

TASK

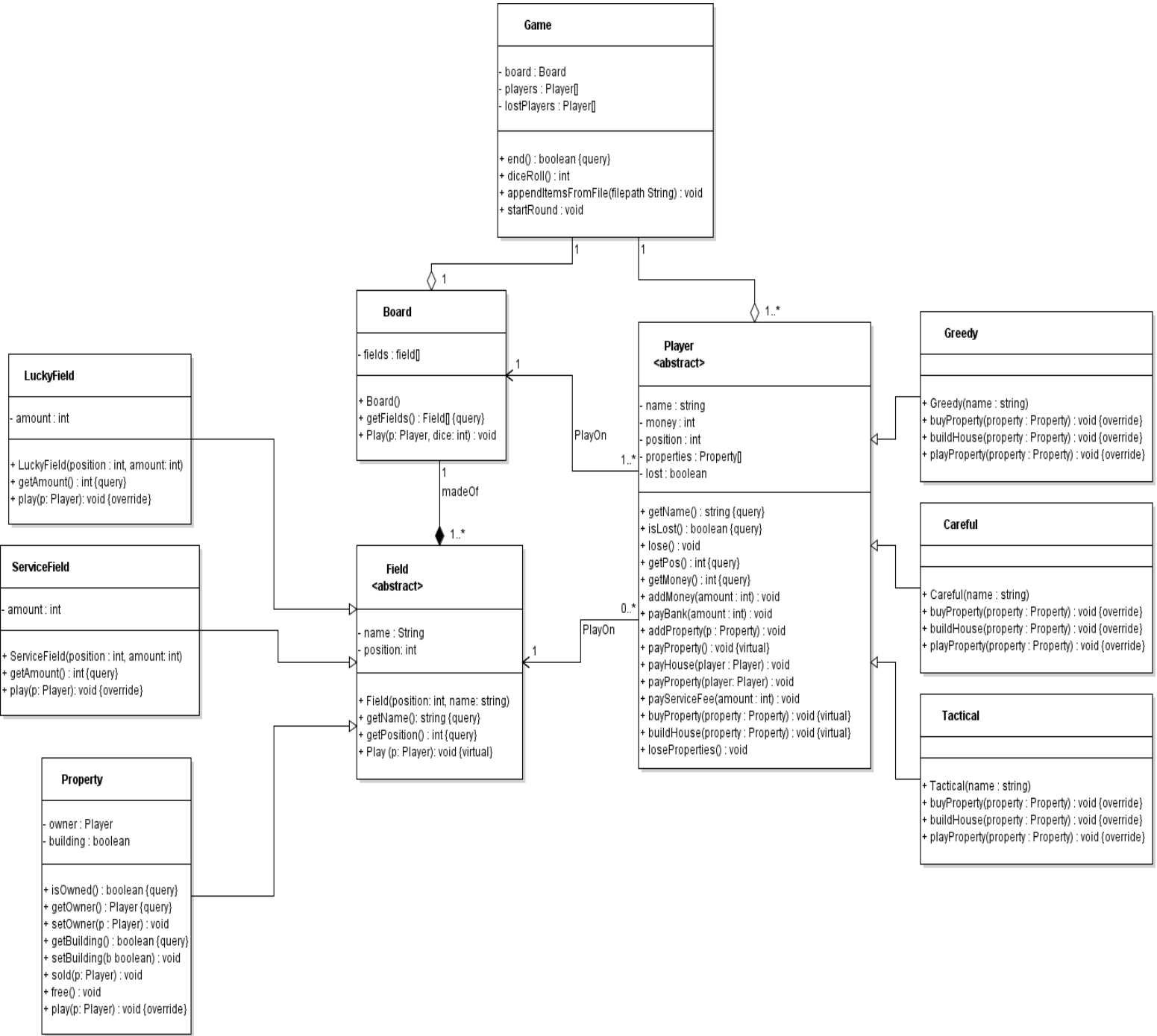
Simulate a simplified Capital game. There are some players with different strategies, and a cyclical board with several fields. Players can move around the board, by moving forward with the amount they rolled with a dice. A field can be a property, service, or lucky field. A property can be bought for 1000, and stepping on it the next time the player can build a house on it for 4000. If a player steps on a property field which is owned by somebody else, the player should pay to the owner 500, if there is no house on the field, or 2000, if there is a house on it. Stepping on a service field, the player should pay to the bank (the amount of money is a parameter of the field). Stepping on a lucky field, the player gets some money (the amount is defined as a parameter of the field). There are three different kind of strategies exist. Initially, every player has 10000.

