

Feladat

3. Készítsünk C++ programot a következő feladat megoldására!

Szimuláljuk az alábbi egyszerűsített Monopoly 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, hogy adott számú kör után hogyan állnak (mennyi a tőkéjük, milyen ingatlanokat birtokolnak) a versenyzők!

Specifikáció

A játékosok leírásához bevezetünk négy osztályt: a játékosok általános tulajdonságait megadó ősosztályt (Játékos), és ebből származtatjuk a konkrét taktikájú játékosok, a mohó, az óvatos és a taktikus taktikájú játékosok osztályait. Attól függetlenül, hogy egy játékos konkrétan kicsoda vagy mi a taktikája, számos közös tulajdonsággal rendelkezik. Mindegyiknek van neve (`_név`), pénze (`_pénz`), egy eltárolt helyzete (`_holtart`) és lehet tulajdona (`_tulajdon`). Meg lehet róla kérdezni, hogy hívják (`név()`), játékban van-e (`játékban()`, amíg van pénze), mennyi pénze van (`penz()`), hol tart a táblán (`holtart()`), hány ingatlana van, és milyen típusúak (`ingSzam()`, `Tul()`). Lehet összeget adni neki, és levonni tőle bizonyos összeget (`penzFizet()`, `penzKap()`), és lehet léptetni (`lep()`). Meg lehet vizsgálni, mi történik, ha a versenypálya egy bizonyos mezőjére lép. Ez utóbbi művelet (`akcio()`) módosítja a játékos pénzét, és végrehajtja a feladatban meghatározott változtatásokat. Az `akcio()` metódus csak a konkrét osztályok szintjén implementálható, de minden más már az ősosztályban. Ennél fogva az általános játékos típusát leíró osztály absztrakt lesz, hiszen az `akcio()` metódus absztrakt, másrészt úgysem akarunk ilyen objektumot létrehozni.

A speciális játékos osztályok konstruktorai az ősosztály konstruktorával inicializálják a nevet, és egyedi módon felüldefiniáljuk az `akcio()` metódust. Az inicializáláshoz és a felüldefiniáláshoz a feladat szövegéből kinyert adatok adnak utasítást:

1. Amennyiben Szerencse, Szolgáltatás, vagy olyan Ingatlan mezőre lép, ami valaki más tulajdona, vagy a saját tulajdona és van már rajta ház, mindegyik taktikájú játékos ugyan azt hajtja végre (amennyiben még játékban vannak):

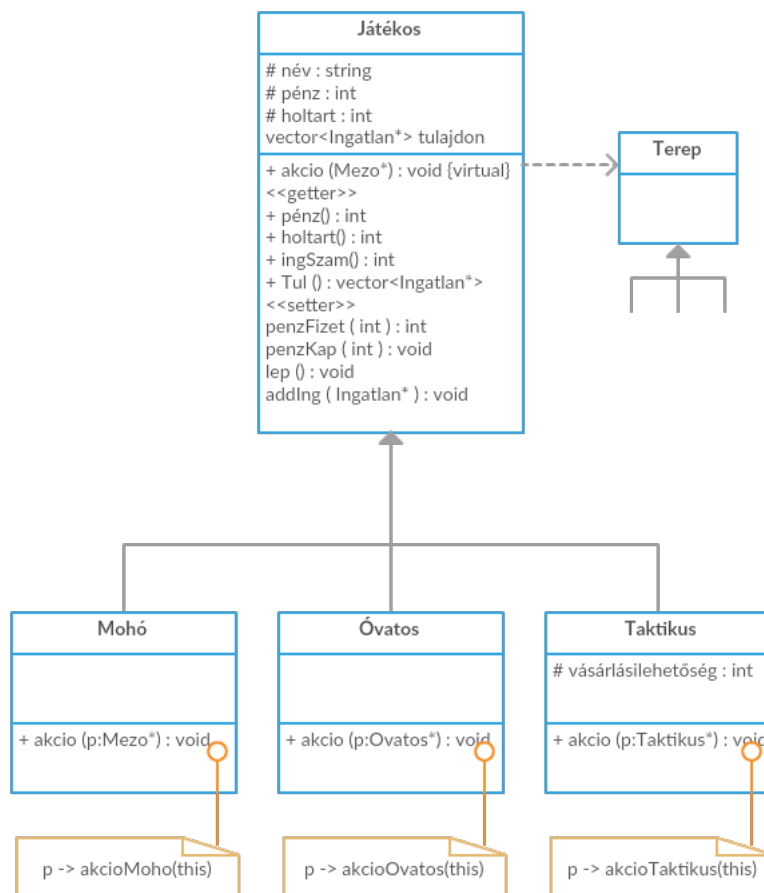
Mező	Pénz változás
Szerencse	+ a bemeneti fájlnak megfelelően
Szolgáltatás	- a bemeneti fájlnak megfelelően
Ingatlan (ha valaki más birtokolja)	kifizet a megfelelő összeget
Ingatlan (ha ő birtokolja, és van rajta ház)	nem csinál semmit

