



Team members:

Mingzhen Ao

U6527600

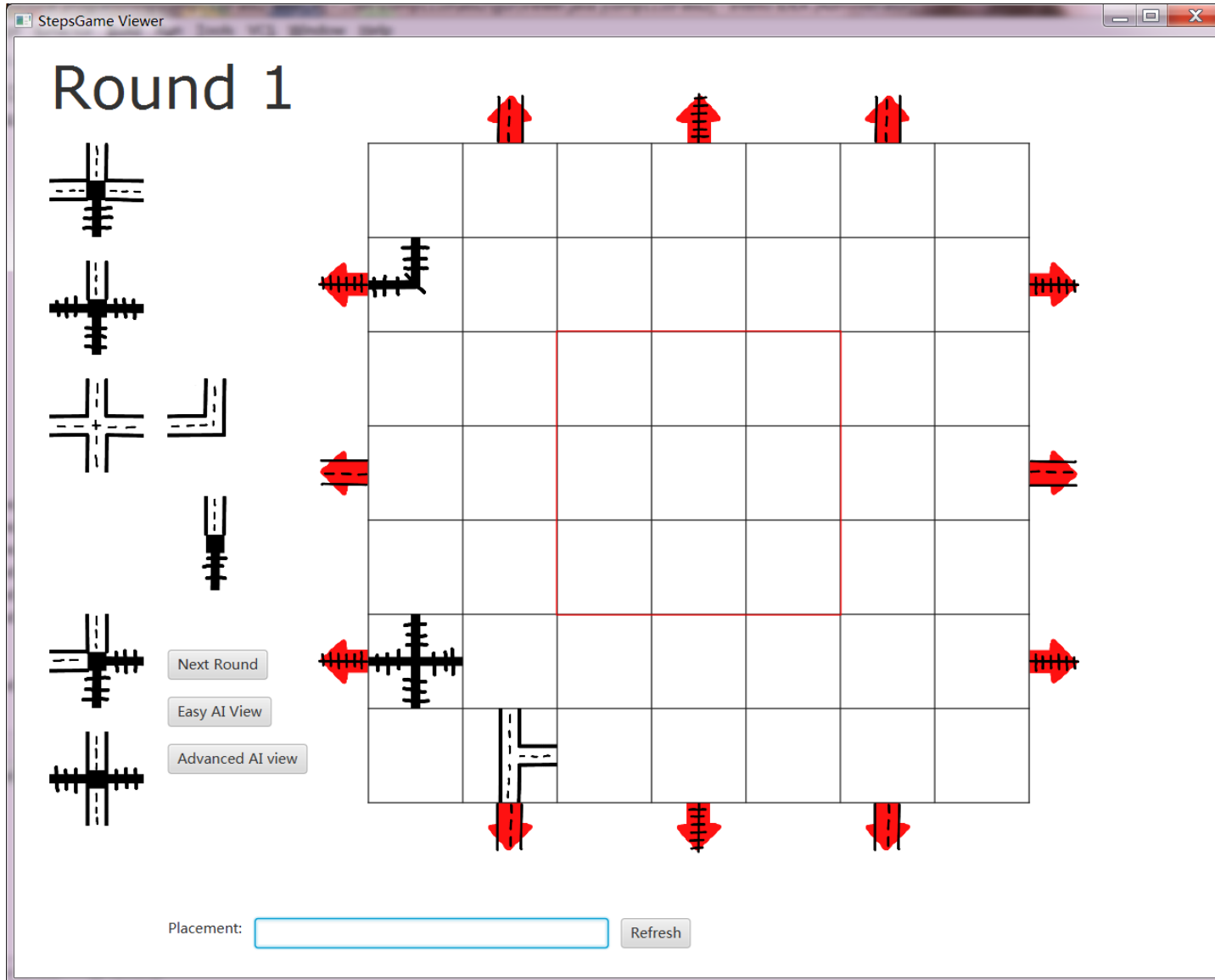
Yusen Wei

U6736425

Zixin Ye

U6605822

JavaFX: Dragging



- Dragging:
- Use `setOnMouseDragged` to drag `imageView`
- Set the figure of X, Y to stay centered
- Use `setOnScroll` to rotate the tile when scroll
- Snapping the tile to nearest possible slot if valid
- Use `setPickOnBounds(true)` to increase the dragging flexibilities
- Any placed tiles will be locked in the next round using `setDisable(true)`