Greedy player: If he steps on an unowned property, or his own property without a house, he starts buying it, if he has enough money for it.

Careful player: he buys in a round only for at most half the amount of his money. Tactical player: he skips each second chance when he could buy. If a player has to pay, but he runs out of money because of this, he loses. In this case, his properties are lost, and become free to buy.

Read the parameters of the game from a text file. This file defines the number of fields, and then defines them. We know about all fields: the type. If a field is a service or lucky field, the cost of it is also defined. After the these parameters, the file tells the number of the players, and then enumerates the players with their names and strategies. In order to prepare the program for testing, make it possible to the program to read the roll dices from the file.

Print out which player won the game, and how rich he is (balance, owned properties).



State	Expected value	Actual value
Testing Board class		
Testing addField() method		
After initiating board	0	Board.getFields().size()
After adding 1 field	1	Board.getFields().size()
Testing play() method		
After initiating board	0	Board.getFields().size()
After adding 1 field	1	Board.getFields().size()
After player play on the field	0	Player position
	9000	Player money
After player play on the field	0	Player position
	5500	Player money
	1	Player property count
Testing Field class		
Testing play() method		
After initiating player and field	10000	Player money
	0	Player property count
After player played on field	9000	Player money
	1	Player property count
Testing Lucky Field class		
Testing play() method		
After initiating greedy player and field	10000	Greedy player money
After player greedy played on field	12000	Greedy player money
After initiating careful player and field	10000	Careful player money
After player careful played on field	12000	Careful player money
After initiating tactical player and field	10000	Tactical player money
After player tactical played on field	12000	Tactical player money
Testing Service Field class		
Testing play() method		
After initiating greedy player and field	10000	Greedy player money
After player greedy played on field	8000	Greedy player money
After initiating careful player and field	10000	Careful player money
After player careful played on field	8000	Careful player money
After initiating tactical player and field	10000	Tactical player money

After player tactical played on field	8000	Tactical player money
Testing Property field class		
Testing isOwned() method		
After initiating property	False	isOwned
After player played on this filed	True	isOwned
Testing play() method		
After greedy player 1 played on property	Greedy player 1	Owner of property
After greedy player 2 played on property	9500	Greedy player 2 money
After greedy player 1 played again	True	House on the property
After greedy player 2 played again	7500	Greedy player 2 money
After greedy player 1 played again	8000	Greedy player 1 money
After careful player played on property	8000	Careful player money
After careful player played on property2	7000	Careful player money
After careful player played on property2 again	3500	Careful player money
	1	Careful player property count
After careful player played on property3	1	Careful player property count
After tactical player played on property3	1	Tactical player property count
	9000	Tactical player money
	Tactical player	Owner of property3
After tactical player played on property3 again	false	Property3 building built
	9000	Tactical player money
After tactical player played on property3 again	true	Property3 building built
	6000	Tactical player money
Test sold() method		
After initiating greedy player and property	Null	Owner of Property
After selling the property to player	Greedy player	Owner of Property
Testing free() method		
After initiating greedy player and property	Null	Owner of Property
After selling the property to player	Greedy player	Owner of Property

