

# Karakter árverés terv

PROGRAMOZÁS ALAPJAI 2. - NHF

ZSOLT SEBESTYÉN (RV6JJO)

# Tartalom

Specifikáció .....	2
Rövid leírás .....	2
Fájlok, külső adatok .....	2
Játék menete .....	2
Tesztelés .....	2
Terv .....	3
Adat struktúra .....	3
Jatek .....	3
Vevo .....	4
Licitalas .....	4
Raktar .....	4
Karakter .....	4
Főbb algoritmusok .....	4
Jatek::beolvas_fajlokbol, Jatek::kiiras_fajlba .....	4
Jatek::fomenu_inditas .....	4
Jatek::licitek_inditas .....	5
Licitalas::futtatas .....	5
Játék menete .....	<b>Hiba! A könyvjelző nem létezik.</b>

# Specifikáció

## Rövid leírás

Egyszemélyes, parancssoros árverős játék. Cél minél nagyobb profithoz jutni. Az árverésen egymás után raktárakért kell licitálni, amiben speciális karakterek vannak. Minden karakternek van egy egyedi értéke, amit a játék során lehet kideríteni/megsejteni. Egy licitálás során 3 gép ellen játszol. Ha egy raktárért te ajánlasz a legtöbbet megkapod a benne lévő karakterekkel együtt. Ezt rögtön el is adod és kiderül mennyit profitáltál vagy buktál rajta. A játékban akkor nyersz, ha elérsz egy bizonyos összeget.

## Fájlok, külső adatok

A speciális karakterek „karakter.dat” fájlból olvasható be. Ez a fájl a játék során nem változtatható, de kézzel átírható. Az első sor tartalmazza a karakterek számát. A többi sor tartalmazza egy karakter adatait szóközzel elválasztva: KARAKTER ÉRTÉK GYAKORISÁG. Ha ez a fájl nem létezik a játék a legelején hibát dob. Ennek hiányát kézzel kell megírni.

A játékos adatait a „jatekos.dat” fájl tartalmazza. Ez a játék során is újra generálódik. Ez teszi lehetővé a program bezárása után is onnan lehessen folytatni a játékot, ahol abba hagyta a játékos. Ez a fájl akkor frissül amikor a felhasználó a menü-vel lép ki a programból. Az adatok külön sorban vannak elválasztva: JÁTSZOTT\_RAKTARAK\_SZAMA JÁTEKOS\_EGYENLEGE MEGVETT\_RAKTARAK\_SZAMA JÁTEKOS\_NEVE. Ha ez a fájl nem létezik a játék nem dob hibát és a játék során a fájl létrejön.

## Játék menete

1. **Program indítása:** alapadatok beállítása (nem első játék esetén korábbi adatok betöltése)
2. **Főmenü:** választhatod, hogy új játékot kezdesz, a régit folytatod (ha már játszottál), kiírod a játék szabályait vagy mentesz és kilépsz
3. **Raktár licitálás:**
  - a. Kirajzol egy új raktárt, benne található speciális karaktereket és kiírja a kezdőárat
  - b. játékos és a 3 robot egymás után jöve licitál vagy kiszál
  - c. aki utoljára bent marad megkapja a raktárt
4. **Profitálás:** ezután rögtön kiderül mennyit ér a raktár és mennyit profitáltál vagy vesztettél
5. **(Vége):** ha elfogy a pénzed vagy eléred a maximális összeget vissza ugrik a 2. ponthoz máskülönben megy tovább
6. **Menübe vissza vagy folytatás:** választhatod, hogy abba hagyod a játékot és visszamész a főmenübe (2. pont) vagy méysz a következő raktárhoz (ugrás a 3. ponthoz)

## Tesztelés

Tesztelés módban több fajta teszt futtatására van lehetőség, ami a program egészét teszteli hibás bemenetekre is. Ezt a TEST\_MODE makróval lehet beállítani, majd a parancssorból kiválasztani a futtatni szánt tesztfüggvény sorszámát.

