**Program report(world of zuul text-based game)**

## The purpose of the game:

This gaming program I did is a simple text-based game that player could have an experience on entering different caves and fighting with monsters. Player will enjoy on killing monster, earning gold and updating attack or defence rate. At the beginning, as a player, player will start player gaming progress from outside the cave entrance without fighting with monster. After that section, player have several options that player can choose which cave player prefer to get in. There will be a random weight monster in every caves include the outside when player return from any other caves. Different weight monster will deal different damage to player based on their weight. Player also have several tool options during the fight. The tool deal more damage on monster will also have larger damage to player as well. Meanwhile, player could also gain gold after killing the monster. Gold could be used in shop(input “shop”), attack(deal more damage to monster), defence(monster deal less damage to player) and blood(add playerr current blood) tool are available for player.

## Classes review:

There are 6 classes have been added to the work.

**Blood**: Return the player’s blood.

**Gold**: Return gold that player has.

**itemsRate**: Initialize and build setter and getter of attack rate and defence rate(could be bought in shop).

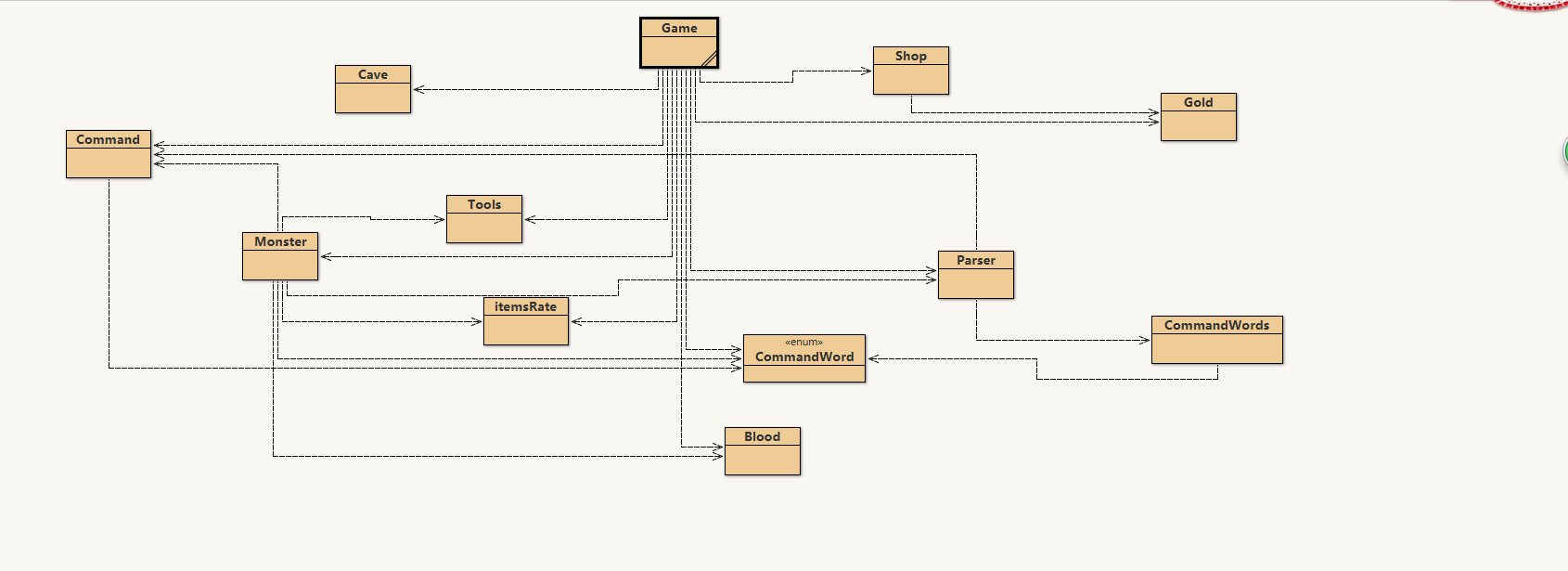
**Tools**: Initialize and generate setter and getter of three different tools’ rate and description. (“start”, ”intermediate”, ”upper”) .