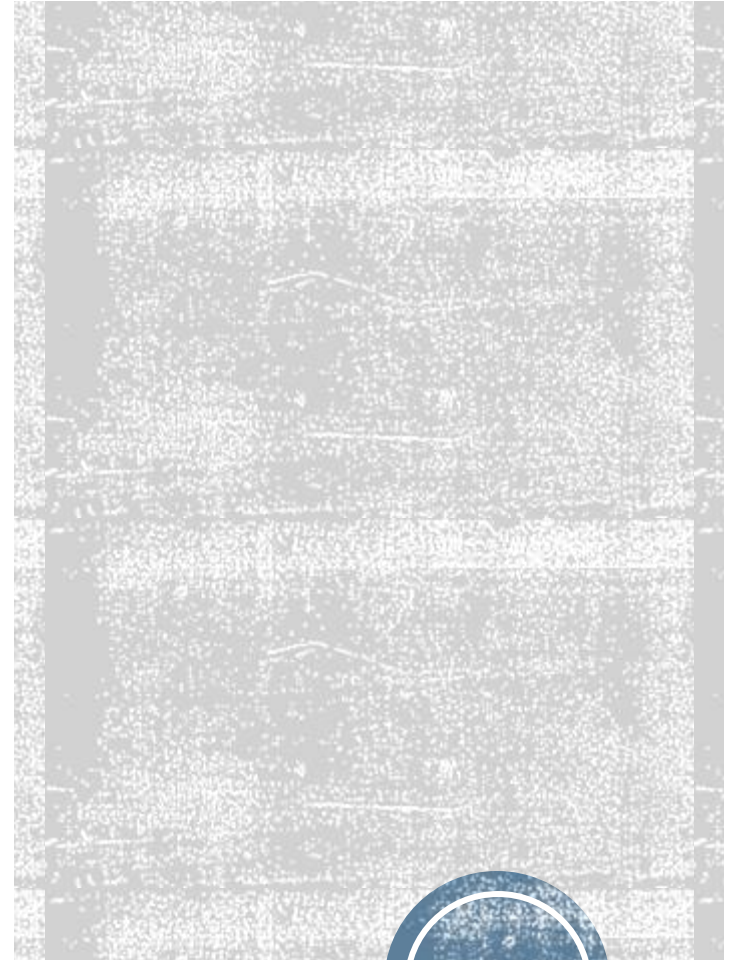
JavaFX: Getting boardString

- Set the parameter boardString
- Update the boardString every time the left mouse button is released using `setOnMouseReleased`
- If the boardString is not valid, entering next round will be forbidden
- Clear any unused normal tiles in the next round `setVisible(false)`
- End the game if finished round 8