2., Ha olyan ingatlan mezőre lép, ami saját tulajdona, és van vásárlási lehetősége (tehát nincs még rajta háza)

Játékos	Cselekvés
Mohó	Ha van elég pénze, vásárol
Óvatos	Ha legalább kétszer annyi pénze van, mint amennyiért vásárolni akar, vásárol
Taktikus	Minden második vásárlási lehetőséget kihagy

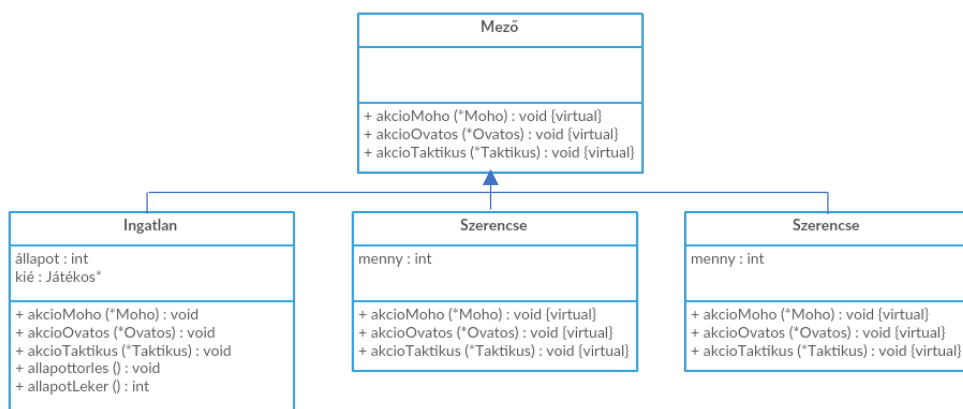
Programterv

A mezők általános tulajdonságait egy őszosztály (Mezo) írja le, és ebből származtatjuk a konkrét mezőket, az ingatlan, a szolgáltatás, a szerencse osztályait. Minden konkrét mező három-három metódussal bír: ezek rendre azt mutatják meg, hogyan változik az adott terepen egy zöldike, egy tocsogó, illetve egy buckabogár, és hogyan esetleg módosul-e eközben a terep.



A megfelelő függvények a megfelelő változó értékét kéri le, vagy állítják be.

A konkrét játékosok `akcio()` metódusa paraméterként vár egy mező objektumot, mint látogatót, és ennek a konkrét játékos típusához illeszkedő metódusát hívja vissza. A mohók esetén ez az `akcioMoho()`, óvatosoknál az `akcioOvatos()`, taktikusoknál az `akcioTaktikus()`.

