IQ-FIT

Smart game

Ass2-Tue15J

Chensheng Zhang	U6615215
Paul Won	U7158520
Yuxuan Yang	U7078049

1. Summary

What we have done

Basic tasks

- Completed all tasks (task1 – task11)

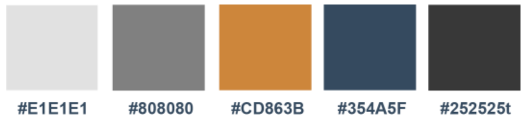
Interesting features

- Implemented “autosave, load and delete” games
- Introduce reward mechanism, use “stars” to represent the achievement
- Players can earn “stars” by completing challenges
- Players can use “stars” to unlock the unknown challenge
- Players can use “stars” to get a hint of solutions.
- Exploring interesting challenge will get extra “stars”
- Press “SPACE” to tidy pieces up
- A preview (rotated shape/flipped shape) of selected piece

2. Design approaches

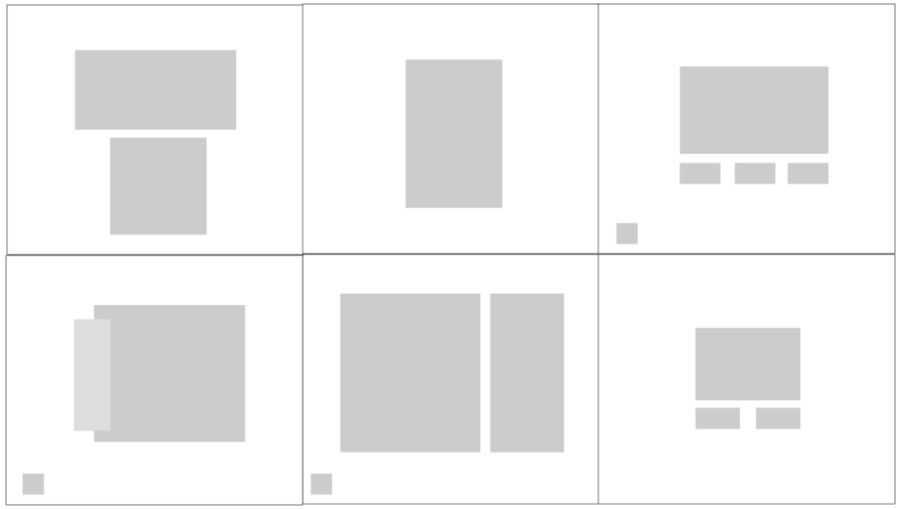
GUI design

- **Color palette**



- **Elements**

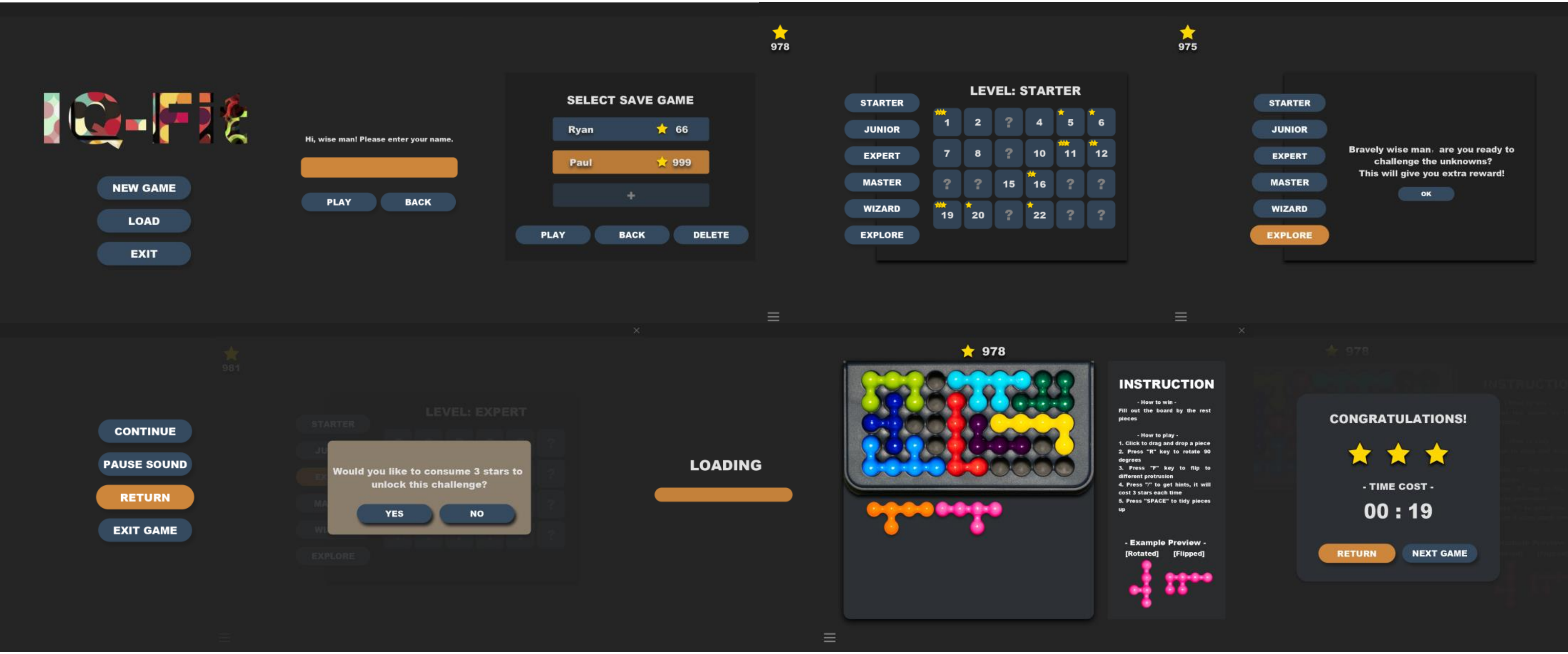
- **Layout**



2. Design approaches

GUI design

- **Screen shot**



Hi, wise man! Please enter your name.

Bravely wise man, are you ready to challenge the unknowns? This will give you extra reward!

Would you like to consume 3 stars to unlock this challenge?

- How to win -
Fill out the board by the rest pieces

