

<b>A</b>	Név:	Neptun kód:
	Értékelő tanár:	Jegy:

### Rubik tábla

Készítsünk egy WPF alkalmazás MVVM architektúrában, amellyel a következő játékot lehet játszani.

A Rubik tábla lényegében a Rubik-kocka kétdimenziós változata. A játékban egy  $n \times n$  mezőből álló táblán  $n$  különböző színű mező lehet, mindegyik színből pontosan  $n$  darab, kezdetben véletlenszerűen elhelyezve. A játék célja az egyes sorok, illetve oszlopok mozgatásával (ciklikus tologatásával, azaz ami a tábla egyik végén lecsúszik, az ellentétes végén megjelenik) egyszínűvé alakítani vagy a sorokat vagy az oszlopokat (azaz vízszintesen, vagy függőlegesen csíkokat kialakítani).

### Részfeladatok:

1. A program jelenítse meg a játéktáblát, amelynek méretét ( $3 \times 3$ ,  $4 \times 4$ , vagy  $6 \times 6$ ) a felhasználó választhatja ki. Legyen lehetőség új játékot kezdeni, amelyhez az alkalmazás megkeveri a kezdetben sorban elhelyezett színeket (tehát a megoldásból indulunk ki). A keverés a szabályos játéklépés  $n^3$  alkalommal véletlenszerűen kiválasztott mezőkön történő alkalmazásával történjen. Legyen lehetőség lépni a játékban (azaz egy sort/oszlopot eltolni a négy irány valamelyikében).
2. A játék ismerje fel, ha vége a játéknak (sorrendbe kerültek a számok), ekkor előugró üzenettel mutassa meg a játékos lépéseinek számát, és automatikusan kezdjen új játékot.
3. A program folyamatosan számolja a játékos játékidejét, és ezt jelenítse meg a felületen.
4. Legyen kiszűrős a játék. Minden  $n$  lépés után a program automatikus kicserél két véletlenszerűen kiválasztott bábút. Garantáljuk, hogy a két bábu mindig különbözik.

A megoldást az elfogadást követően **<Neptun-kód>.zip** formátumban fel kell tölteni a `\\nas2.inf.elte.hu\zh\eva2\` címre.

**Jó munkát!**