

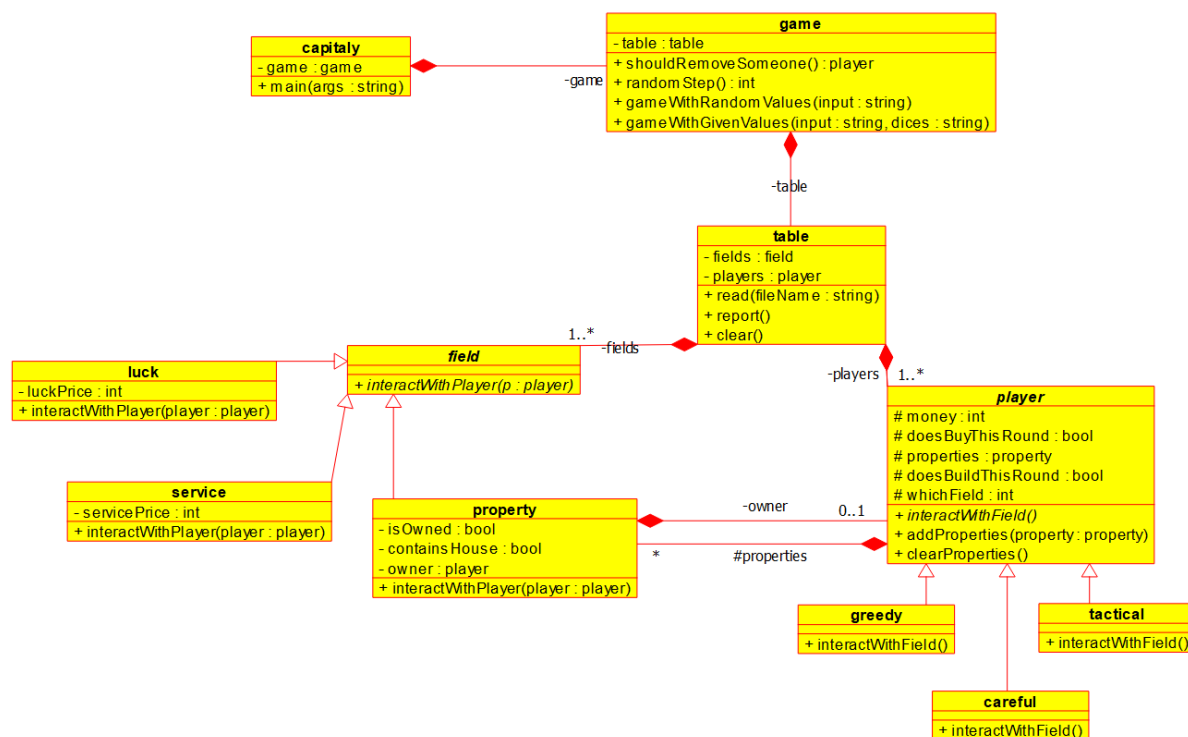
Nagy Balázs (LHDN48) Programozási technológia 1.beadandó

5. Szimuláljuk az alábbi egyszerűsített Capitaly társasjátékot! Adott néhány eltérő stratégiájú játékos és egy körpálya, amelyen különféle mezők sorakoznak egymás után. A pályát körbe-körbe újra és újra bejárják a játékosok úgy, hogy egy kockával dobva mindig annyit lépnek, amennyit a kocka mutat. A mezők három félék lehetnek: ingatlanok, szolgáltatások és szerencse mezők. Az ingatlant meg lehet vásárolni 1000 Petákért, majd újra rálépve házat is lehet rá építeni 4000 Petákért. Ha ezután más játékos erre a mezőre lép, akkor a mező tulajdonosának fizet: ha még nincs rajta ház, akkor 500 Petákot, ha van rajta ház, akkor 2000 Petákot. A szolgáltatás mezőre lépve a banknak kell befizetni a mező paramétereként megadott összeget. A szerencse mezőre lépve a mező paramétereként megadott összegű pénzt kap a játékos. Háromféle stratégiájú játékos vesz részt a játékban.

Kezdetben mindenki kap egy induló tőkét (10000 Peták), majd a „mohó” játékos ha egy még gazdátlan ingatlan mezőjére lépett, vagy övé az ingatlan, de még nincs rajta ház, továbbá van elég tőkéje, akkor vásárol. Az „óvatos” játékos egy körben csak a tőkéjének a felét vásárolja el, a „taktikus” játékos minden második vásárlási lehetőséget kihagyja. Ha egy játékosnak fizetnie kell, de nincs elegendő pénze, akkor kiesik a játékból, házai elvesznek, ingatlanjai megvásárolhatókká válnak.

