

A PROGRAMOZÁS ALAPJAI 2.

HÁZI FELADAT DOKUMENTÁCIÓ

SAKK TÁBLAJÁTÉK

KÉSZÍTETTE: KOCSIS LÁSZLÓ, IG72OK
Kocsislaci95@gmail.com

KÉSZÍTÉS FÉLÉVE: 2016/17/2

TARTALOMJEGYZÉK

Felhasználói dokumentáció	4
Osztályok statikus leírása.....	4
Game.....	4
Felelőssége.....	4
Ősosztályok.....	4
Attribútumok	4
Metódusok.....	4
Player	5
Felelőssége.....	5
Ősosztályok.....	5
Attribútumok	5
Metódusok.....	5
Piece.....	5
Felelőssége.....	5
Ősosztályok.....	5
Attribútumok	5
Metódusok.....	6
Pawn	6
Felelőssége.....	6
Ősosztályok.....	6
Attribútumok	6
Metódusok.....	6
Rook	6
Felelőssége.....	6
Ősosztályok.....	6
Attribútumok	7
Metódusok.....	7
Knight.....	7
Felelőssége.....	7
Ősosztályok.....	7
Attribútumok	7
Metódusok.....	7
Bishop	7
Felelőssége.....	7

Ősosztályok.....	7
Attribútumok	8
Metódusok.....	8
Queen	8
Felelőssége.....	8
Ősosztályok.....	8
Attribútumok	8
Metódusok.....	8
King	8
Felelőssége.....	8
Ősosztályok.....	8
Attribútumok	9
Metódusok.....	9
UML osztálydiagramm	10
Összegzés	10
Mit sikerült és mit nem sikerült megvalósítani a specifikációból?	11
Mit tanultál a megvalósítás során?	11
Továbbfejlesztési lehetőségek.....	11
Képernyőképek a futó alkalmazásról.....	12

Felhasználói dokumentáció

A házi feladatomban implementáltam egy sakk táblajátékot. Mivel grafikus környezetet nem hoztam létre ezért a játék konzolon játszható. A program indítása és egyedi azonosítók megadása után már lehet is látni a felrajzolt táblán a rajta elhelyezkedő bábukkal. A kis betűk a sötét, a nagy betűk a világos figurákra utalnak. A főmenüben elindíthatunk egy lépést vagy kiléphetünk a játékból. Az utóbbi választásával a program leáll. Ha a lépés indítását választjuk utána meg kell adni, hogy mely pozícióról mely másik pozícióra szeretnénk lépni. A pozíciót alakja egy kisbetű és egy szám egymás után, minden egyéb karaktert mellőzve. A lehetséges lépéseket ismerni kell ahhoz, hogy megengedett lépést tudjunk beadni a programnak, mivel alpból nem ajánlja fel a lehetséges eseteket. A program a beadott koordináták után leellenőrzi, hogy a lépés lehetséges-e és ha igen akkor megteszi azt, ha nem akkor hibát dob és újra kell kezdeni a lépés megadását.

Osztályok statikus leírása

Game

Felelőssége

Ez az osztály felel a felhasználói interakciókért, valamint egységbe foglalja a játékhoz szükséges elemeket.

Ősosztályok

Ez az osztály nem öröklődik.

Attribútumok

Privát

- blackPlayer – Player típusú játékos
- whitePlayer – Player típusú játékos
- board – 8x8-as char tömb. A tábla.
- isThereAWinner – logikai érték, a játék végét jelzi
- isWhitesTurn – logikai érték, a játékosok körét jelzi

Védett

Publikus

Metódusok

Privát

Védett

Publikus

- Konstruktor
- Destruktor
- playGame – a játék felhasználói interakcióit valósítja meg
- print – kirajzolja a táblát
- move – egy lépést valósít meg a bemenetek alapján, valamint vizsgálja, hogy lehetséges-e a lépés
- getPiceOnPosition – egy tábla pozícióról lekérdezi, hogy mi helyezkedik ott el
- setPiceOnPosition – elhelyez egy figurát a tábla egy adott pontjára
- getIsWhitesTurn – a kimenete, hogy éppen a világos vagy a fekete jön

Player

Felelőssége

Ez az osztály felel azért, hogy egy játékos és annak bábai azonosíthatók legyenek, valamint gondoskodik, hogy a felhasználó által megadott lépések szerint a bábuk a helyükre kerüljenek.

