

Reflection

Motivations for The Program Design

The design of project classes is intuitive. In a game, there are several players, each of whom controls his dices, rolls the dices in turn and chooses the dices. Easily, we can design three main types: *Game*, *Player* and *Dice*. In the subsequent implementation, I found that it was not easy to express the combination of dice, so I designed the *CombiSchema* class to represent the combination of dice, and did some necessary operations.

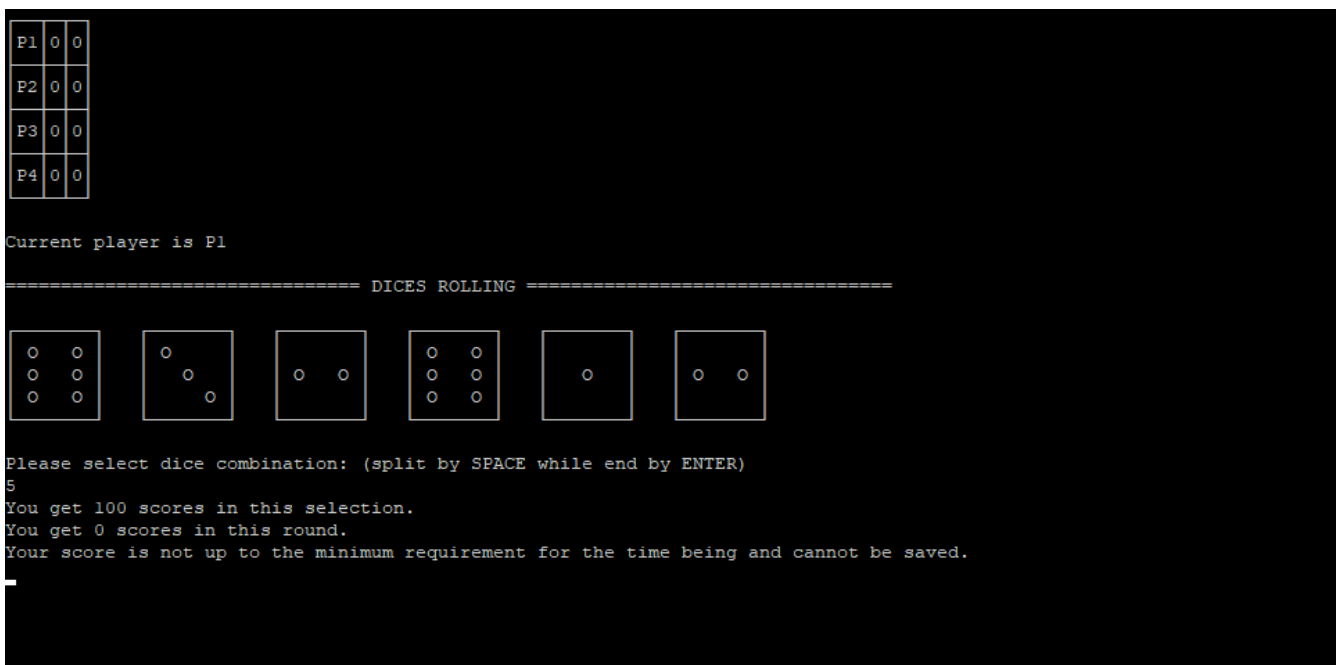
Problems in Implementation

Before the implementation, I designed class diagram and logic flow chart. However, as development progressed, I found that my design was not complete enough. Many details are worth pondering.

For example, at the beginning of the design, I thought that *Dice* should return an *int* value to represent the result, but then I expected it to return a set of strings to show a image of dice pips. As the increasing of the number of changes I made to the original solution, I found that many of the designs clashed, so that I had to reconstruct the class *Player* later.

However, after refactoring, my development progress has been significantly improved. I divide many operations into smaller ones, which simplifies my code and makes it easier to modify. I realized this when I worked on the extra task.

I completed the first four in the *Extra Functionality* list. In the development of the second function, I did not consider the deduction in the design. I get the total score by recording each saved score and summing it up. Thanks to the good design, I quickly completed the revision of the scoring system.



Conclusion

If I do this task again, I will spend more time on project design. It has to be said that project design is a boring task, which requires a lot of repetitive thinking and inspection. However, a good project design brings about a huge improvement in development efficiency, which can avoid many unexpected situations.