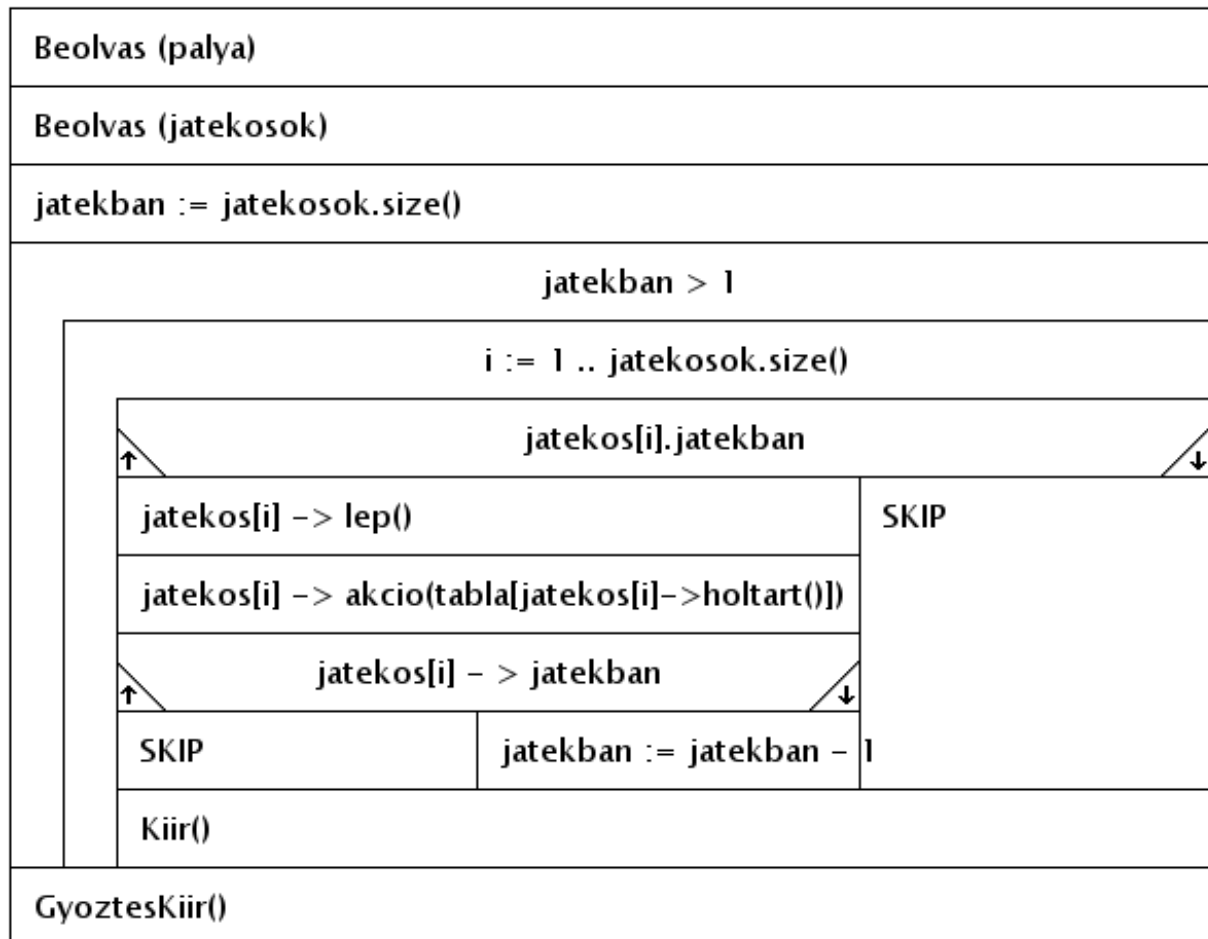


Állapottér: tábla: Mező^m , játékosok: Játékos^n

Előfeltétel: $\text{játékosok} = \text{játékosok}_0 \wedge \text{lény} = \text{lény}_0$

A specifikációnak megfelelő algoritmus:

