Elifnur Kabalcı

1801042617

- ➔ I defined almost all functions and elements in the boardgame2d class. Since calling the derived class from the base class will cause errors, I used the type statement.
- ➔ Except for the init function, I didn't use a derived class element.
- ➔ I have defined all the necessary functions in the boardgame.h file.

- ➔ In the test.cpp file, I tested all situations. For the arrengment, I use the global functions.
- ➔ I described the all functions in boargame.cpp file.

- ➔ I defined initialize function for each derived class because the table structure of each is different.
- ➔ I did initialize manually in klotski init. I used the choc ho type posted in the assignment pdf. When I searched to get to know the game, there were easier ones, but I used this one, thinking it was the original.
- ➔ I wrote 8 numbers that I will use in Eightpuz init into a vector and wrote a function that mixes the. This function consists of rand and swap. See. Line395 in boardgame.cpp file. I then manually defined the interfering elements. Because when I used for loops it was meaninglessly not assigning correctly.
- ➔ While playing peg game, I didn't fully understand whether it is the 2nd table in the other assignment or the two tables named English and European on the wiki page sent to us. So I made both available by asking the user.
- ➔ In the common initialize function, I made the size of the vector and filling it with space for editing purposes.

- ➔ I called the boardscore function inside the endgame func. To run it after the game is over. I set up a system where the remaining pegs gor the first game are 0 if they finish for the second game and if they do.
- ➔ Since peggame in control will take a long time in the engame func. , I designed a function called controlofpeggame and used the function I used in other assignments. In others, I manually defined the position or number value order they should reach.
- ➔ I designed the playautoll function as a computer play control point. It called playauto for each step.
- ➔ I printed the output checked in playauto to the screen and had the numbers assigned as rand checked according to the game type.
- ➔ I called the same play func. In both computer game and human game. Which allowed me to test at the same time.

- ➔ The parameteterless playuser func. Is the control point of the user game. It calss the other playuser func. With the parameter it receives from the users at every step. In this playuser, the string is parsed and sent to funcstions to play.
- ➔ The most complicated part of the code is playpeg, playpuzi playklot.
- ➔ The playpeg part was set up with the same logic as 3rd assignment.
- ➔ In playpuz, I designed a structure that is processed after checking the direction and table exceeding.
- ➔ I had a hard time with the playklot part. I have mistakes, It's the hardest part to describe and play. Because I couldn't figure out the strategy of playing blocks larger than 1x1. Accordingly, I thought that if we want to process from any point belonging to that block, it should cause a change in the entire block.
- ➔ All my games made with user login are playing and running smoothly, but the computer cannot find the right steps due to the excess of options in the games played. I dint know, maybe its because I didn't expect too much.
- ➔ Due to time constraints, I could not include an explanations with ss's in the report. For the same reasons, some of my mistakes did not grow.