A játék paramétereit egy szövegfájlból olvassuk be. Ez megadja a pálya hosszát, majd a pálya egyes mezőit. Minden mezőről megadjuk annak típusát, illetve ha szolgáltatás vagy szerencse mező, akkor annak pénzdíját. Ezt követően a fájl megmutatja a játékosok számát, majd sorban minden játékos nevét és stratégiáját. A tesztelhetőséghez fel kell készíteni a megoldó programot olyan szövegfájl feldolgozására is, amely előre rögzített módon tartalmazza a kockadobások eredményét. *Írjuk ki, melyik játékos esik ki másodszorra a játékból!*

Osztálydiagram:



Tesztelés:

Három különböző tesztet hoztam létre. A következő táblázatok mutatják a meghívott függvényeket, valamint az elvárt kimeneteket, amiket a Capitaly osztályban tehetünk meg.

input1.txt:

8

P P S 500 L 1000 L 500 P S 2000 L 500

G Jani T Olga C Bandi

input2.txt:

8

P P S 10000 L 1000 L 500 P S 11000 L 500

G Géza T Tamás C Csilla

dices2.txt:

1 2 6 6 1 2 4 5

input3.txt:

4

P P S 1000 L 1000

G Mohó T Taktikus C Óvatos

dices3.txt:

1

input4.txt:

11

S 2000 P L 5000 S 5000 P P S 2000 L 500 P S 300 P

G Ákos T Béla C Cecíl T Dénes

dices4.txt:

1 2 3 4

input5.txt:

11

S 2000 P L 5000 S 5000 P P S 2000 L 500 P S 300 P

G Ákos T Béla C Cecíl T Dénes

dices5.txt:

4 3 2 1

	1.
Bemenet/Függvény	game.gameWithRandomValues("input1.txt")
Elvárt kimenet	A kimenet random, mivel random számokkal dolgozunk, de elvárható, hogy a kiírt kezdeti állapot mindig ugyanolyan legyen, valamint váltakozzon, hogy ki esett ki utoljára.
	2.
Bemenet/Függvény	game.gameWithGivenValues("input2.txt", "dices2.txt");
Elvárt kimenet	Kezdeti állapot kiírása helyesen, valamint a második kieső Csilla.
	3.
Bemenet/Függvény	game.gameWithGivenValues("input3.txt", "dices3.txt");
Elvárt kimenet	Kezdeti állapot kiírása helyesen, valamint a második kieső Óvatos.
	4.
Bemenet/Függvény	game.gameWithGivenValues("input4.txt", "dices4.txt");
Elvárt kimenet	Kezdeti állapot kiírása helyesen, valamint a második kieső Dénes.
	5.
Bemenet/Függvény	game.gameWithGivenValues("input5.txt", "dices5.txt");
Elvárt kimenet	Kezdeti állapot kiírása helyesen, valamint a második kieső Cecília.

Metódusok rövid leírása (javadocban szintűgy elérhető):

Game osztály:

public Player shouldRemoveSomeone()

-visszaadja az első playert, akinek 0-nál kevesebb pénze van, ha ilyet nem talál null-t ad vissza

public int randomStep()

-random számot ad 100 alatt

public void gameWithRandomValues(String input)

-lefuttat egy játékot random kockadobásokkal, bemenetként a játékosokat és a pályát tartalmazó szöveget várja

public void gameWithGivenValues(String input, String dices)

-lefuttat egy játékot előre megadott kockadobásokkal, bemenetként a játékosokat és a pályát tartalmazó szöveget, valamint a kockadobásokat tartalmazó szöveget várja

Table osztály:

public void clear()

-kiüríti a mezőket, valamint a játékosokat

public void read(String filename) throws FileNotFoundException,
InvalidInputException

-beolvas, és exceptiont dob, amit a meghívónál kapok el

public void report()

-megjeleníti a tábla aktuális állapotát

Player osztály és alosztályok:

public void addProperties(Property property)

-egy tulajdont fűz egy játékoshoz

public void clearProperties()

-törli a játékos tulajdonait

public void interactWithField()

-vannak alosztályok, amelyeknek azt, hogy vásárolnak-e egy adott körben, vagy sem ezzel a metódussal modifikálom

Field osztály és alosztályok:

public void interactWithPlayer(Player p)

-a mezőkre lépve a játékosokkal ezt a metódust hívom meg, majd ez egyben változtatja a játékost (megváltoztatva a pénzét, vagy megváltoztatva más tulajdonságait az interactWithField-el), valamint a Property-nál vásárlás esetén is ez a metódus állítja be a tulajdonost, valamint adóztatja meg a többi játékost/fizet a tulajdonosnak