ASSIGNMENT 3

B21946529

Fatih Pehlivan

PROBLEM

Our purpose is to make a war game. There are two teams which are called “Zorde” and “Calliance”. Each team occurs 3 different characters. Every character has different heal, attack point, and max move. Some characters have special features. Orks increase team heal, Elfs make a ranged attack in the last step. In the end, find the last alive character and determine the team which won.

SOLUTION

First of all my purpose is reading txt files. I created a java class “ReadFile”. The class method “Read” returns 2D ArrayList. Its elements are lines of txt file. Line elements are occurred strings separated by tab. I read “initialize” file.

I read board shape and make my LinkedHashMap “board” filled value set with empty (“ “) strings, and filled keyset according to y,x coordinates. For example (0,0 0,1 0,2 … so on). This is a bit confused because in initials and in commands coordinates are reverse (x,y). Then according to initials coordinates, I added players' id in proper value.

Then I created classes and subclasses (these classes will be explained next paragraphs) and “public String outputString, public ArrayList<Armies> armiesArrayList, and public LinkedHashMap<String, String> board” in the Main class to solve this problem. I added every output in my “outputString”, so it should be public. I added classes and subclasses, fields, and constructors. I added my objects in properly armiesArrayList. armiesArrayList is public because I have to reach this arrayList in my classes’ methods. Because when players are attacking, the code must know the other players are enemy or ally. board is public because when players are attacking, the code must show their neighbors squares and must show who is in these squares (empty or ally id or enemy id).

ARMIES: I created Armies class which is a subclass of Exception. The class only takes the id parameter. I defined some default variables (AP and heal) and I wrote getter and setter methods. I defined heal as an Integer, not “int”. Because I will use compareTo method and it must be class. I defined these variables because I wrote a method and I have to use getter and setter methods there. Other variables to check fight to death and stop statements. Because both of them finalize the movements. Then I wrote these variables getter methods.

Two methods in very important in this class.

The first one is “movements” method. It takes 2 parameters: String “move” and Armies “champion”. “Move” contains only one “;” I set that when I am using this method. I find the champion id in “board”. I convert type String key to int, I add “move” properly the key. If the key is not on the board.KeySet() the method throws a new exception (Boundary Check). Then I found the champion is an instance of Calliances or Zordes. Because if champion goes to ally square, the command line must be stopped and make “stop” variable true. The second case champion goes to the enemy square program should run “makeFightToDeath” and make “fightToDeath” is true.

The second one is “makeFightToDeath” method. It takes 4 parameters: Armies “championAttacking”, Armies “championDefending”, String “previousKey”, and String “battleArea”. First of all the method reduces champion defending heal according to attacker Attack Point. Then according to heal points champions killed.

CALLIANCES: I created Calliances abstract class which is subclass of Armies. It has one constructer because of Armies class and it has 4 methods (getHeal(), setHeal(), getDefaultHeal(), and attack(). This is an abstract class, so I don’t have to fill methods.

HUMAN: I created Human class which is subclass of Calliances. It has one constructor because of Calliances class and I made only one attribute which is heal. These constants were taken from Constants.java. There 5 methods in this class.

1st is getHeal(): This method returns Integer instant heal. This method was used when comparing heals and measure heals bigger than zero or not.

2nd is setHeal(int heal): This method returns a new heal, I used this method when they are attacking.

3rd getAP(): Ap is constant, it was taken from Constants.java

4th getHumanMaxMove(): This is also constant and taken from Constants.java

5th attack(String move, Armies champion): “move” is taken from according to the command file. Firstly, I checked maximum move and command move are equals to each other. If they are not equal my code throws a new exception (Move Count Check). Then I determined the last step key value. I calculated the champion’s key value by using for loop and calculated neighbor squares. If there is an enemy in neighbor squares I set their heal by using champion.setHeal(int Heal) method. I calculate its parameters by using champion.getHeal () – getAP(). If new heal is less than zero, I removed the champion from arrayListArmies and removed its id the board.

Other classes (Elf, Dwarf, Ork, Troll, Goblin) are like Human class. I will explain only their differences.

ELF: I called movements() method for every step. Because every step Elf makes an attack. If it was stopped or it makes fight to death the loop is broken by break keyword. I set counter and it achieves getElfMaxMove() neighbor squares range will be 2.