- How to play -
1. Click to drag and drop a piece
2. Press "R" key to rotate 90 degrees
3. Press "F" key to flip to different protrusion
4. Press "?" to get hints, it will cost 3 stars each time
5. Press "SPACE" to tidy pieces up

2. Design approaches

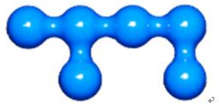
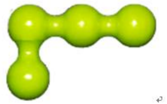
Program design

- **Game Translation**

We consider the whole game as a big matrix, and the board is the matrix with five rows and ten columns as following:

```
[0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
[0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
[0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
[0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
[0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
```

Each piece is regarded as a matrix with two rows and three/four columns. Some examples are as followed:

	[1, 1, 1, 1] [0, 1, 0, 1]
	[1, 1, 1] [1, 0, 0]

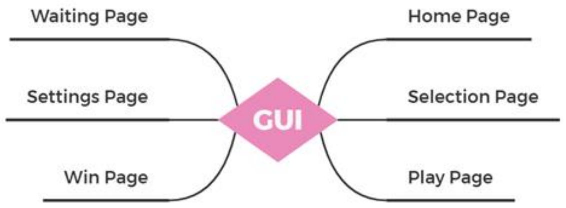
2. Design approaches

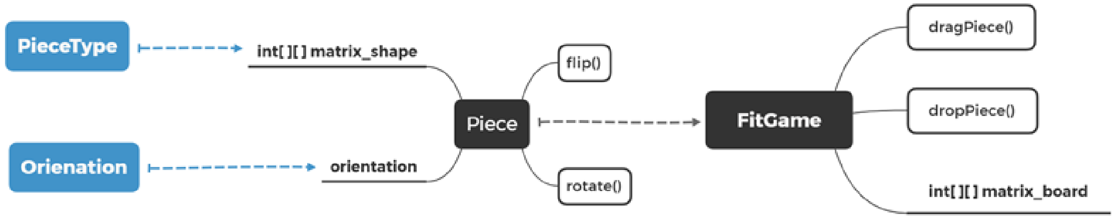
Program design

- **Modular Design**

We individually designed different pages of the GUI, and each page is well-functional.