**Shop**: Return text after buying attack tool or defence tool in shop. Print text after buying blood tool.

**Monste**r: Build an object which is “monster”, with random weight between 0-20(include 0). Three tools have been added to this class. Calculation and fighting rules have been added as well in this class. This class also has receiving function of second command words which is the keyword of executing order

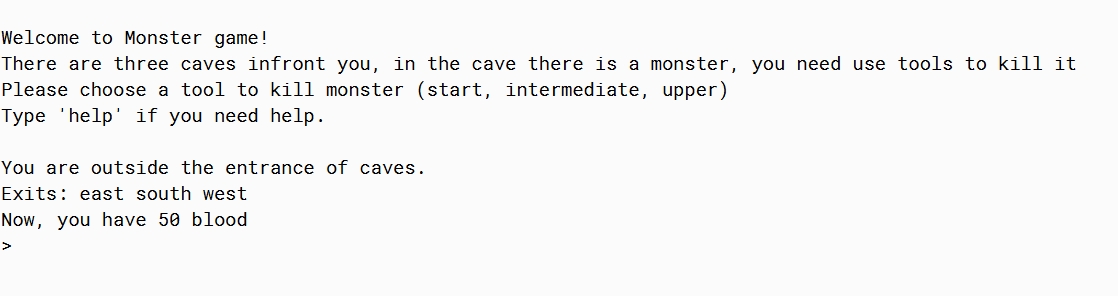
**Other classes**: Other classes from start code have not been modified too much. However some functions in original code have not been used in the game, for example the description of room. Hopefully, these functions will be used in next version of the game.

## Classes diagram(screenshot):

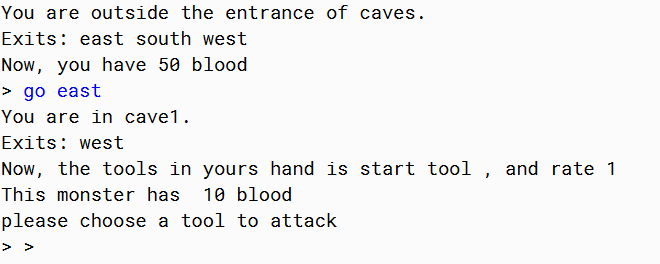


### Running progress of gaming program

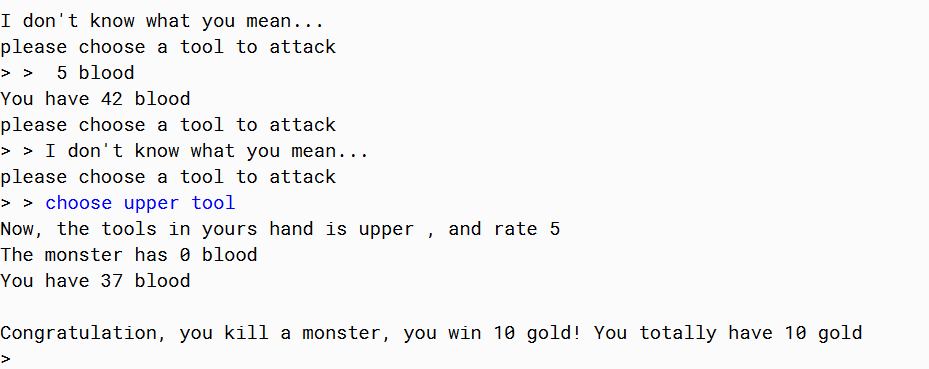
**Print welcome notification:**



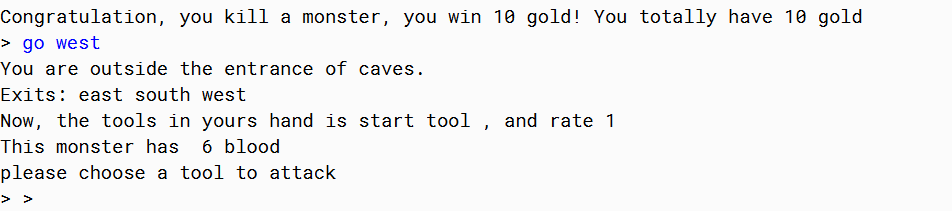
**Input go east, meet with first monster. Print reminder of choosing tool to attack monster.(Choose + tool name )**



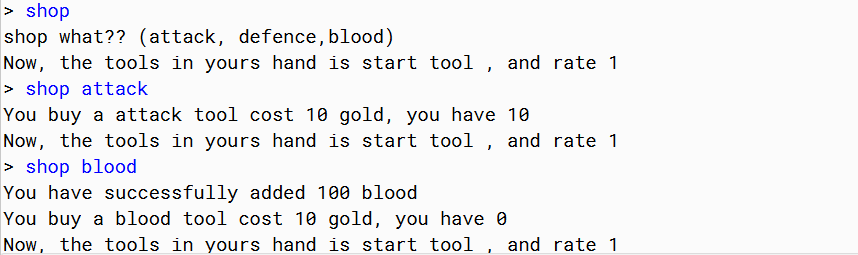
**Print reward player get from killing the monster.**