DWARF: The logic the same as Elf class. The only difference is there is no ranged attack.

ORK: The logic the same as Human. This class is subclass of Zordes. Some constants are different like Heal, AP so on. The other difference is I calculated the area where it is located and increase the neighbor ally's heal according to HP which is in Constant.java

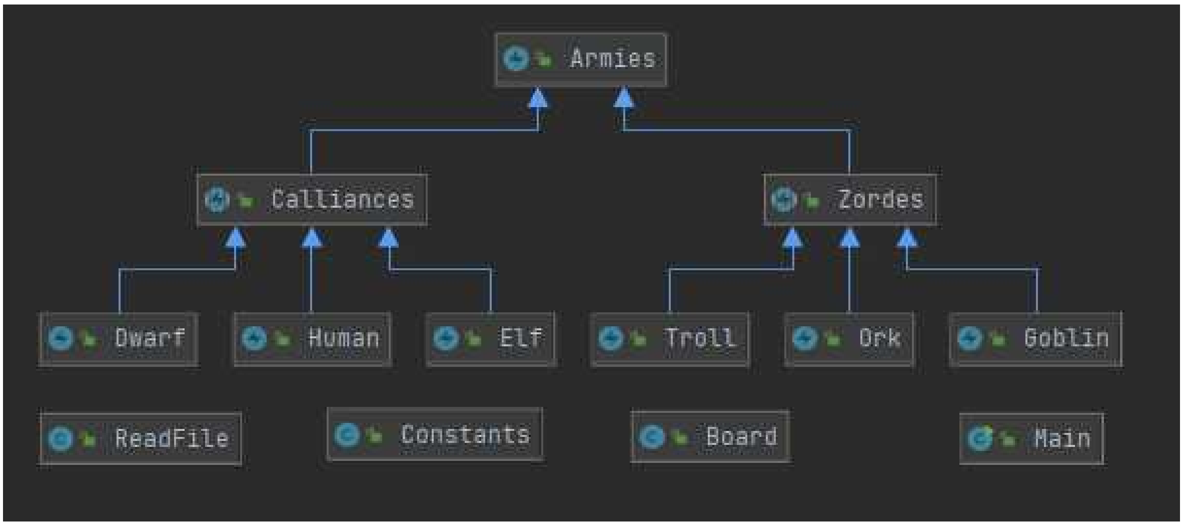
TROLL: The logic the same as Human. This class is subclass of Zordes.

GOBLIN: The logic the same as Dwarf. This class is subclass of Zordes.

I read "command.txt" and I started to apply commands by using these classes.

I add outputString, my board, by using "Board" class. board method calculates how many stars it should be bottom and top. Then by using for loops add the board on outputString properly.

At the end of the commands, I wrote my outputString into the given txt file.



Main: It makes rules and sets the run orders. It creates objects and the first board.

Board: It makes only add the board on outputString.

