Division of labor

Hugo did the Checker.java While Thibault and Thomas started to develop the main part. Once Hugo finished the Checker.java, he joined the others on the development on the main part. We worked all together throughout the whole project because we already all worked together before. We kind of pair programed but between the three of us. We brainstormed together and coded together until we had a finished project.

Design

We took a board object in the queue, test each move possible per block (up, right, down, left), if a move is possible we do the move, check if the puzzle and solved and if not we store the new state of the tray in the queue. if the configuration of the tray has been already seen, the pass to the next board on the queue. We do that until we solve the puzzle or the queue is empty.

The storage each move done on a board inside the board class, so when we find the board which solved the puzzle, we just have to display all the move storage in the ArrayList.

For solve the puzzle with the min move possible, we use the board and the possible move like a tree, we go through floor by floor.

Experimental Results

It workin g // fqire du bullshit

Program Development

- we coded first the checker and after the solver. First we did the solver with recursion but it took too much time and we didn’t find the solution with the less move possible.

- because it was logical

- with the files given, mostly the hard files.

Disclaimers

No Bugs

Improvements

For Solve the puzzle with less move we find to oriente the block which can solve the puzzle in the direction of them.