As for the main game program, we also separated pieces and the board, and we used the enumerated type to record the matrix shape and orientation for each piece



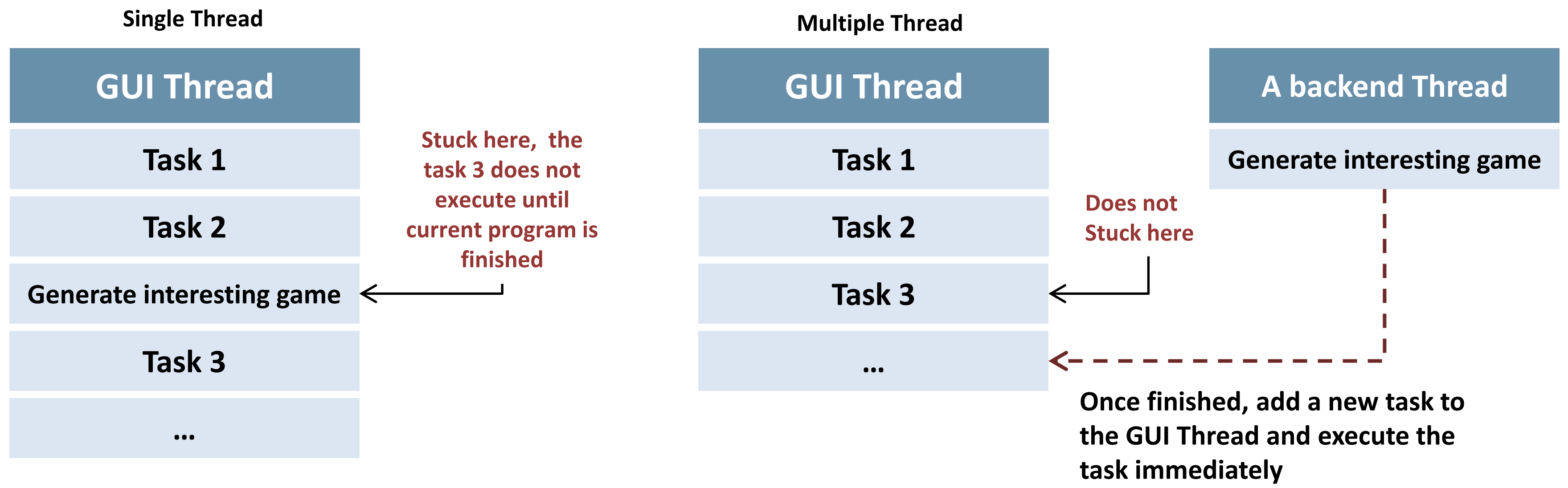


2. Design approaches

Program design

- **Multiple Threads**

In order to make GUI update more fluently, we created a new thread undertake the backend jobs. For example, generating an interesting game (task11) needs around 2s, and if we put the code into the GUI thread, the GUI will get stuck. To solve it, we put the generation task to a new thread.


Single Thread
GUI Thread
Task 1
Task 2
Generate interesting game
Task 3
...
Stuck here, the task 3 does not execute until current program is finished
Multiple Thread
GUI Thread
Task 1
Task 2
Task 3
...
Does not Stuck here
A backend Thread
Generate interesting game
Once finished, add a new task to the GUI Thread and execute the task immediately