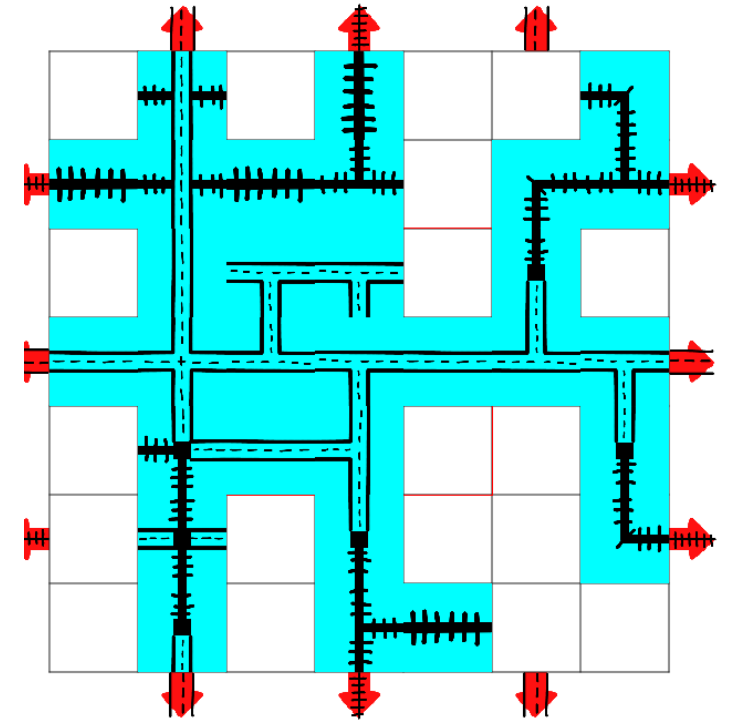
1.SCORES CALCULATES

2.COMPUTER PLAYERS



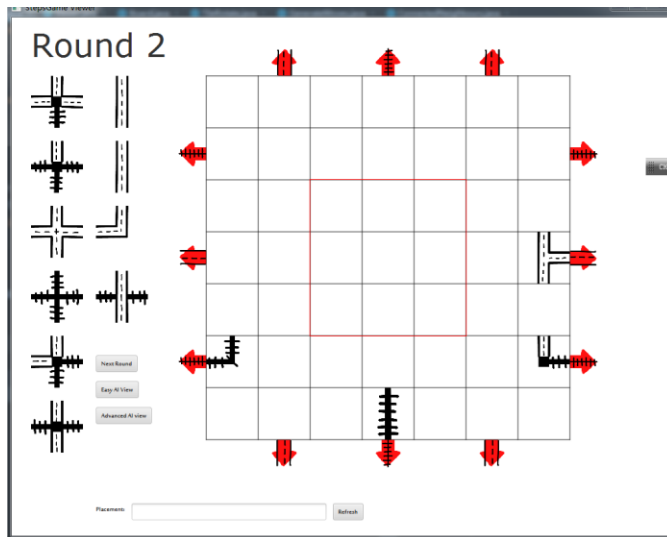
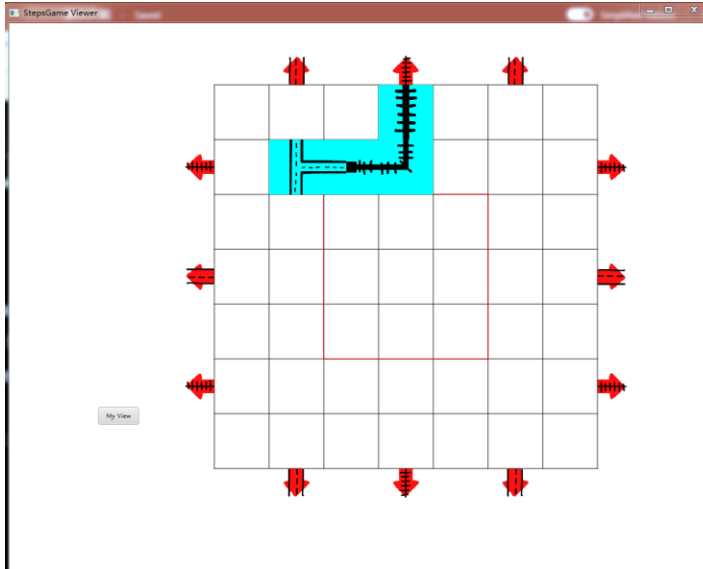
Scores calculates

- 1.centrail tiles part
- Divided string into string arrays which length 5 each.
- Check each String's third and forth characters which in the arrays.
- 2.exits numbers
- Firstly pick up each lines.
- Calculate each lines exits numbers then add them all.
- 3.dead tiles.
- Key point is handle B2.
- reversed thinking
- Each string arrays minus 4.
- If one direction no connector add one.
- If connect with edges add one.
- If there are connect neighbors add one.



COMPUTER PLAYERS

- For easy AI
 - Firstly create another board.
 - Just use task 10 to judge the positions then put them in the board.
 - When human finished one time , repeat last step again.
- For advanced AI
 - Basis on task 10 , we add S .