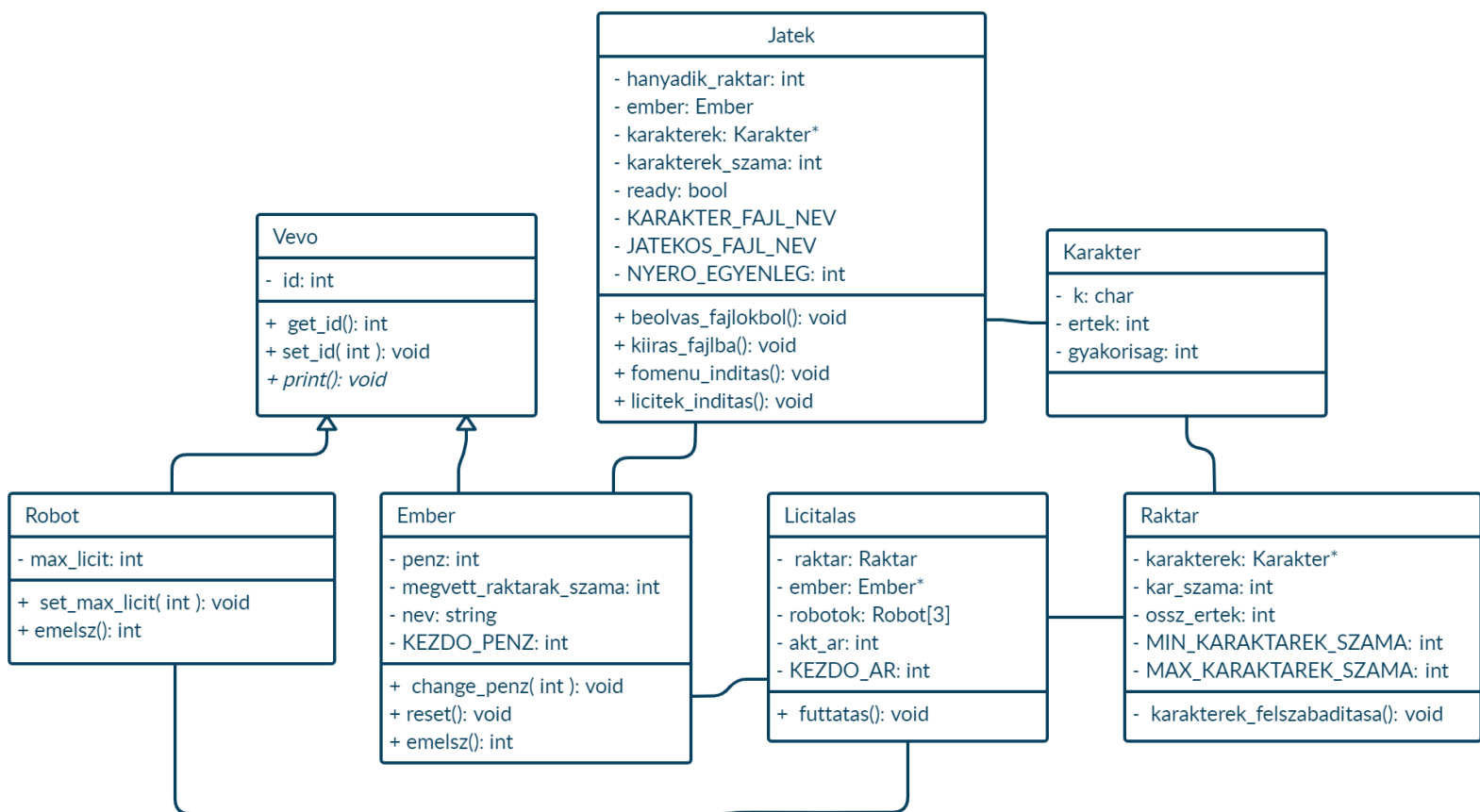
# Terv

## Adat struktúra

A program OO elvek mentén van tervezve. 5 különböző osztályra van bontva, amik a következők:

- Játék (Jatek)
- Vevő (Vevo): 2 gyerek osztállyal: Ember (Ember), Robot (Robot)
- Licitálás (Licitalas)
- Raktár (Raktar)
- Karakter (Karakter)

Ezek részletesebb specifikációját és kapcsolatát a következő UML diagram ábrázolja:



## Jatek

Ebben az osztályban található meg minden olyan fontosabb adat, ami a játék folyamatához szükséges. A „karakterek” és a „karakterek\_szama” tartalmazza azokat a karaktereket, amik a játék folyamán használhatóak (fájlból beolvasva). Az „ember” tartalmazza a játékos adatait, a „hanyadik\_raktar”, hogy hány raktárra lehetett már licitálni és a „ready” jelzi, ha az objektum készen áll a játék indításához (szükséges fájlok sikeres beolvasása esetén).