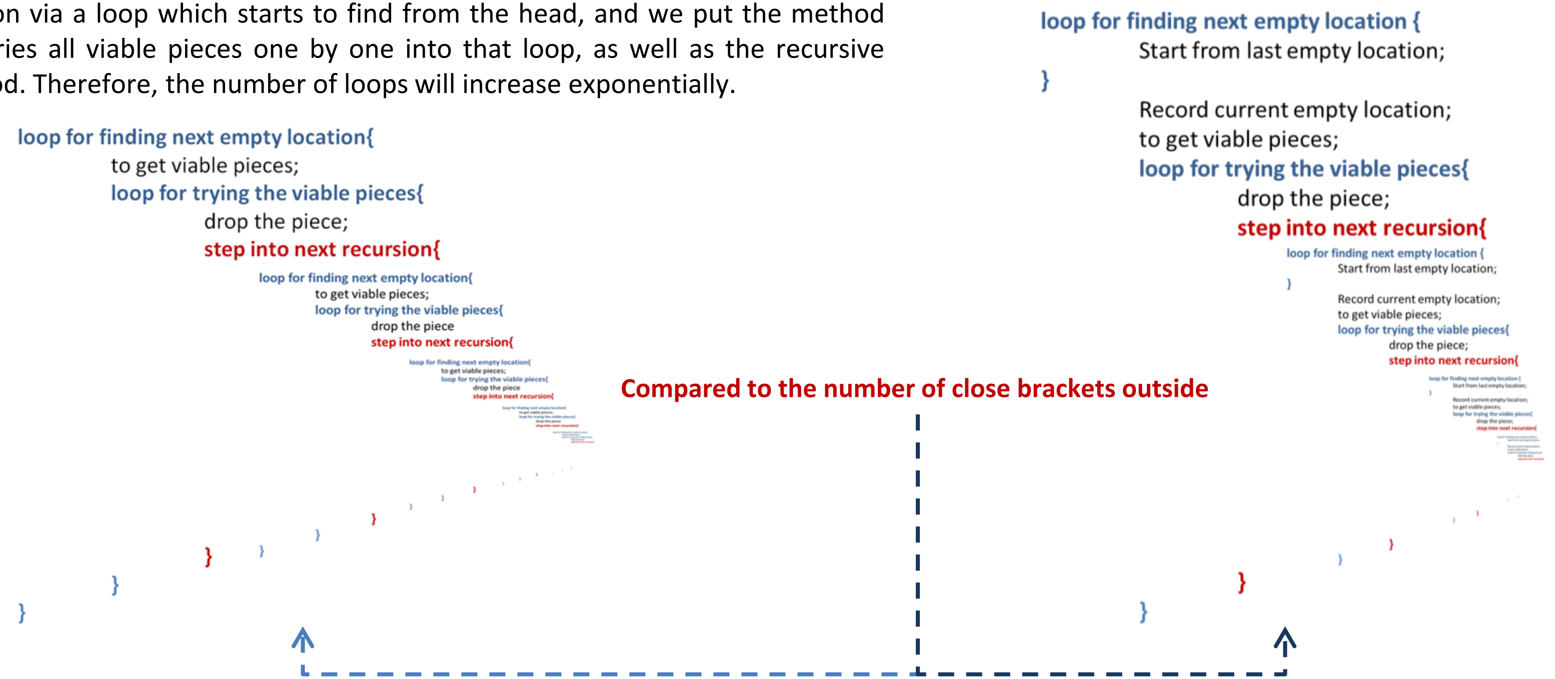
3. Problems we looked at

- **Task9**

We used the depth-first searching to find a solution. At first, it couldn't pass the expert, master and wizard tests and even cost hundreds million second to find the solution for a starter level of games.

The problem is that in each next layer of recursion, we get the next empty location via a loop which starts to find from the head, and we put the method that tries all viable pieces one by one into that loop, as well as the recursive method. Therefore, the number of loops will increase exponentially.

To solve that, we not only put the loop (try all viable pieces and step to next location) outside the loop (find next empty location), but also recorded the last empty position, so that we didn't need to find the next empty location from the head of the board.



```
loop for finding next empty location{
    to get viable pieces;
    loop for trying the viable pieces{
        drop the piece;
        step into next recursion{
            loop for finding next empty location{
                to get viable pieces;
                loop for trying the viable pieces{
                    drop the piece
                    step into next recursion{
                        loop for finding next empty location{
                            to get viable pieces;
                            loop for trying the viable pieces{
                                drop the piece
                                step into next recursion{
                                ...
                                }
                            }
                        }
                    }
                }
            }
        }
    }
}
```

```
loop for finding next empty location {
        Start from last empty location;
}
        Record current empty location;
        to get viable pieces;
        loop for trying the viable pieces{
                drop the piece;
                step into next recursion{
                    loop for finding next empty location {
                            Start from last empty location;
                    }
                            Record current empty location;
                            to get viable pieces;
                            loop for trying the viable pieces{
                                drop the piece;
                                step into next recursion{
                                ...
                                }
                            }
                }
        }
```

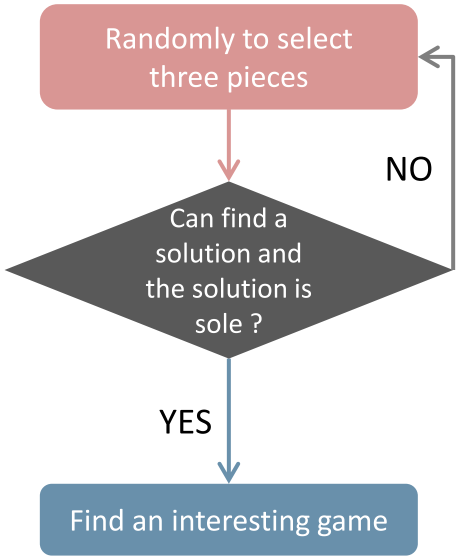
Compared to the number of close brackets outside

