eva2

Első beadandó  
1. Feladat  
Gonda Dávid  
BIXU0S  
[gonda.david18@gmail.com](mailto:gonda.david18@gmail.com)

# Feladat leírása

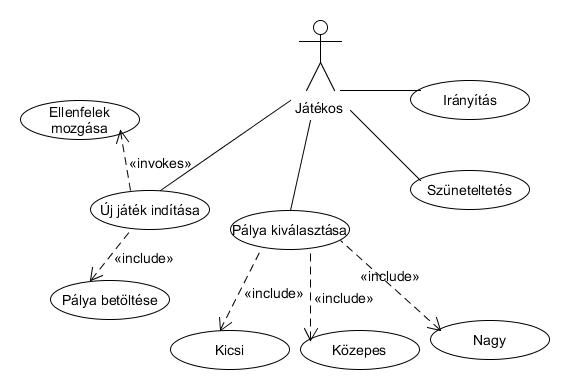
## Maci Laci

Készítsünk programot, amellyel a következő játékot játszhatjuk.  
Adott egy 𝑛 × 𝑛 elemből álló játékpálya, amelyben Maci Lacival kell piknikkosarakra vadásznunk. A játékpályán az egyszerű mezők mellett elhelyezkednek akadályok (pl. fa), valamint piknikkosarak. A játék célja, hogy a piknikkosarakat minél gyorsabban begyűjtsük.  
Az erdőben vadőrök is járőröznek, akik adott időközönként lépnek egy mezőt (vízszintesen, vagy függőlegesen). A járőrözés során egy megadott irányba haladnak egészen addig, amíg akadályba (vagy az erdő szélébe) nem ütköznek, ekkor megfordulnak, és visszafelé haladnak (tehát folyamatosan egy vonalban járőröznek). A vadőr járőrözés közben a vele szomszédos mezőket látja (átlósan is, azaz egy 3 × 3-as négyzetet.   
A játékos kezdetben a bal felső sarokban helyezkedik el, és vízszintesen, illetve függőlegesen mozoghat (egyesével) a pályán, a piknikkosárra való rálépéssel pedig felveheti azt. Ha Maci Lacit meglátja valamelyik vadőr, akkor a játékos veszít  
A pályák méretét, illetve felépítését (piknikkosarak, akadályok, vadőrök kezdőpozíciója) tároljuk fájlban. A program legalább 3 különböző méretű pályát tartalmazzon.  
A program biztosítson lehetőséget új játék kezdésére a pálya kiválasztásával, valamint játék szüneteltetésére (ekkor nem telik az idő, és nem léphet a játékos). Ismerje fel, ha vége a játéknak, és jelezze, győzött, vagy veszített a játékos. A program játék közben folyamatosan jelezze ki a játékidőt, valamint a megszerzett piknikkosarak számát

# Elemzés

* Három pálya lesz elérhető, egy kicsi, egy közepes, és egy nagy. Az alapértelmezett a közepes.
* Az ablak 2 menüpontból fog állni, egy Új játék-ból, és egy Pálya méret-ből, amiben helyet foglalnak a méretek. A kiválasztott méret mellett egy check mark jelenik meg. Az ablak alján található lesz egy eltelt időt mutató label, és egy pontszámot mutató label.
* A játék területét dinamikusan hozzuk létre, egy NxN-es label rácsszerkezettel.
* A játékos a nyilakkal tud mozogni a játéktéren, az ellenfelek pedig a bemenet által meghatározott módon, másodpercenként mozognak egy mezőt.
* Győzelem és vesztés eesetén a játék feldob automatikusan egy dialógusablakot, a megfelelő üzenettel ellátva.

# Felhasználói esetek



# Program szerkezete

### Tervezés:

A program háromrétegű architektúrában (M/N/P) épül fel.