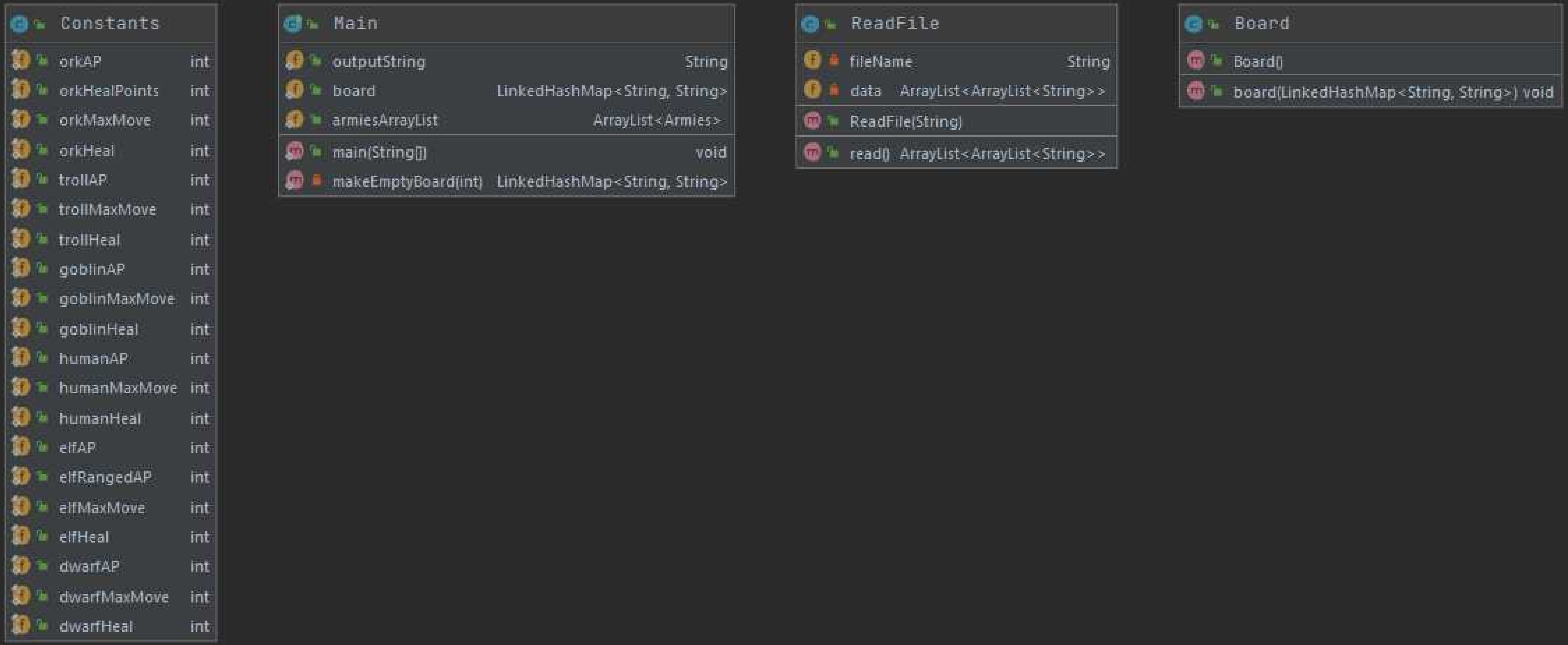
ReadFile: It makes only input files make 2D ArrayLists.

Armies: The class provides movements and fight-to-death attacks.

Calliances & Zordes: These are abstract classes. They provide only separate teams.

Dwarf & Human & Elf & Troll & Ork & Goblin: These classes provide make attack set and get heals, attack point so on.

Constants: It determines constant values. For example: heal, attack point, max move so on.