3. Problems we looked at

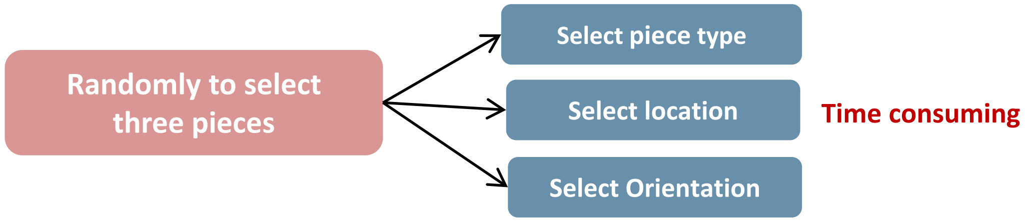
- **Task11**

Having solved the task9, implementing task11 is a bit easy. Our idea is that we randomly generate several pieces and place them on the board as our new challenge and try to find the solution, if there is only one solution and that solution is not shown in exist game, then this challenge is available.

The whole process


Randomly to select three pieces
NO
Can find a solution and the solution is sole ?
YES
Find an interesting game

- The problem is that randomly selecting valid pieces sometimes is time consuming. Because a piece involves piece type, location, and orientation. For example, a piece named "B23N", and the time consuming part is selecting the location

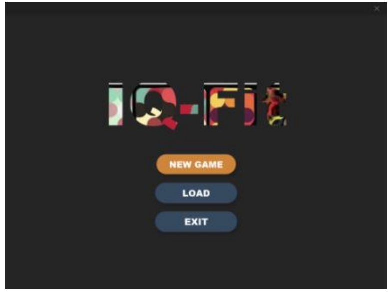


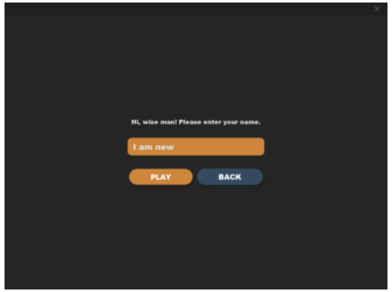
To solve it, we set a limitation on the times of random, if the times of selecting the location is large than 30, then restart to select a piece

Let us to have a look on interesting features

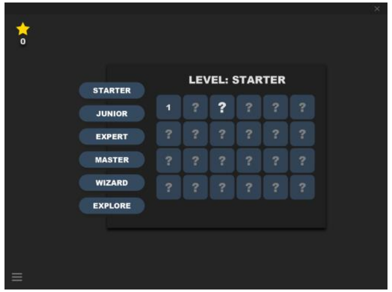
Autosave, Load and delete game

New a game and autosave

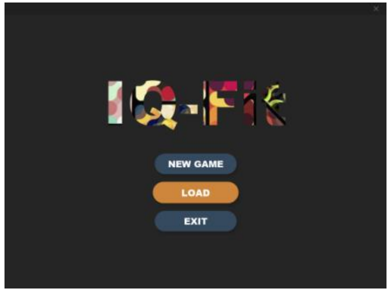


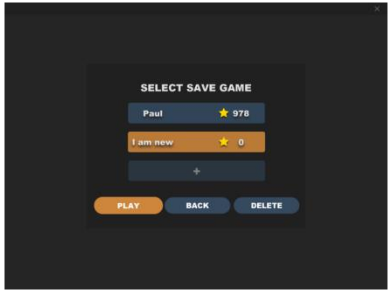

Hi, wise man! Please enter your name.