## Vevo

Ez az osztály, ami „részt vesz” a licitálásokon. 2 gyerekosztálya a Robot és az Ember, hisz ez a 2 fajta részvétel lehet (a licitáláson mindig 1 Ember és 3 Robot vesz részt). Az „id” a vevők azonosítója, ami mindenkinél egyedi. A Robot „max\_licit” paramétere tartalmazza az aktuális raktárért fizetendő maximum árat. Ezt a Robot egy új licitálás elején rögtön „kitalálja” és e fölé nem megy. Az Ember osztályban a játékos adatai találhatóak. A „nev” játékos felhasználónevét (szóköz nélkül), a „penz” az aktuális egyenlegét (Ft-ban) és a „megvett\_raktarak\_szama” pedig a már megvett raktarak számát tartalmazza.

## Licitalas

A játék során egymás utáni licitálások zajlanak le, ami által mindig új „Licitalas” objektumok jönnek létre. Az ember paraméter köti össze a játékost a licitálással. A többi paramétere a konstruktorban „generálódik le”. Ilyen például a „robotok”, ami 3 Robot-ot tartalmazó tömb. A „raktar” paraméterben található a licitálásra váró Raktar és az „akt\_ar” mindig a raktár aktuális árát (azaz a licit állását) tartalmazza.

## Raktar

Ez az osztály tartalmazza egy licitálható raktár adatait. A „karakterek” egy olyan Karakter tömb, ami az elérhető Játék karaktereiből generálódik le (random), a „kar\_szama” pedig ennek a tömbnek a mérete. Az „ossz\_ertek” menti el, hogy mennyi az összes karakternek az értéke (Ft-ban).

## Karakter

Ez egy karakter tulajdonságát írja le. A „k”-ban található maga a karakter „kinézete”. Az „ertek” az értékét (Ft-ban), a „gyakorisag” pedig a gyakoriságát (%-ban) tartalmazza.

## Főbb algoritmusok

Az osztályok legfontosabb algoritmusai, amik nagyban befolyásolják a program működését.

### Jatek::beolvas\_fajlokbol, Jatek::kiiras\_fajlba

Ez a 2 függvény felelős a fájlok kezeléséért. A beolvas\_fajlokbol a program elején beolvassa a „karakter.dat” és a „jatekos.dat” fájlokból az adatokat. Majd a kilépéskor a kiiras\_fajlba függvény írja ki az új játékos adatait a „jatekos.dat” fájlba (karakter.dat fájl adata nem változik).

### Jatek::fomenu\_inditas

Ez futtatja a főmenüt ameddig a felhasználó ki nem lép. ('q'). Megjelenik az összes lehetőség, amit játékos választani tud, majd a bemenettől függően meghívja a további függvényt. A lehetséges bemenetek a következők:

- n: új játék indítása
- c: meglévő játék folytatása (csak ha már fájlba ki volt mentve régebbi adat)
- i: info adatok kiírása
- q: mentés és kilépés

## Jatek::licitek\_inditas

Ez a függvény addig generál új liciteket és indítja el őket ameddig a felhasználó vissza nem lép a főmenübe. Minden licitálás után megkérdezi, hogy szeretné-e folytatni egy következő raktár licitálással vagy visszalép (lehetséges bemenetek: i vagy n). Ha visszalép a főmenübe, leteszteli, hogy elérte-e megnyeréshez szükséges értéket és ha igen, kiírja azt.

## Licitalas::futtatas

Itt bonyolódik le a licitálás. Először is kirajzolja a raktárt és kiírja az adatokat (pl. aktuális ár, játékos egyenlege). Majd addig kérdezgeti a Vevőket, hogy mennyivel emelnek ameddig az a következők közül az egyik be nem teljesül:

- az ember passzol (0-val emel)
- a 3 robot passzol (az ember megnyeri a raktárt)

Ha az ember nyer hozzáadja a profitot az egyenlegéhez (lehet mínusz is) és kiírja az új adatokat.