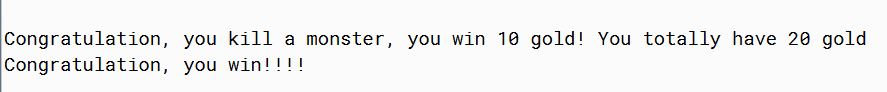
**Input “go west” return to outside, monster will automatically generated.**



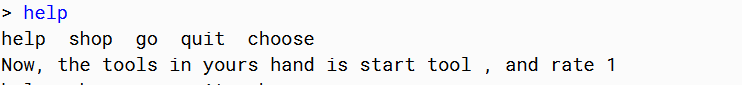
**Input “shop” to enter shop function. Then input “shop” +”(the function player want to buy)” to buy any tools.**



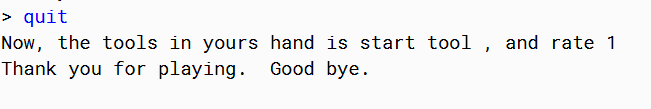
**While killed 4 monsters, player win the game, system print victory notification and quit program.**



**Input “help” to print keywords.**



**Input “quit” quit game. Print quitting text.**



## Current status:

Program now could be run smoothly. The logic of gaming progress is clear and fundamental factors of game have been built up successfully. Enjoyability, factors and complexity of this game is acceptable as a small starter text-based game. However, victory condition is too easy for player to achieve. According to limiting knowledge and understanding of object oriented programming, I was stuck in adding a special cave and a final boss to the game. In the next version, hopefully I will find a new way to add these factors and change winning condition to kill the final boss. In the other hand, the principles and ideology have not fully implemented in this course work due to the limitation of the knowledge and practice experience on java programming. The concept of “object oriented” has not been applied in the program perfectly at this version. I may have a misunderstanding between “object oriented” and “Procedure Oriented”. I will try to solve it out before next version. According to textbook and online resources, the idea is “interface” may be able to be used in next version.

## Next:

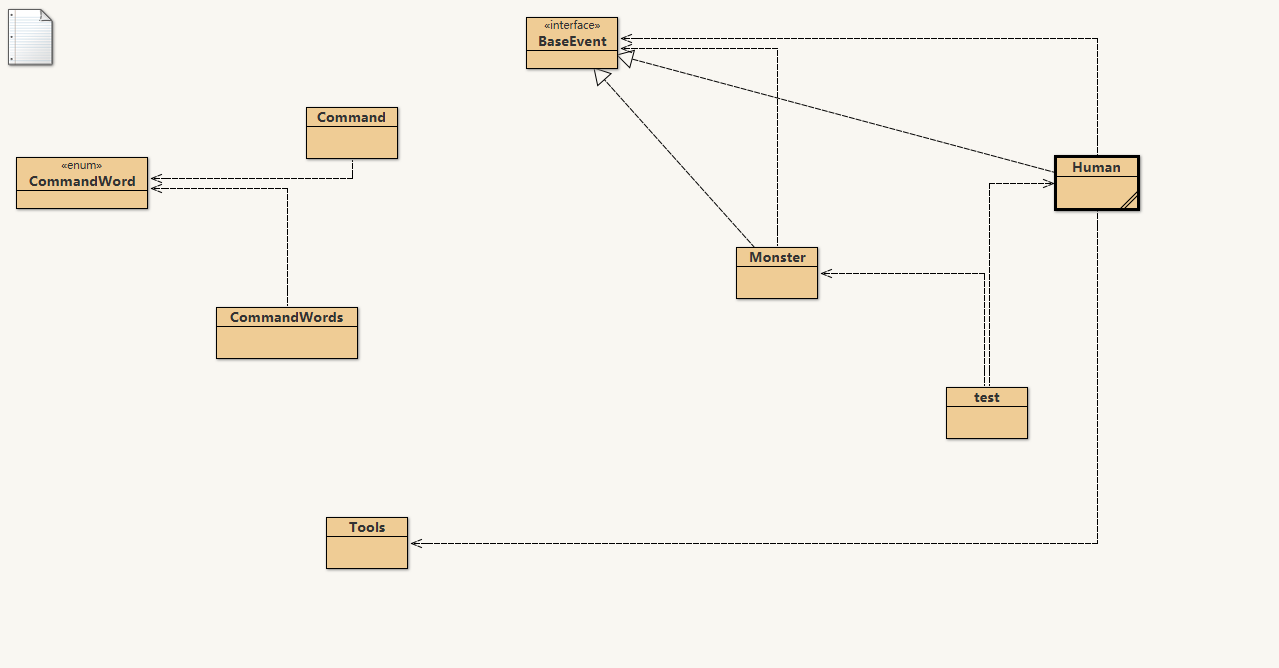
I have already started to create a new version of writing the code. “Object oriented” principle and interface should be used in new version.