I am new

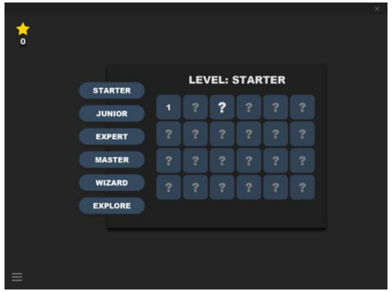


Load a saved game

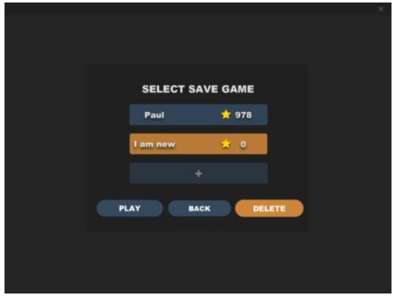



SELECT SAVE GAME

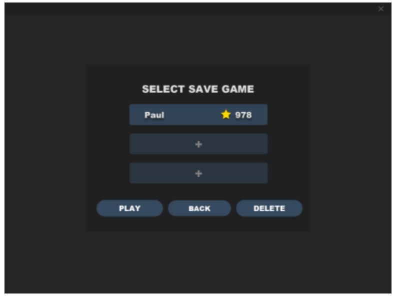
Paul	★ 978
I am new	★ 0
+	



Delete a saved game


SELECT SAVE GAME

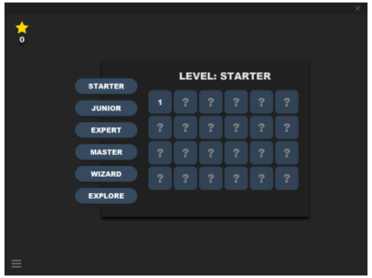
Paul	★ 978
I am new	★ 0
+	

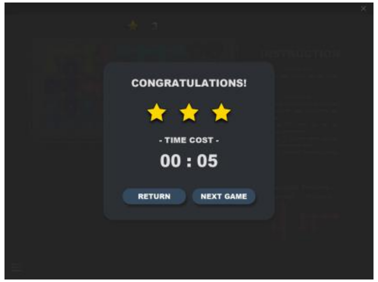

SELECT SAVE GAME

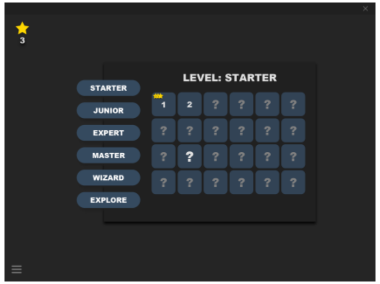
Paul	★ 978
+	
+	

Stars

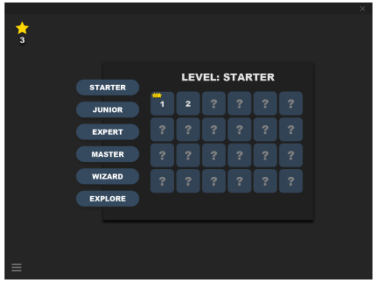
Earn “stars” by completing challenges

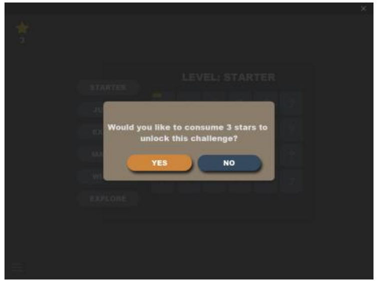


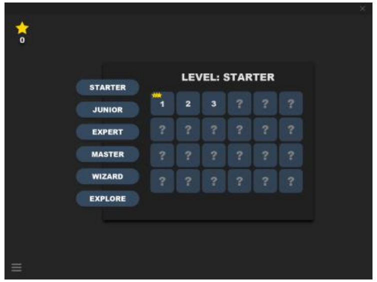




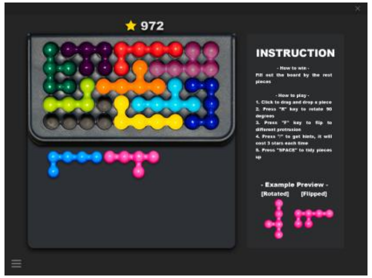
Use “stars” to unlock the unknown challenges




Would you like to consume 3 stars to unlock this challenge?



use “stars” to get a hint of solutions



INSTRUCTION

- How to win -

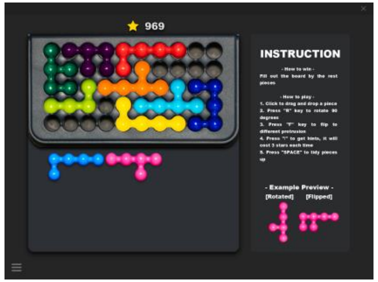
Fill out the board by the rest pieces

- How to play -

1. Click to drag and drop a piece
2. Press "R" key to rotate 90 degrees
3. Press "F" key to flip to different protrusion
4. Press "I" to get hints, it will cost 3 stars each time
5. Press "SPACE" to tidy pieces up

- Example Preview -

[Rotated] [Flipped]



