Készítette:
Kocsis Ábel
Neptun kód: FGSDV2
E-mail: kocsis.abel.98@gmail.com

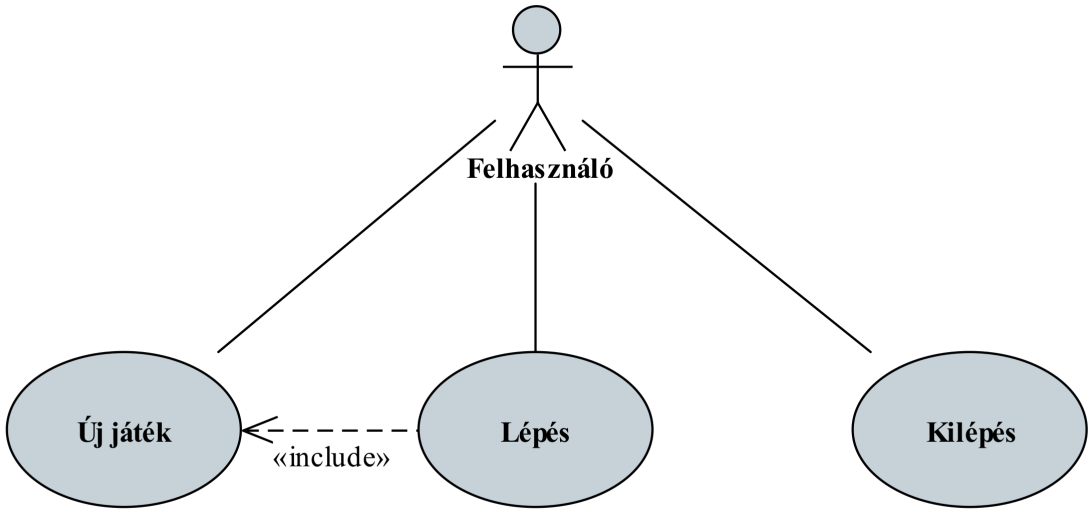
Feladat:

Készítsünk programot, amellyel a következő játékot lehet játszani.
Adott egy $n \times n$-es tábla, melyen királynőket helyezhetünk el sorban egymás után. A tábla kezdetben üres, és a játék célja, hogy elhelyezzünk n királynőt úgy, hogy azok közül semelyik kettő ne üsse egymást (vízszintesen, függőlegesen, vagy átlósan). Minden elhelyezés után jelöljük meg a táblán azokat a mezőket, ahova már nem rakhatunk újabb királynőt (amelyeket az eddig elhelyezett bábúk ütnek), és természetesen ne is engedjük ezeket a mezőket használni. A lehelyezett királynőt lehessen visszavenni, ekkor a program szabadítsa fel a megfelelő mezőket.
A program biztosítson lehetőséget új játék kezdésére a táblaméret megadásával (4 × 4, 6 × 6, 8 × 8), és ismerje fel, ha vége a játéknak. Ekkor jelenítse meg, hány lépéssel győzött a játékos (a levételek is lépésnek számítanak), majd kezdjen automatikusan új játékot.

Elemzés:

- A játékot egy grafikus felületen jelenítjük meg, ahol a táblaméretnek megfelelő számú nyomógombot helyezünk el tábla gyanánt. A nyomógombhoz közös eseménykezelőt rendelünk, amely egérkattintás hatására megjeleníti a megfelelő színeket. Ahová királynőt helyeztünk, zöld színű lesz, míg az általa ütött mezők pirosak. A lépésszámot és a visszamaradó lerakható királynőket számoljuk. Amennyiben már van királynő a mezőn, akkor ezt eltávolítjuk, és az általa ütött mezőket szükség szerint átállítjuk. Az ütés alatt lévő mezőt inaktívvá tesszük.
- Az ablak tetején helyet kap a "Tábla mérete" beállító és az "Új játék" gomb, amellyel bármikor új játékot kezdhetünk. Így a felületen összesen méret*méret+1 gomb helyezkedik el.
- A játék felületét fix méretűre készítjük el. Minden esetben négyzetben jelenítjük meg, a négyzet oldala méret*75 pixel hosszú.
- A játék állását egy egészeket tartalmazó mátrixban tároljuk. Ahol királynő áll, ott 1-es, szabad mező 0-s, ütés alatt álló mező pedig negatív egész. Az ütés alatt állók abszolút értéke megmutatja, hogy hány királynő üti azt a mezőt.
- Egy-egy egész szám típusú változóban tároljuk a méretet és a hátralévő elhelyezendő királynők számát, illetve az eddig megtett lépések számát. A mező összes gombját egy vektorban is eltároljuk, hogy azokat szükség esetén tödölni tudjuk a felületről.