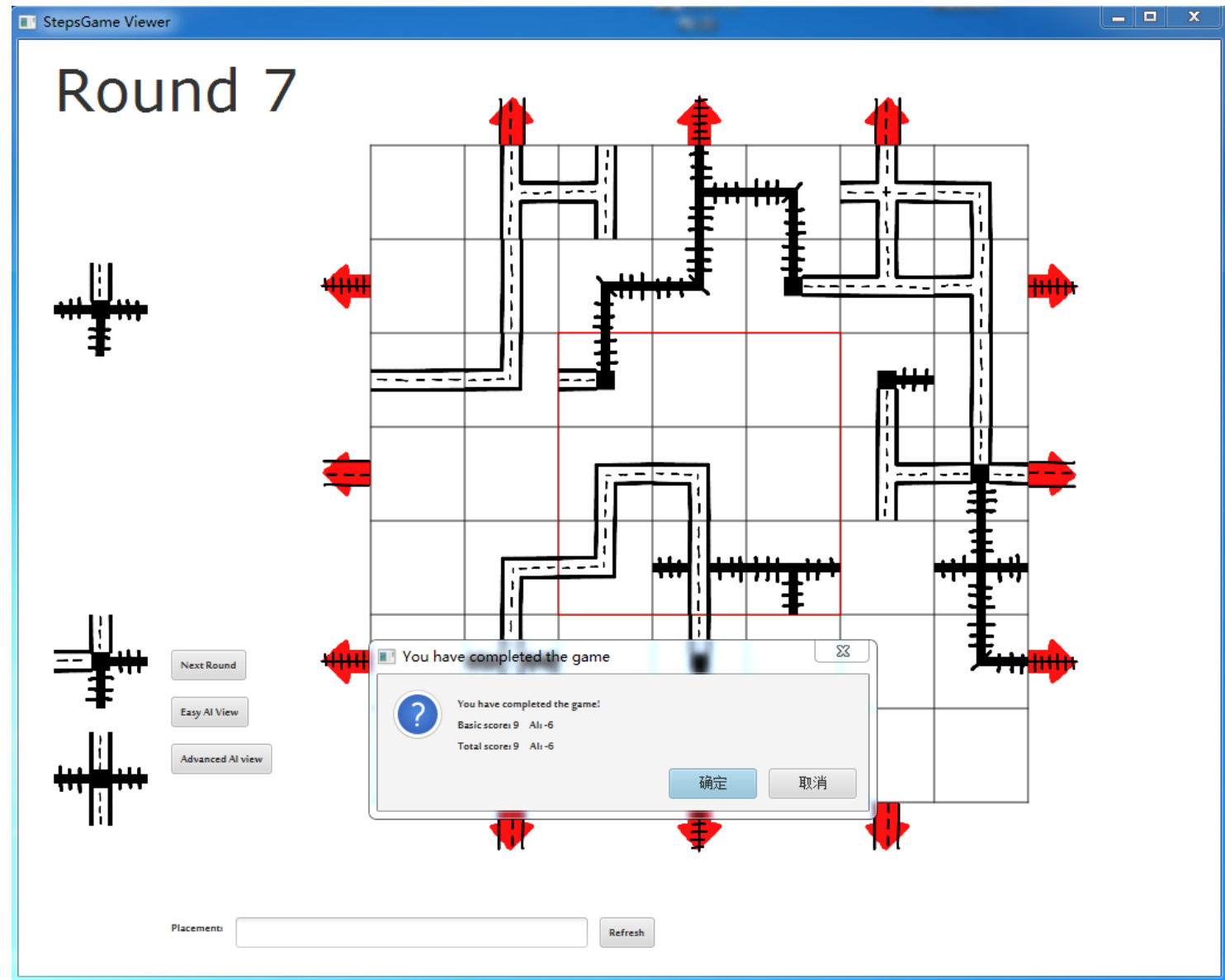


Generate Move

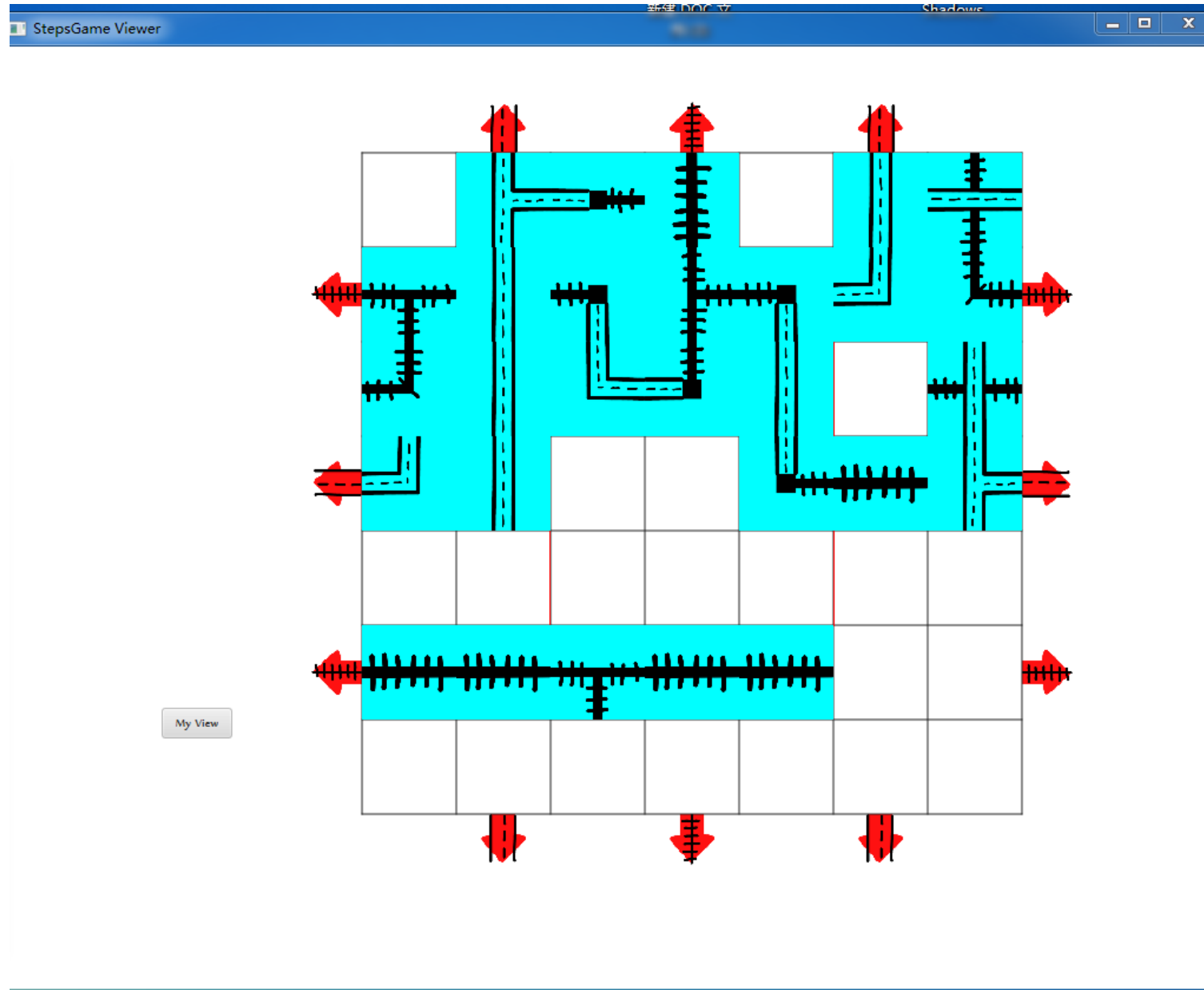
- 1. Gets all unoccupied grids based on the boardString that is passed in.
- 2. According to the combination of each of the Dice with unoccupied Spaces respectively, obtain all potential placement strings (like A1B20) of the four Dices.
- 3. Create a method that returns the first placementString that when it concatenates boardString, the boardString can still be legal.
- 4. Get all combinations of placement strings returned by different placement orders for all four dices. Because, in some conditions, you can place a tile only after you have placed some other tiles.
- 5. Get the longest combination of placement strings, and we only need to return one of them. The longest one must be true, as it avoids the problem I just mentioned (in step 4).



The final result



Easy one



Difficult one

