block

A dropped block is restored to its original state in the your fifth next turn (20 turns later). A dropped player will not be restored.

End of Game

If you make the other players drop, you win. If there are multiple players after the end of the 1000th turn, the game ends in a draw.

Input and Output Format of an AI Program

Output Format of a Ready Message

When your AI becomes ready to start a game, print READY to the standard output. If a ready message is not printed within **1 second**, the AI program will be terminated.

Input Format of Game Settings

```
id
T
B_{0,0} B_{0,1} B_{0,2} ... B_{0,5}
```

```
B₁,₀ B₁,₁ B₁,₂ ... B₁,₅
...
B₅,₀ B₅,₁ B₅,₂ ... B₅,₅
R₀ C₀ D₀ S₀
R₁ C₁ D₁ S₁
R₂ C₂ D₂ S₂
R₃ C₃ D₃ S₃
EOD
```

- id: player ID
- T: current turn
- $B_{r,c}$: status of a block in the _r_th row and _c_th column
  - 0 means the square is stable and is not going to drop
  - positive number means the square is going to fall that number turns later
  - negative number means the square has been dropped, and is restored that number's absolute value turns later
- R_i: row of the position of the player whose ID is i
- C_i: column of the position of the player whose ID is i
- D_i: direction of the player whose ID is i
- S_i: the number of turns the player whose ID is i will be able to take an action. 0 means that player can take an action this turn.
- EOD: end of input

Output Format of an Action

Print one of the following commands. If your AI program does not print its action within 1 second from the beginning of the turn, it will be terminated.

- "U": move to the upper square
- "R": move to the right square
- "D": move to the lower square
- "L": move to the left square
- "A": attack toward your facing direction
- "N": do nothing

Notice

When you output READY at the start of the game or an action in each turn, be sure to output a newline character ("\n") at the end of the line and flush the standard output after printing.