Össztályok

Ez az osztály nem öröklődik.

Attribútumok

Privát

- name – std::string típusú név, azonosító
- isWhite – logikai érték, a játékos színét adja meg
- isItsTurn – logikai érték, azt adja meg, hogy éppen ez a játékos jön-e
- pieces – Piece típusú pointereket tároló dinamikus tömb pointere

Védett

Publikus

Metódusok

Privát

Védett

Publikus

- Konstruktor
- Destruktor
- getIsWhite – visszaadja az isWhite adattagot
- getName – visszaadja az name adattagot
- getPieces – visszaadja a pieces adattagot
- setTaken – egyet a játékos bábai közül levesz az asztalról
- takeMove – lép egyet az egyik bábuval

Piece

Felelőssége

Egy bábu általános tulajdonságait és metódusait zárja egységbe. Absztrakt osztály ezért nem példányosítható. A ténylegesen példányosodó bábuk az ebből származtatott különböző figura osztályokból jönnek létre.

Össztályok

Ez az osztály nem öröklődik.

Attribútumok

Privát

Védett

- isTaken – logikai érték, ami megmondja, hogy adott bábu játékban van-e
- isWhite – logikai érték, ami megmondja, hogy adott bábunak milyen a színe
- character – char típusú változó, ami a bábu jelölését tartalmazza
- xpos – char típusú változó, ami az vízszintes tengely szerinti pozícióját tartalmazza a figurának
- ypos – char típusú változó, ami az függőleges tengely szerinti pozícióját tartalmazza a figurának

Publikus

Metódusok

Privát

Védett

Publikus

- Konstruktork
- Destruktork
- getCharacter – visszaadja a character adattagot
- getXPos – visszaadja az xpos adattagot
- getYPos – visszaadja az ypos adattagot
- setPos – megadott paraméterek alapján beállítja a pozíciót leíró adattagokat
- setTaken – a figura pozícióját átállítja, úgy, hogy ne legyen a pályán, valamint az isTaken adattagot igazzá teszi
- move – virtuális tagfüggvény, minden figura esetében más logikai vizsgálatok alapján működik

Pawn

Felelőssége

Egy speciális figura, gyalog.

Ősosztályok

Piece

Attribútumok

Privát

- isInstartPos – logikai érték, ami megmondja, hogy a figura a kezdeti pozíciójában van-e

Védett

Publikus

Metódusok

Privát

Védett

Publikus

- Konstruktork
- Destruktork
- move – adott paraméterek alapján lép a gyalog és ellenőrzik, hogy ezt megteheti-e

Rook

Felelőssége

Egy speciális figura, bástya.

Ősosztályok

Piece

Attribútumok

Privát

Védett

Publikus

Metódusok

Privát

Védett

Publikus

- Konstrktor
- Destruktor
- move – adott paraméterek alapján lép a gyalog és ellenőrzi, hogy ezt megteheti-e

Knight

Felelőssége

Egy speciális figura, ló.

Ősosztályok

Piece

Attribútumok

Privát

Védett

Publikus

Metódusok

Privát

Védett

Publikus

- Konstrktor
- Destruktor
- move – adott paraméterek alapján lép a gyalog és ellenőrzi, hogy ezt megteheti-e

Bishop

Felelőssége

Egy speciális figura, futó.

Ősosztályok

Piece

Attribútumok

Privát

Védett

Publikus

Metódusok

Privát

Védett

Publikus

- Konstruktor
- Destruktor
- move – adott paraméterek alapján lép a gyalog és ellenőrzi, hogy ezt megteheti-e

Queen

Felelőssége

Egy speciális figura, vezér.

Ősosztályok

Piece

Attribútumok

Privát

Védett

Publikus

Metódusok

Privát

Védett

Publikus

- Konstruktor
- Destruktor
- move – adott paraméterek alapján lép a gyalog és ellenőrzi, hogy ezt megteheti-e

King

Felelőssége

Egy speciális figura, király.

Ősosztályok

Piece

Attribútumok

Privát

Védett

Publikus

Metódusok

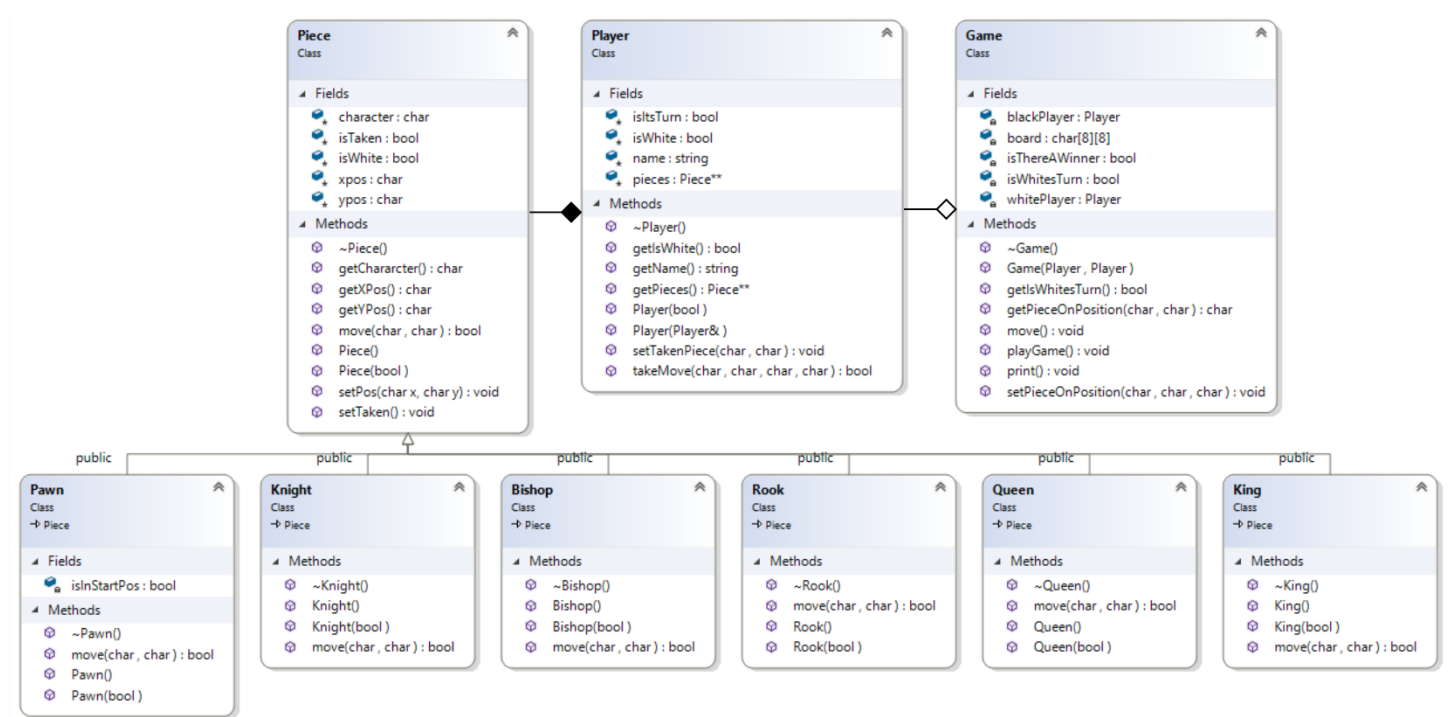
Privát

Védett

Publikus

- Konstruktor
- Destruktor
- move – adott paraméterek alapján lép a gyalog és ellenőrzi, hogy ezt megteheti-e

UML osztálydiagramm



Ábra 1 - UML diagram

Összegzés

Mit sikerült és mit nem sikerült megvalósítani a specifikációból?

Sikerült egy objektum orientált irányelvekkel működő sakktáblát és rajta mozgatható figurákat létrehoznom, úgy, hogy a figuráknak van „gazdája”, vagyis létre kell hoznunk játékosokat a játék működéséhez. Nem sikerült tökéletesre befejeznem a lépések helyességét figyelő függvényeket, mesterséges intelligenciát és a játék végén az eredmények kiíratását, azonban ezeket a továbbiakban be fogom fejezni.