INSTRUCTION

- How to win -

Fill out the board by the rest pieces

- How to play -

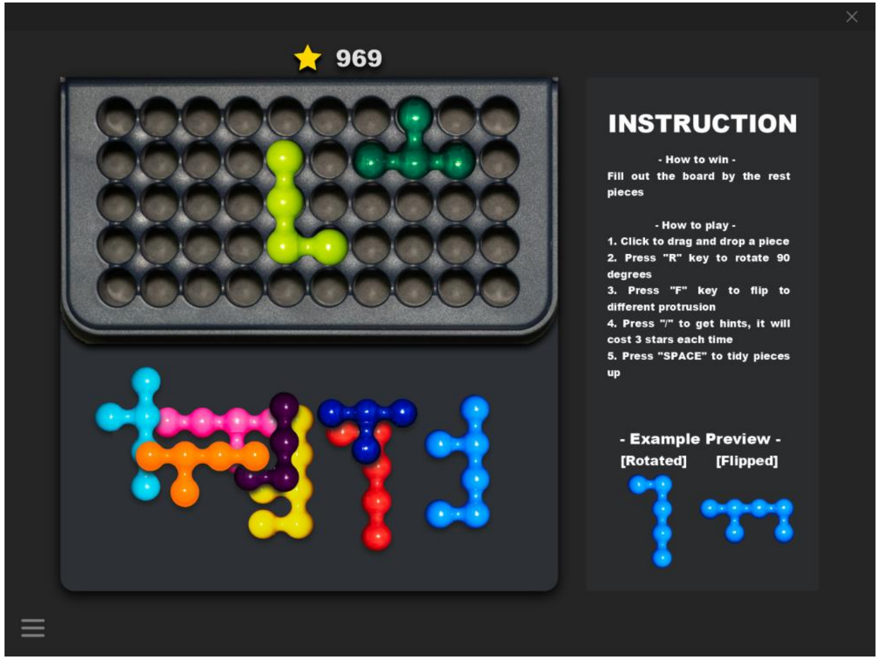
1. Click to drag and drop a piece
2. Press "R" key to rotate 90 degrees
3. Press "F" key to flip to different protrusion
4. Press "I" to get hints, it will cost 3 stars each time
5. Press "SPACE" to tidy pieces up

- Example Preview -

[Rotated] [Flipped]

Tidy pieces up

To press "SPACE" key to tidy pieces up



969

INSTRUCTION

- How to win -

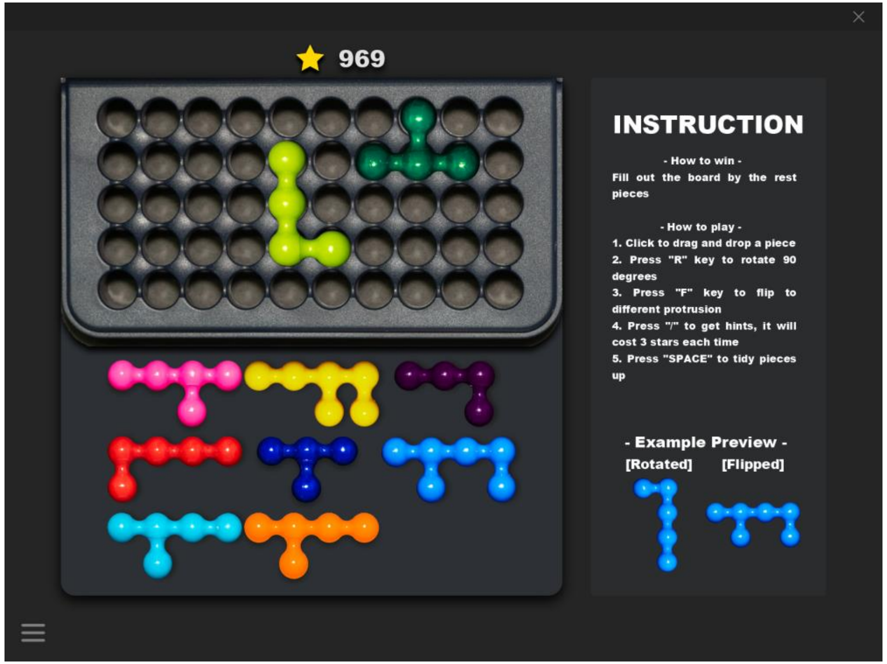
Fill out the board by the rest pieces

- How to play -

1. Click to drag and drop a piece
2. Press "R" key to rotate 90 degrees
3. Press "F" key to flip to different protrusion
4. Press "/" to get hints, it will cost 3 stars each time
5. Press "SPACE" to tidy pieces up

- Example Preview -

[Rotated] [Flipped]