The screenshot in appendix is several classes what I have done so far.

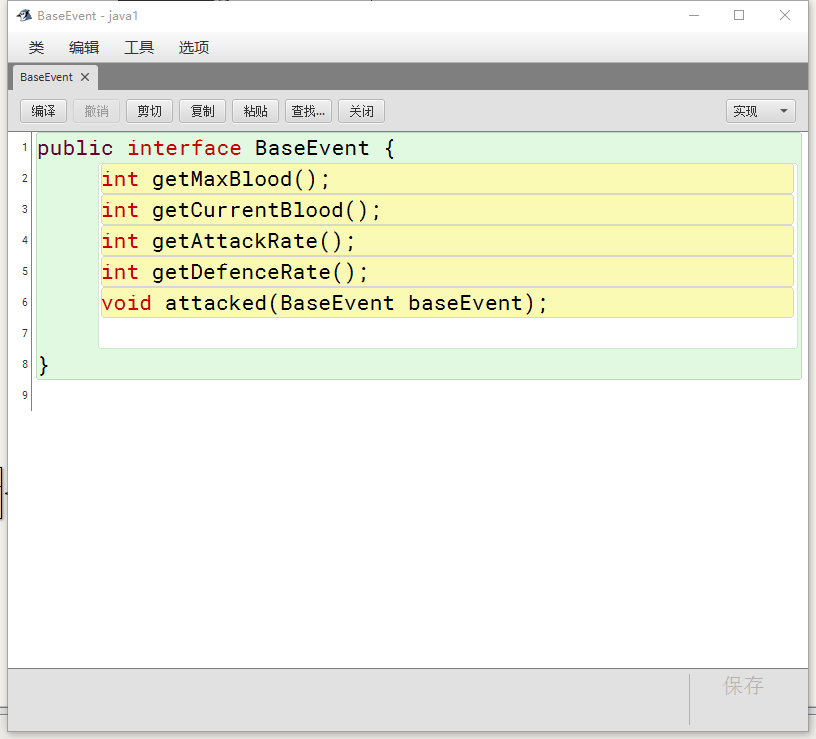
## Appendix

### Classes diagram:

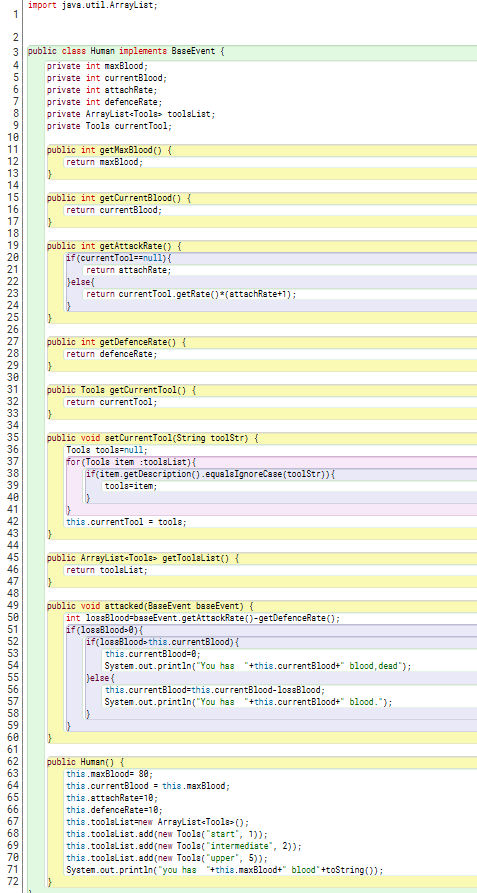
Here is part of work screenshots of the works using interface and object oriented ideology should be fully applied（Unfinished）:



##### Class BaseEvent(interface):



##### Class Human:



##### Class Monster:



Class Tools: