1. beadandó feladat

Készítette:

Mikus Márk

NK-kód: CM6TSV

email: [kyussfia@gmail.com](mailto:kyussfia@gmail.com)

# Feladat:

## Fénymotor párbaj

Készítsük programot, amellyel a Tronból ismert fénymotor párbajt játszhatjuk. Adott egy 𝑛 × 𝑛 elemből álló játékpálya. A két játékos a bal, illetve jobb oldal közepén indul egy-egy fénymotorral, amely egyenesen halad (rögzített időközönként) a legutoljára beállított irányba (függőlegesen, vagy vízszintesen). A motorokkal lehetőség van balra, illetve jobbra fordulni. A fénymotor mozgás közben fénycsíkot húz, ami a játék végéig ott marad. Az a játékos veszít, aki előbb nekiütközik a másik játékos motorjának, bármelyikük fénycsíkjának vagy a pálya szélének. A program biztosítson lehetőséget új játék kezdésére a pályaméret megadásával (12 × 12, 24 × 24, 36 × 36), valamint játék szüneteltetésére (ekkor nem telik az idő, és nem mozognak a motorok). Ismerje fel, ha vége a játéknak, és jelenítse meg, melyik játékos győzött.

# Elemzés:

* A játékot három pályamérettel játszhatjuk: kicsi (12 x 12), közepes (24 x 24), nagy (36 x 36).
* A feladatot egyablakos asztali alkalmazásként Windows Forms grafikus felülettel valósítjuk meg.
* Az ablakban elhelyezünk egy Új játék menüt a következő menüpontokkal: Kis, Közepes, Nagy. Ahol a felhasználó a pályaméretet választhatja ki. A felületen helyet kap még egy óra, amelyen az eltelt időt lehet követni, továbbá egy szüneteltetésre való gomb, amelynek kezdetben nincs funkciója, csak a játék közben.
* A játéktáblát egy a n x x-es gombrács fogja reprezentálni, a gomboknak kattintás eseménye nincsen.
* Egy implementált időzítő eseményeire reagál majd a nézet, s lépkednek a játékosok.
* A játékosok irányítása legyen:
  + Kék játékos balra: A
  + Kék játékos jobbra: D
  + Piros játékos balra: Bal nyíl
  + Piros játékos jobbra: Jobb nyíl
* A játék automatikusan feldob egy dialógusablakot, amikor vége a játéknak (Döntetlen vagy Győzelem esetén), s megáll az időmérő.

# 

# Tervezés:

### Programszerkezet

A programot kétrétegűarchitektúrában valósítjuk meg: a megjelenítés a View, azaz Nézet, míg a játéklogikáért a Model komponens lesz felelős.

### Modell

A modell lényegi részét a **LightDuelModel** osztály adja. A könnyebb karbantartás végett egyéb segédtípusok kerültek bevezetésre. A **Players** felsoroló a mezők lehetséges értékeit (No, Blue, Red), míg a Player típus magát a játékos objektumot reprezentálja. A Player típusnak 4 tagja van: egy oszlop és egy sorszám, egy irány és egy típus, hogy a játékos melyik színt képviseli. Az irány a mozgása irányát a sorszámok pedig a pozícióját határozzák meg.

A mezők tárolására a **List <List <Players>>** lista szolgál. A időzítőt a **System.Windows.Forms.Timer** osztályból injektáltuk az osztályba.

A modell két alapvető eseménnyel rendelkezik: **ticked** és a **gameOver**. A **ticked** eseményt az időzítő váltja ki adott időközönként, amennyiben új játékot kezdünk a nézeten, a **newGame(size)** interfész metódus közvetítésével. A **gameOver** esemény akkor váltódik ki amikor vége a játéknak, valamilyen okból.

További két segédosztály kerül implementálása:

1. **PlayerMoveEventArgs**: A játéktábla frissítését segítő esemény argumentumok.
2. **GamoOverEventArgs**: A játék végeredményét közlő esemény argumentumok.

Ezen esemény argumentumok segítségével üzen a Model, a Nézetnek.

### Nézet

A nézet alapja a beépített **Form**. Ebből származik a **LightDuel\_Winforms.View** névtér alatt a **MainWindow** nézetosztály. Feladata, hogy reagáljon a modell változásaira, és azokat reprezentálja és szinkronizálja, hogy mindig a modellnek megfelelő játékállás tükröződjön a nézeten, azaz megfelelően színezze a játéktábla mezőit.

A nézet része a beinjektált model példány, egy gombrácsokat tároló tömb, **isPaused** változó a szünet, **disbledKeys** a billentyűesemények figyelésére, illetve a nézet konfigurációk (játékos billentyűk).

A nézet része továbbá egy primitív órát példányosító objektum (**Clock**), amely egész számok segítségével számolja az eltelt időt.

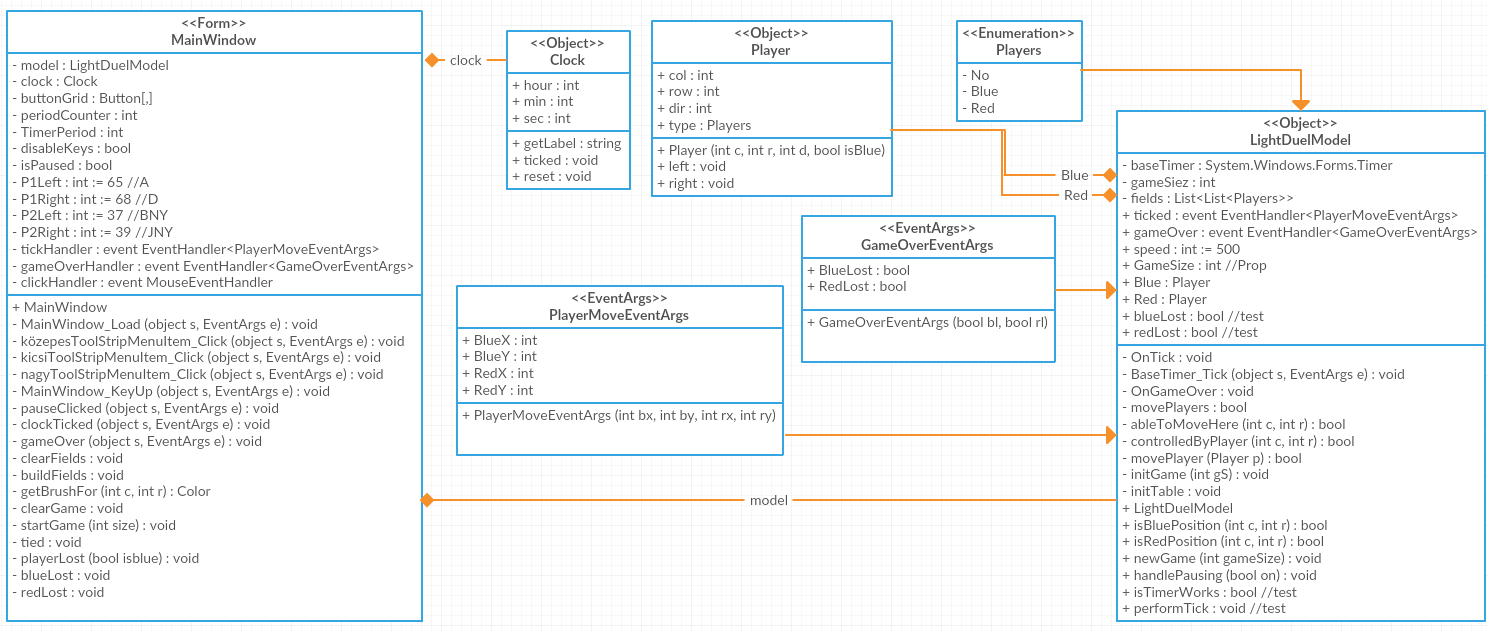
Közvetlen eseményekkel is rendelkezik a nézet:

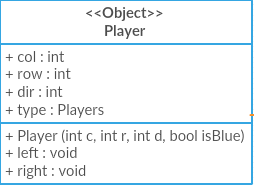
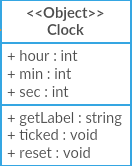
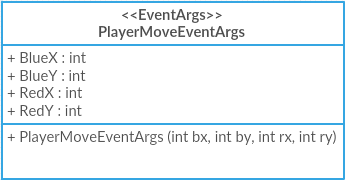
* **közepesToolStripMenuItem\_Click**: Közepes mérettel indul a játék
* **kicsiToolStripMenuItem\_Click**: Kicsi mérettel indul a játék
* **nagyToolStripMenuItem\_Click**: Nagy mérettel indul a játék
* **MainWindow\_KeyUp**: Billentyűesemények kezelése (játékosok)
* **PauseClicked**: Szüneteltetés gomb eseménye.

Ezenkívül három fontos eseménykezelője is van:

* **tickHandler**: egy **EventHandler<PlayerMoveEventArgs>** példány az időzítő jelzéseinek elkapására.
* **gameOverHandler**: egy **EventHandler<GamoOverEventArgs>** példány a játék végéhez.
* **clickHandler**: egy **MouseEventHandler** a szüneteltetés észleléséhez.

Amennyiben Döntetlen vagy Győzelem született a nézet egy **MessageBox**-ban üzen, s kiírja az eltelt idő és az eredményt is szövegesen.



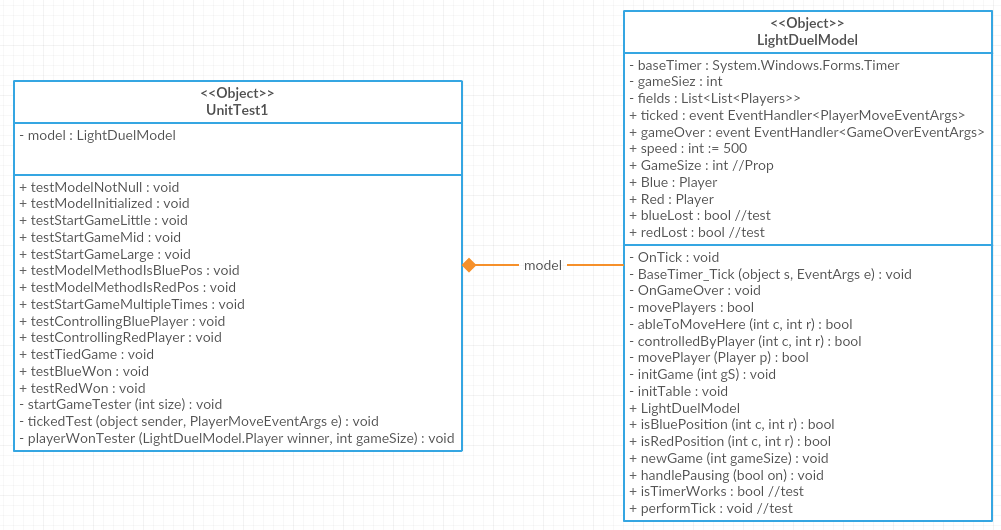


# 

# Tesztelés:

### A modell funkcionalitása egységtesztek segítségével lett ellenőrizve a LightDuelModelTest névtér alatti UnitTest1 nevű osztályban.

### Az alábbi tesztesetek kerültek megvalósításra:

* + testModelNotNull
  + testModellInitialized
  + testStartGameLittle
  + testStartGameMid
  + testStartGameLarge
  + testModelMethodIsBluePos
  + testModelMethodIsRedPos
  + testStartGameMultipleTimes
  + testPauseGame
  + testControllingBluePlayer
  + testControllingRedPlayer
  + testTiedGame
  + testBlueWon
  + testRedWon