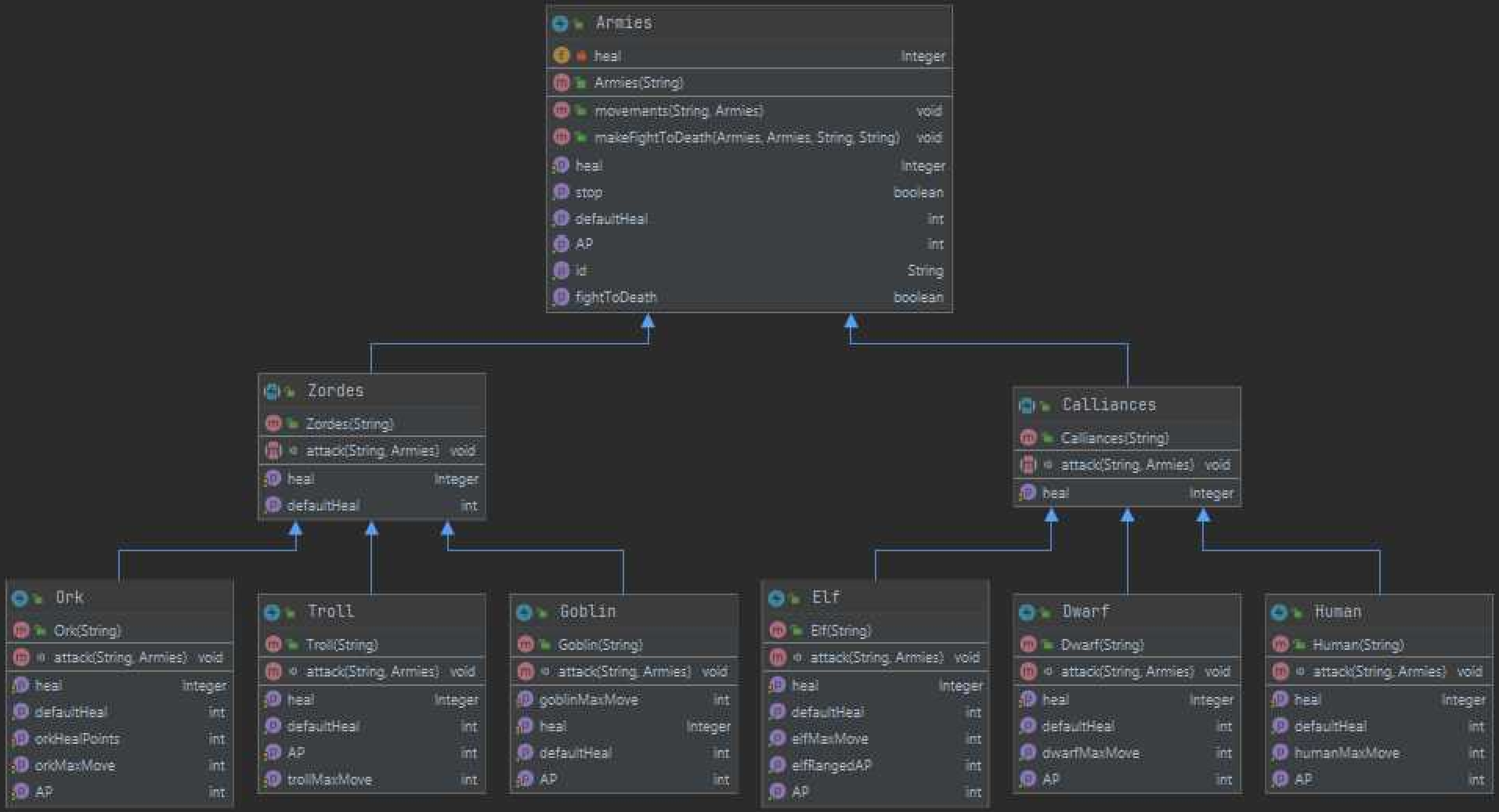


Constants	
orkAP	int
orkHealPoints	int
orkMaxMove	int
orkHeal	int
trollAP	int
trollMaxMove	int
trollHeal	int
goblinAP	int
goblinMaxMove	int
goblinHeal	int
humanAP	int
humanMaxMove	int
humanHeal	int
elfAP	int
elfRangedAP	int
elfMaxMove	int
elfHeal	int
dwarfAP	int
dwarfMaxMove	int
dwarfHeal	int

Main	
outputString	String
board	LinkedHashMap<String, String>
armiesArrayList	ArrayList<Armies>
main(String[])	void
makeEmptyBoard(int)	LinkedHashMap<String, String>

ReadFile	
fileName	String
data	ArrayList<ArrayList<String>>
ReadFile(String)	
read()	ArrayList<ArrayList<String>>

Board	
Board()	
board(LinkedHashMap<String, String>)	void



Armies
heal Integer
Armies(String)
movements(String, Armies) void
makeFightToDeath(Armies, Armies, String, String) void
heal Integer
stop boolean
defaultHeal int
AP int
id String
fightToDeath boolean

Zordes
Zordes(String)
attack(String, Armies) void
heal Integer
defaultHeal int

Calliances
Calliances(String)
attack(String, Armies) void
heal Integer

Ork
Ork(String)
attack(String, Armies) void
heal Integer
defaultHeal int
orkHealPoints int
orkMaxMove int
AP int

Troll
Troll(String)
attack(String, Armies) void
heal Integer
defaultHeal int
AP int
trollMaxMove int

Goblin
Goblin(String)
attack(String, Armies) void
goblinMaxMove int
heal Integer
defaultHeal int
AP int

Elf
Elf(String)
attack(String, Armies) void
heal Integer
defaultHeal int
elfMaxMove int
elfRangedAP int
AP int

Dwarf
Dwarf(String)
attack(String, Armies) void
heal Integer
defaultHeal int
dwarfMaxMove int
AP int

Human
Human(String)
attack(String, Armies) void
heal Integer
defaultHeal int
humanMaxMove int
AP int