

# Quoridor Specifikáció

## Játékmenet és szabályok:

A játékot általában 2, de akár 4 játékos is játszhatja. Minden bábu a pálya egyik szélén kezd és az a cél, hogy elérjen a szemben levő szélére a pályának. A játékosok felváltva következnek. Egy körben a soron lévő játékos **egyetlen** akciót hajthat végre a következők közül:

- Bábu mozgatása:** A bábút egy szomszédos, üres mezőre mozgathatja (fel, le, jobbra, balra, átlósan nem).

- i. A bábu nem léphet olyan mezőre, amit fal blokkol.

- ii. **Ugrás szabály:** Ha két bábu egymással szemben áll egy szomszédos mezőn, a soron lévő játékos átugorhatja a másik bábút, így két mezőt haladva. Ha az átugrott bábu mögött közvetlenül egy fal van, a játékos a bábu mellé ugorhat (átlósan egyet).

- Fal lehelyezése:** A játékos elhelyezhet egy falat a táblán, hogy akadályozza az ellenfelet (vagy saját magát). Összesen minden játékosnak 10 fala van.

- i. A falnak pontosan két mezőt kell érintenie.

- ii. Falak nem keresztezhetik egymást.

- iii. Tilos olyan falat letenni, ami bármelyik játékos számára teljesen elzárja az utat a céljáig. Minden bábu számára mindig kell maradnia legalább egy lehetséges útvonalnak a célterületre. Ezt a játék logikájának minden fal lehelyezése előtt ellenőriznie kell.

- A felhasznált falak számát a programnak nyilván kell tartania. Ha egy játékosnak elfogytak a falai, már csak bábút mozgathat.



## Tervezett adatstruktúrák:

**Bábuk:** A bábuokról csak a pozíciót kell tárolni, amit egy két dimenziós vektorban lesz tárolva, minden bábunak sajátként.

**Falak:** A falakat két különböző két dimenziós listában fogom tárolni, egy a függőleges egy pedig a vízszintesen rakható falaknak. Ez így kettő 8x8 as tömböt jelent egy 9x9 méretű pályán.

Ezek a struktúrákon implementálni kell egy útkereső algoritmust, ahhoz hogy meg lehessen határozni, hogy egy falrakás legális e. Ennek az implementálásához BFS vagy DFS algoritmust is lehet használni.

**Felhasználói felület:** A java swing gui elemeit is Összetettebb struktúrákban kell majd tárolni, illetve mindig a fenti adatok alapján újraépíteni vagy kiegészíteni őket, a falak lerakása vagy játékos mozgása után.

## Öröklések:

**Bábuk:** Lesz egy absztrakt bábu osztály ami tartalmaz minden szükséges adatmezőt és fontos függvényt.

Például a move() függvénye csak a leszármazottakban lesz implementálva.

Ettől az absztrakt osztálytól fog örökölni a játékos által vezérelt bábu és a számítógép által vezérelt bábu is, amik különbözőféleképpen fogják megvalósítani a játékos által adott bemenetre váró függvényeket.

**SwingUI:** Bizonyos elemeihez készítek saját osztályt ami örököl a swinges változatától, annak érdekében hogy áttekinthetőbb legyen a program. Például a GameFrame osztály örököl a JFrame-től, és a konstruktorában minden szükséges beállítást el lehet végezni. Hasonlóan a játékosmezővel is el lehet járni, ami előre láthatólag a JPanel től fog örökölni.

## További Funkciók:

A programban lehetőség lesz az aktuális játék állását elmenteni, és később visszatölteni.

Új játékot kezdeni kettő vagy négy játékosal, vagy egy játékosal számítógép ellen.

Ezekhez a funkciókhoz is swing gui-s menük lesznek készítve.