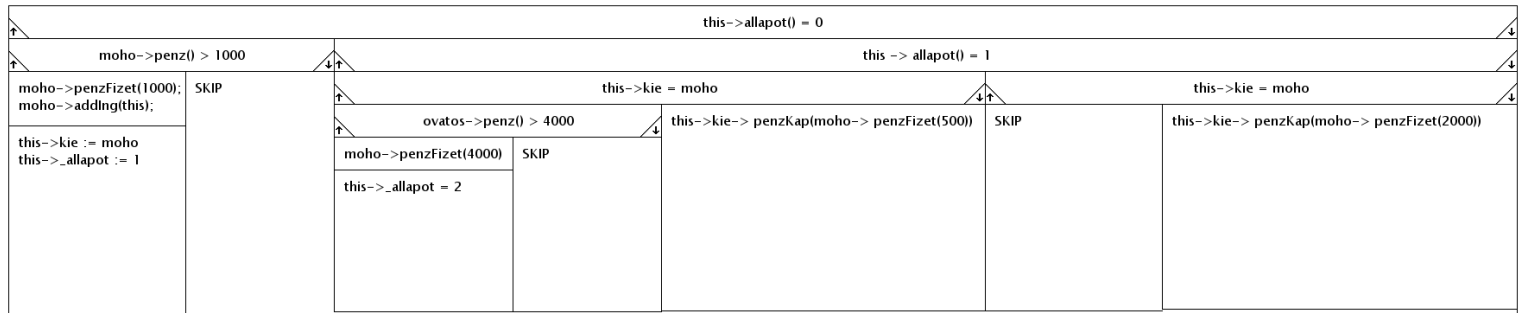
main:



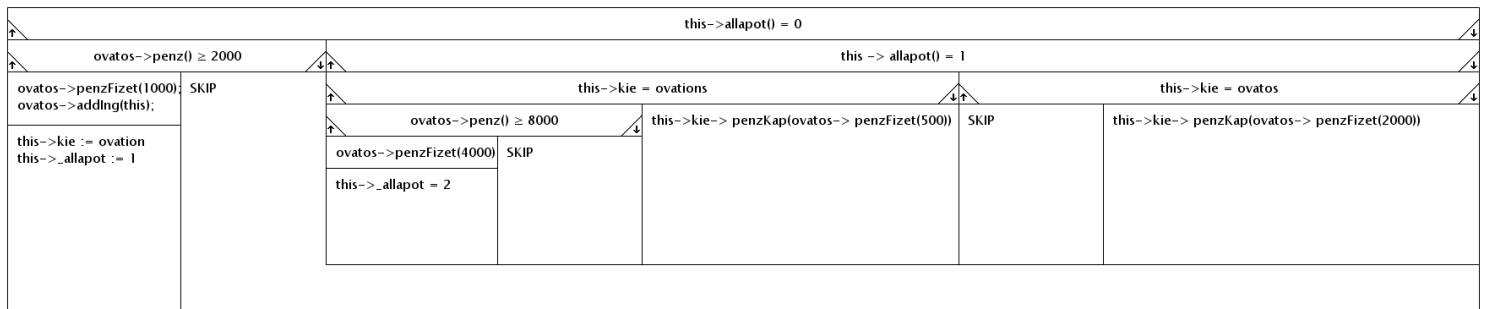
Az *akcioMoho*, *akcioOvatos* és *akcioTaktikus* metódusok Szolgáltatás és Szerencse esetén levonják és hozzáadják a megfelelő értékű pénzt a játékos egyenlegéhez.

A *penzFizet(int)* : int metódus ha az egyenlegen a fizetendőnél több pénz van az egyenlegen, levonja ezt, és visszatér a kifizetett értékkel. Amennyiben nincs elegendő pénz: 1, felszabadítja a játékos által birtokolt ingatlanokat, mivel kiesett a játékból. 2., -1-gyel teszi egyenlővé az egyenlegét 3., visszatér azzal az értékkel, amennyit ki tudott fizetni. Ez abban az esetben fontos számunkra, ha a fizetés egy másik játékosnak megy végbe, de nem a teljes összeget fizeti ki.

akcioMoho::Ingatlan



akcioOvatos::Ingatlan



akcioTaktikus::Ingatlan



Tesztelési terv

Szürke doboz tesztesetek:

Alap létrehozás tesztelése

- 1., Alap pénze 10 000

Mezők tesztelése

- 2., Szolgáltatás mező tesztelése
- 3., Szerencse mező tesztelése

Játékosok tesztelése

- 4., Mohó tesztelése
- 5., Óvatos tesztelése
- 6., Taktikus tesztelése

Játék tesztelése

- 7., Fejben lejátszható játék tesztelése