Használati esetek:



	Felhasználói eset	Leírás	
1	Alkalmazás indítása	GIVEN:	az alkalmazás telepítve van
		WHEN:	alkalmazás indítása
		THEN:	4x4-es játékfelület megjelenik
2	Kilépés	GIVEN:	játék felület
		WHEN:	játék felület ablakának lezáró ikonjára kattint
		THEN:	alkalmazás befejezése
3	Lerakás	GIVEN:	játék felület
		WHEN:	a játék felület fehér mezőjére kattint
		THEN:	Lerak egy királynőt, zöld színnel megjelenik, az általa ütött mezők pirosak lesznek
4	Felvétel	GIVEN:	játék felület
		WHEN:	a játék felület zöld mezőjére kattint
		THEN:	a mezőről felveszi a királynőt, az fehérre vált, az általa ütött mezők ha másik nem üti, felszabadulnak, fehérre váltanak
5	Játék vége	GIVEN:	játék felület
		WHEN:	az összes királynő lerakásra került
		THEN:	kiírja hány lépésből sikerült nyerni, új játékot kezd

Tervezés:

A program lényegi váza a **`Kiralynok`** grafikus felület osztály, amely a játék felületét a **`buttonTable: QVector<QVector<QPushButton>>`** mátrixban, míg magukat az értékeket a **`gameTable: Int32[][]`** mátrixban tárolja.
A lépések számolásáért a **`stepCount: int`**, a hátrelévő királynők számolásáért pedig a **`hatra: int`** felel.

Az eseményvezérlőkön túl az új játék kezdését a **`newGame`**, a lerakás, felvétel végrehajtását a **`stepGame`**, a játék végének ellenőrzését a **`checkGame`** metódusok hajtják végre. Az ütött mezők beállításáért a **`disableButton`**, míg a felszabadított mezők beállítását az **`enableButton`** végzi. Új játék indításakor a **`generateTable`** generál új játéktáblát.

Osztályszerkezet:

QWidget **Kiralynok**
- stepCount :int - size :int - hatra :int - tableLayout :QTableLayout* - mainLayout :QVBoxLayout* - newGameButton :QPushButton* - buttonTable :QVector<QVector<QPushButton*>> - _label :QLbale* - _spinbox :QSpinBox* - _buttonGrid :QVector<QPushButton*> - gameTable :int**
+ Kiralynok :(QWidget*) - buttonClicked() :void - newGameButtonClicked() :void - newGame() :void - stepGame(int, int) :void - disableButton(int, int) :void - enableButton(int, int) :void - generateTable() :void - checkGame() :void

Eseményvezérlés:

- **`newGameButtonClicked`**: új játék gomb kattintása.
  - **Forrás:** az új játék gomb **`Click`** eseménye.
  - **Feladata:** új játék kezdése, azaz a játéktábla újrainicializálása
- **`buttonCLicked`**: egérkattintás kezelése a játéktáblán.
  - **Forrás:** gombmátrix valamennyi gombjának **`Click`** eseménye.
  - **Feladata:** a küldő gomb megállapítása, és az alapján a színek megfelelő változtatása, a lépésszám növelés, valamint a játék állásának ellenőrzése.