### Perzisztencia:

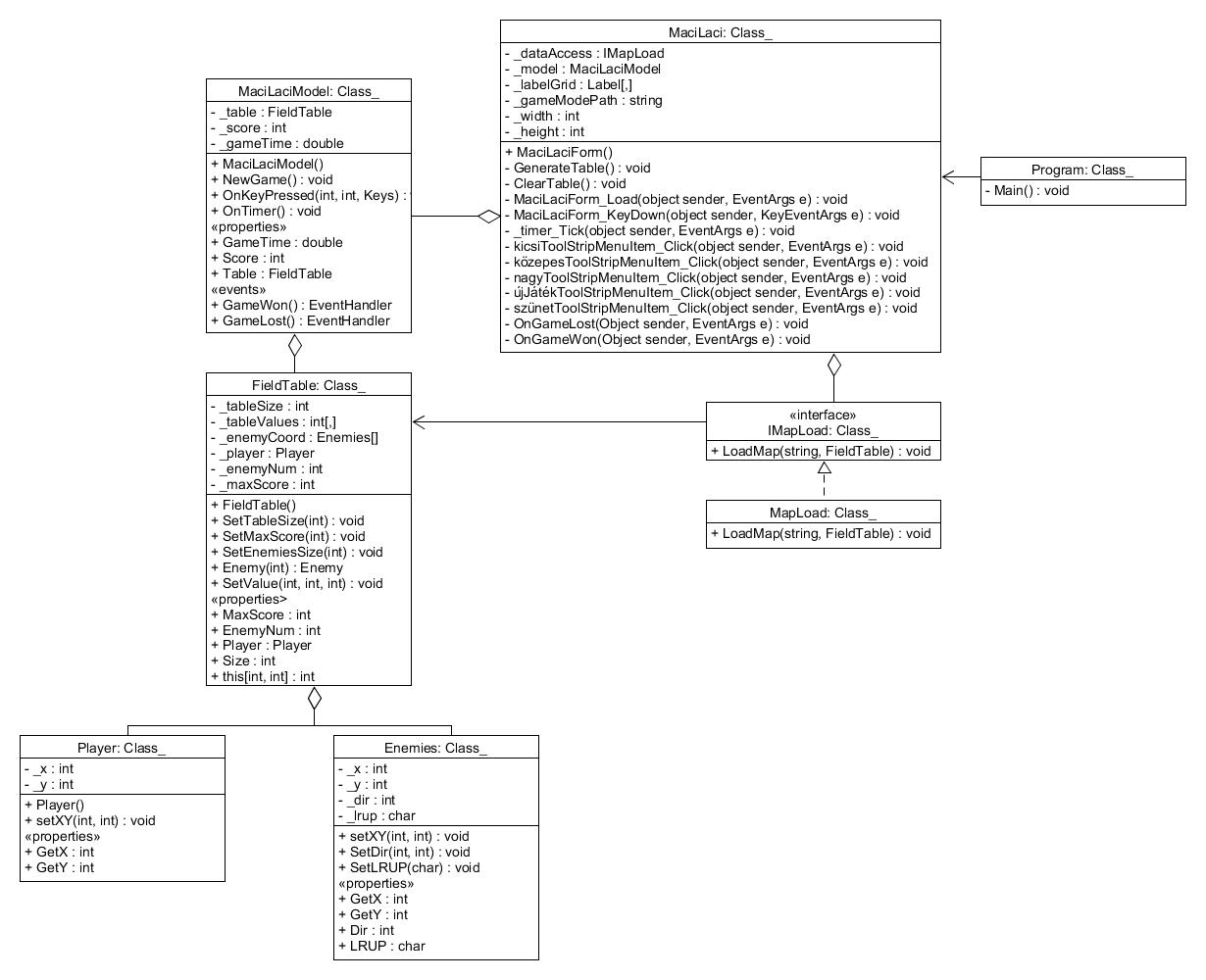
A pálya tárolásáért/betöltéséért less felelős. A játékos, és az ellenfelek pozícióját is itt tároljuk.

### Modell:

A lényegi játéklogikát tárolja, módosítja a táblát, a játékos koordinátáit, és az ellenfelek koordinátáit.

### Nézet:

A kinézetet kezeli, szigorúan a modell adataiból rajzolja ki a játéteret minden lépésben.



# Tesztesetek

* A modell főbb részeit egységtesztek segítségével ellenőrizzük a MaciLaciModelTest osztályban.
* Az alábbi tesztesetek kerültek megvalósításra:
  + **- \_timer\_TickTest():**Az ellenfelek lépését, illetve a játékidő növelését ellenőrzi.
  + **- KeyDownEvent():**  
    A játékos lépésének helyességét ellenőrzi.