After freeing property	Null	Owner of Property
Testing Player class		
Testing payBank() method		
After initiating player	10000	Player money
After paying bank 2000 with the method	8000	Player money
Try paying 10000 to the bank	NotEnoughMoney	Exception
Test addProperty() method		
Initiating new greedy player	0	Count of properties of player
After adding property	1	Count of properties of player
Test pay() method		
After initiating Player1 and player2	10000	Player1 money
	10000	Player2 money
After Player1 pay 2000 to player2	8000	Player1 money
	12000	Player2 money
Try to player1 pay 10000 to player2	NotEnoughMoney	Exception
Test payHouse() method		
After initiating Player1 and player2	10000	Player1 money
	10000	Player2 money
After Player1 pay house money to player2	8000	Player1 money
	12000	Player2 money
Test payProperty() method		
After initiating Player1 and player2	10000	Player1 money
	10000	Player2 money
After Player1 pay property money to player2	9500	Player1 money
	10500	Player2 money
Test payServiceFee() method		
After initiating Player1 and player2	10000	Player1 money
After Player1 pay service field fee	8000	Player1 money
Test loseProperties() method		
Initiate player	0	Player properties count
Add 2 properties to player	2	Player properties count
After lose property method	0	Player properties count
Testing Careful class		
Testing buyProperty() method		

Initiate 6 different property and careful player	Null	Property1 owner
	Null	Property2 owner
	Null	Property3 owner
	Null	Property4 owner
	Null	Property5 owner
	Null	Property0 owner
After player played on all 6 property from 0-5	player	Property0 owner
	player	Property1 owner
	player	Property2 owner
	player	Property3 owner
	player	Property4 owner
	Null	Property5 owner
Testing buildHouse() method		
Initiate property and player	Null	Property owner
After player bought on the property	Player	Property owner
	False	Building on the property
After player build house on the property	true	Building on the property
Testing playProperty() method		
Initiate player and property	10000	Player money
	0	Player properties count
Player played on the property	9000	Player money
	1	Player properties count
After subtracting 5000 from the player money player played again on property	9000	Player money
	1	Player properties count
Testing Greedy Player class		
Testing buyProperty() method		
Initiate 6 different property and greedy player	Null	Property1 owner
	Null	Property2 owner
	Null	Property3 owner
	Null	Property4 owner
	Null	Property5 owner
	Null	Property0 owner
After player played on all 6 property from 0-5	player	Property0 owner
	player	Property1 owner

	player	Property2 owner
	player	Property3 owner
	player	Property4 owner
	player	Property5 owner
Testing buildHouse() method		
Initiate property and player	Null	Property owner
After player bought the property	Player	Property owner
	False	Building on the property
After player build house on the property again	true	Building on the property
Testing playProperty() method		
Initiate player and property	10000	Player money
	0	Player properties count
Player played on the property	9000	Player money
	1	Player properties count
Testing Tactical Player class		
Testing buyProperty() method		
Initiate 6 different property and greedy player	Null	Property1 owner
	Null	Property2 owner
	Null	Property3 owner
	Null	Property4 owner
	Null	Property5 owner
	Null	Property0 owner
After player played on all 6 property from 0-5	player	Property0 owner
	null	Property1 owner
	player	Property2 owner
	null	Property3 owner
	player	Property4 owner
	null	Property5 owner
Testing buildHouse() method		
Initiate property and player	Null	Property owner
After player bought the property	Player	Property owner
	False	Building on the property
After player build house on the property again	false	Building on the property
After player build house on the property again	true	Building on the property
Testing playProperty() method		

Initiate player and property	10000	Player money
	0	Player properties count
Player played on the property	9000	Player money
	1	Player properties count
Testing Game class		
Testing AppendFromFile() method		
Appending from file fail1.txt	InvalidInput	Exception
Appending from file fail2.txt	NumberFormatException	Exception
Appending from file fail3.txt	NumberFormatException	Exception
Appending from file fail4.txt	NumberFormatException	Exception
Appending from file fail5.txt	InvalidInput	Exception

Default input file:

9

P Kiraly utca

S 2000

P Wesseleny utca

S 2000

P Rakoczi ter

S 2000

P Blaha Luiza ter

P Corvin negyed

L 2000

3

G greedy

C careful

T tactical

Default dice roll file: (but 1 dice roll in 1 row)

2,2,2,2,2,2,2,2,2,2,4,4,3,2,2,4,3,3,2,4,4,9,3,3

This one is designed to let the Greedy player win with 3 properties and 14000 credits worth. When program finishes it prints following:

Name: Sutú

Money left: 14000 Properties count: 3

Properties: [

Property name: Wesseleny utca with building

Position: 2,

Property name: Rakoczi ter without building

Position: 4,

Property name: Blaha Luiza ter with building

Position: 6]