969

INSTRUCTION

- How to win -

Fill out the board by the rest pieces

- How to play -

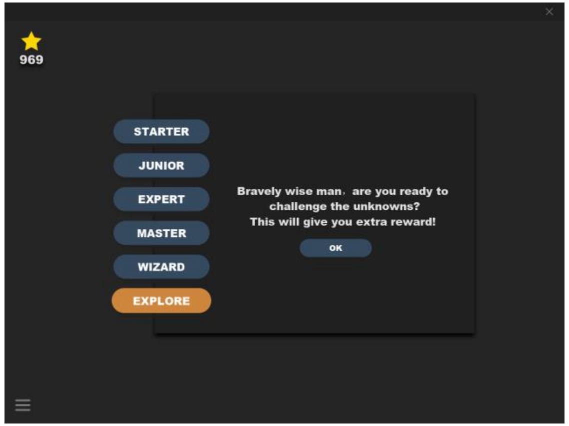
1. Click to drag and drop a piece
2. Press "R" key to rotate 90 degrees
3. Press "F" key to flip to different protrusion
4. Press "/" to get hints, it will cost 3 stars each time
5. Press "SPACE" to tidy pieces up

- Example Preview -

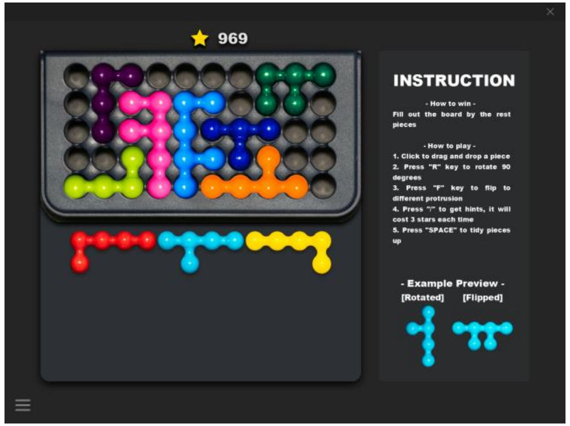
[Rotated] [Flipped]

Explore interesting games

Explore the interesting game (This game does not exist in Starter, Junior, Expert, Master , and Wizard level)



Bravely wise man, are you ready to challenge the unknowns?
This will give you extra reward!



INSTRUCTION

- How to win -

Fill out the board by the rest pieces

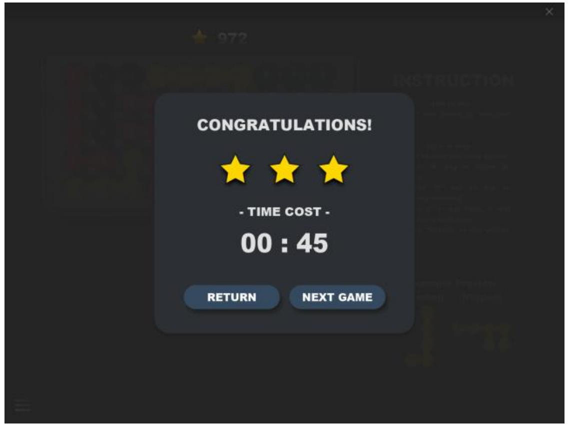
- How to play -

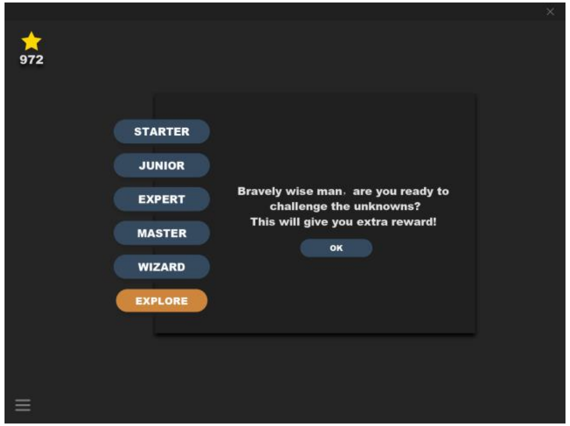
1. Click to drag and drop a piece
2. Press "R" key to rotate 90 degrees
3. Press "F" key to flip to different protrusion
4. Press "/" to get hints, it will cost 3 stars each time
5. Press "SPACE" to tidy pieces up

- Example Preview -

[Rotated] [Flipped]

Gain extra stars reward





Bravely wise man, are you ready to challenge the unknowns?
This will give you extra reward!

Thank you

Enjoy the game!