A házfeladat készítésekor megismertem pontosabban a heterogén kollekció, kompozíció és aggregáció működését.

Mit tanultál a megvalósítás során?

A játék sokkal komplexebb, mint elsőre gondoltam, még úgy is, hogy előtte papíron összeírtam milyen elemek lesznek szükségesek. A lépések vizsgálata önmagában derék feladat, és azok után a még a speciális „lépések” nincsenek megemlítve se.

Továbbfejlesztési lehetőségek

Jelen állapotában még nem tökéletes játékként funkcionál, ezt szeretném kijavítani, hogy teljesértékű legyen. Valamint egy „minimax” elven működő IA-t is szeretnék bevezetni a programba, hogy egyedül is lehessen játszani.

Képernyőképek a futó alkalmazásról

```

Please choose a nickname (max 10 characters): Emma
Please choose a nickname (max 10 characters): Béla
-----
 8 | ... | n | ... | q | ... | b | ... | r |
   | .r. |  | .b. | q | .k. | b | .n. | r |
   | ... |  | ... |  | ... |  | ... |  |
   | --- | --- | --- | --- | --- | --- | --- | --- |
 7 | p | .p. | p | .p. | p | .p. | p | .p. |
   | ... |  | ... |  | ... |  | ... |  |
   | --- | --- | --- | --- | --- | --- | --- | --- |
 6 | ... |  | ... |  | ... |  | ... |  |
   | . . |  | . . |  | . . |  | . . |  |
   | ... |  | ... |  | ... |  | ... |  |
   | --- | --- | --- | --- | --- | --- | --- | --- |
 5 |  | ... |  | ... |  | ... |  | ... |
   | . . |  | . . |  | . . |  | . . |  |
   | ... |  | ... |  | ... |  | ... |  |
   | --- | --- | --- | --- | --- | --- | --- | --- |
 4 | ... |  | ... |  | ... |  | ... |  |
   | . . |  | . . |  | . . |  | . . |  |
   | ... |  | ... |  | ... |  | ... |  |
   | --- | --- | --- | --- | --- | --- | --- | --- |
 3 |  | ... |  | ... |  | ... |  | ... |
   | . . |  | . . |  | . . |  | . . |  |
   | ... |  | ... |  | ... |  | ... |  |
   | --- | --- | --- | --- | --- | --- | --- | --- |
 2 | .P. | P | .P. | P | .P. | P | .P. | P |
   | ... |  | ... |  | ... |  | ... |  |
   | --- | --- | --- | --- | --- | --- | --- | --- |
 1 | R | .N. | B | .Q. | K | .B. | N | .R. |
   | ... |  | ... |  | ... |  | ... |  |
   | --- | --- | --- | --- | --- | --- | --- | --- |
   | A | B | C | D | E | F | G | H |
Commands: (M)ove      (Q)uit
Type only one character here: _

```

```
Type only one character here: m
White's turn, alias Emma
Move from: d2
Move with P to: d4
```

8	.r.	n	.b.	q	.k.	b	.n.	r
7	p	.p.	p	.p.	p	.p.	p	.p.
6
5
4	P
3
2	.P.	P	.P.	. .	.P.	P	.P.	P
1	R	.N.	B	.Q.	K	.B.	N	.R.
	A	B	C	D	E	F	G	H

```
Commands: (M)ove      (Q)uit
Type only one character here:
```

4	P
3
2	.P.	P	.P.	. .	.P.	P	.P.
1	R	.N.	B	.Q.	K	.B.	N
	A	B	C	D	E	F	G

```
Commands: (M)ove      (Q)uit
Type only one character here: f7
Just one character!
Commands: (M)ove      (Q)uit
Type only one character here: e6
Just one character!
Commands: (M)ove      (Q)uit
Type only one character here: m
Black's turn, alias Béla
Move from: f7
Move with p to: d6
Can't move that.
Commands: (M)ove      (Q)uit
Type only one character here:
```

```
Commands: (M)ove      (Q)uit
Type only one character here: q
Bye!
```