

## Neptun kód: ASPP08

### Feladat

Készítsünk programot, amellyel egy Rubik táblát lehet kirakni.

A Rubik tábla lényegében a Rubik-kocka két dimenziós 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).

A program biztosítson lehetőséget új játék kezdésére a táblaméret (és így a színek számának) megadásával ( $2 \times 2$ ,  $4 \times 4$ ,  $6 \times 6$ ), és ismerje fel, ha vége a játéknak. Ekkor jelenítse meg, hány lépéssel győzött a játékos, majd kezdjen automatikusan új játékot.

### Elemzés<sup>1</sup>

Minden Field jellemzői: a színe (Color), típusa (String)

Minden Board jellemzői: matrix (Field[][]), mérete (int)

### Terv<sup>2</sup>

Szükségünk lesz egy BoardGUI osztályra a boardSize konstruktoron keresztül fogjuk megadni. Ezen felül létre kell hoznunk a megfelelő metódusokat, newRefresh(String direction,int line) , oldRefresh(), getBoardPanel () az osztályon belül létre kell hozni egy másik osztályt: ButtonListener , metódusai ButtonListener(int x, int y), actionPerformed(ActionEvent e)

Szükségünk lesz továbbá egy Board osztályra, metódusai: Field(String type), getColor(), setColor(Color color), getType()

Szükségünk lesz továbbá egy Fieldosztályra, metódusai: isOver(),Field get(int x, int y), Field get(Point point), getBoardSize()

Szükségünk lesz továbbá egy RubikTableGUI osztályra, ami a konstruktorában létrehozza a Board-ot

---

<sup>1</sup> Ez az elemzés rész a hallgatói beadandó dokumentációjából elhagyható, az átalakítási táblázatokat a tervezés részben elég feltüntetni.

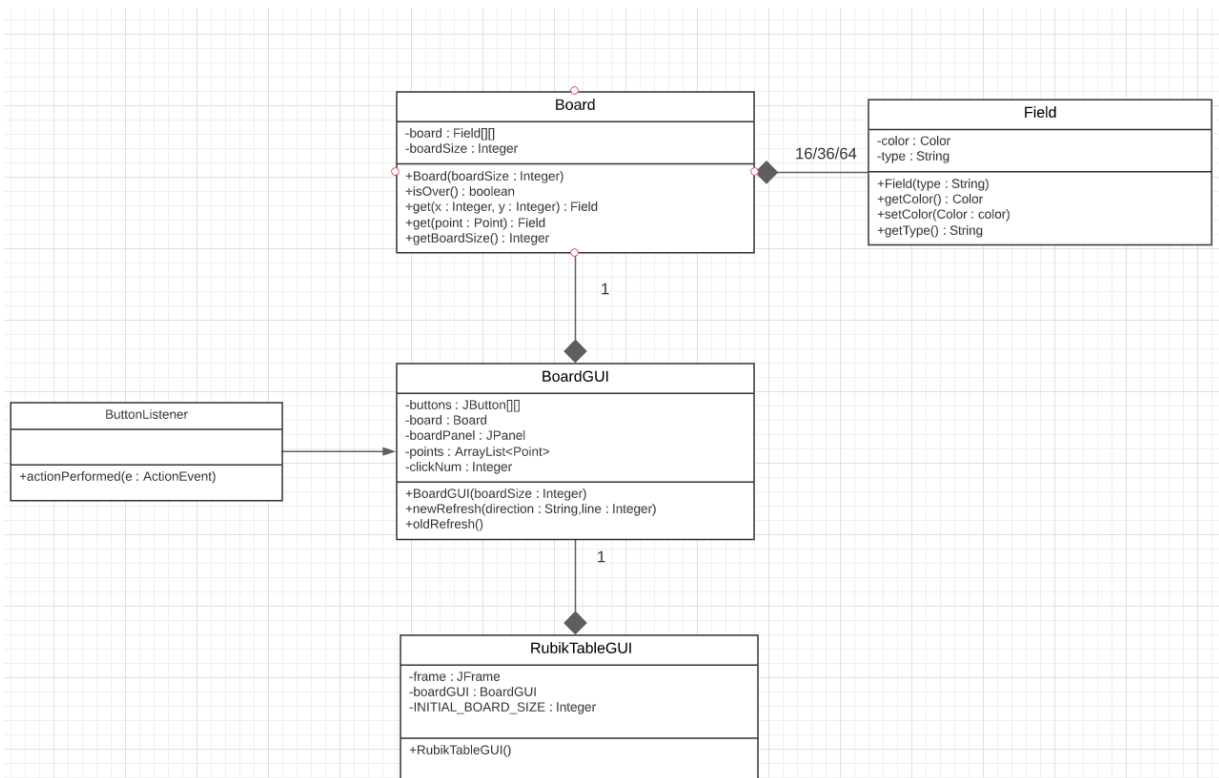
<sup>2</sup> A szöveges magyarázatra a hallgatói beadandók dokumentumaiban nincs szükség.

A Board isOver() metódusa fogja ellenőrizni a játék végét(sor/oszlop szerint kirakva).

A BoardGUI newRefresh(String direction,int line) metódusa fogja léptetni a színeket

A BoardGUI oldRefresh() metódusa fogja beállítani a színeket kezdéskor

A ButtonListener actionPerformed() metódusa fogja a gombokat ellenőrizni,és meghívni a newRefresh metódust



## Tesztelési terv

2x2 kirakása

4x4 kirakása

6x6 kirakása