

Risk Analysis

List of risks

1. Input risk. When selecting in the starting menu, the inputs are all integers in the program, and if inputting another type, the program may throw some exceptions. For example, the integer inputs are scanned by the `Scanner.nextInt()`, but if the input is a string or other types, the program will throw the exception. And if the input is not a diamond, the program may run in errors. For example, at the beginning menu, there are 1, 2, 3 three diamonds, and if the player inputs other diamonds, the program may occur errors.
2. Data risk. The guessing words and the saving record will be saved in a file, and if the players see the file's content, they can know the words and can win every time, and they can even edit the record data, so that they can modify the results. This risk may be a big problem.

Strategies

1. Input risk. To solve the exceptions, we can use 'try...catch' to catch the problem, and deal with the mistakes so that the problem can continue running. And to deal with the diamond risks, we can compare the inputting diamond to the designed diamonds, and also think about the logic, for example, if there is record, players cannot load the game. If the input is not in the designed diamonds, the program will show the mistakes and continue running.
2. Data risk. There may be two methods to solve the risks. One is to put the data file in the server rather than on the local directory, and read the data file the inner system, but I think this way is a little complex, maybe in the future, the program can read data in this way. And another method is to save and load encrypt data, this way is easier and also can solve the risk. If the players modify the data file by themselves, they don't know the encrypt key, so the data is not correct, and the program can let the players failed.