Учебна 2012/2013

Курсов проект ООП2

Изготвил :

Щилиян Узунов

КСТ III Курс проверен:

FN : 106013 /..................../

**Информация за проекта**

Проекта представлява реализация на класическата игра шах. Играта е направена за двама души, без опция за игра срещу компютър. Целта на играта е да се матира противниковия цар. На дъската освен стандртните икони на фигурите са показани и черни и бели точки, които означават съответно – полето е атакувано от черния / белия играч.

Повече информация за правилата на играта можете да намерите на следния интернет адрес : <http://en.wikipedia.org/wiki/Rules_of_chess>

Приложението се стартира от файилa Chess.jar. За успешно стартиране на приложението е нужно да има инсталиран Java Runtime Environment.

**Ръководство на потребителя**

След стартиране на програмата се отваря прозорецът показан по-долу.

Правилата на играта са стандартни ,с реализирани всички възможни ходове на фигурите, както и специалните ходове като рокада, n passant, пешка 2 напред, пешка на 8 ранк, шах и т.н.

За извършване на ход кликнете веднъж върху фигурата която желаете да преместите, и веднъж върху полето върху което желаете да я преместите. При възможен ход на фигурата, около нея се появява червено квадратче. При неявяване на червено квадратче, това значи че тази фигура неможе да бъде преместена.

Освен местене на фигури, програмата предлага на потребителя и възможноста да :

-Започва нова игра.

-Запазва съществуващата игра.

-Зарежда запазеня игра.

-Напуска приложението.

За започване на нова игра кликнете върху меню “Game” и след това върху бутон „New Game”

За запазване на съществуваща игра кликнете върху меню “Game” и след това върху бутон “Save Game”

За зареждане на запазена игра кликнете върху меню “Game” и след тоа върху бутон „Load game”

За напускане на приложението (изход от програмата) кликнете върху меню “Game” и след това върху бутон „Quit”.

****

**Техническа реализация**

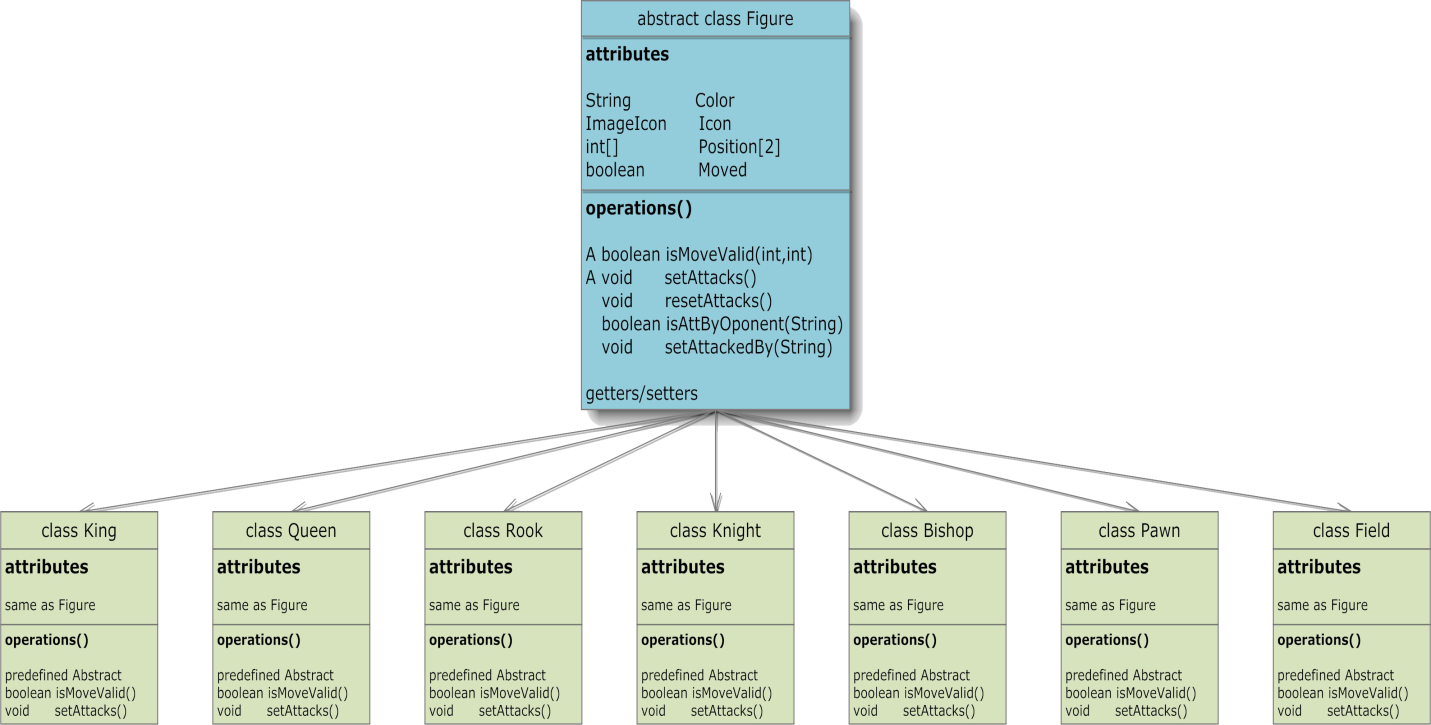
За създаването на проекта използвах програмния език Java. Приложението се стартира от файилa Chess.jar. За успешно стартиране на приложението е нужно да има инсталиран Java Runtime Environment.

Организацията на класовете се съдържа във 3 пакета.

* Figures
* Chess
* Graphics

Йерархия на класовете

в пакет Figures



**abstract Class Figure**

Главният клас в пакета, родител на останалите 7.

**Атрибути :**

**String Color** – Информация за цвета на фигурата.

**ImageIcon Icon** – Съдържа пътят към .png изображението на фигурата.

**Int[] Position[2]** – Съдържа информация за позицията наф игуратa.

**isAttByWhite**  - Информация дали фигурата е атакувана от белия играч

**isAttByBlack**  - Информация дали фигурата е атакувана от черния играч

**Boolean Moved** – Информация дали фигурата е мърдана от началото на играта

**Методи :**

**Abstract boolean isMoveValid(int,int)** – Методът проверява дали извършеният ход от фигурата е валиден. Тъй като всички фигури имат различни ходове, методът е абстрактен. Приема като параметри две цели числа (int,int) – позицията на която ще отиде фигурата.

**Abstract void setAttacks()** – Методът маркира полетата които фигурата заплашва (съответно полето е марирано с черно, ако фигурата е черна, и със бяло, ако фигурата е бяла).

**boolean isAttByOponent(String Color)** – Проверява дали фигурата е атакувана от противника. Приема като параметър даден цвят. Ако параметърът съвпада с цвета на фигурата – връща false. В противен случаи – true.

**void resetAttacks()** – Задава на полетата isAttByWhite и isAttByBlack стойност false

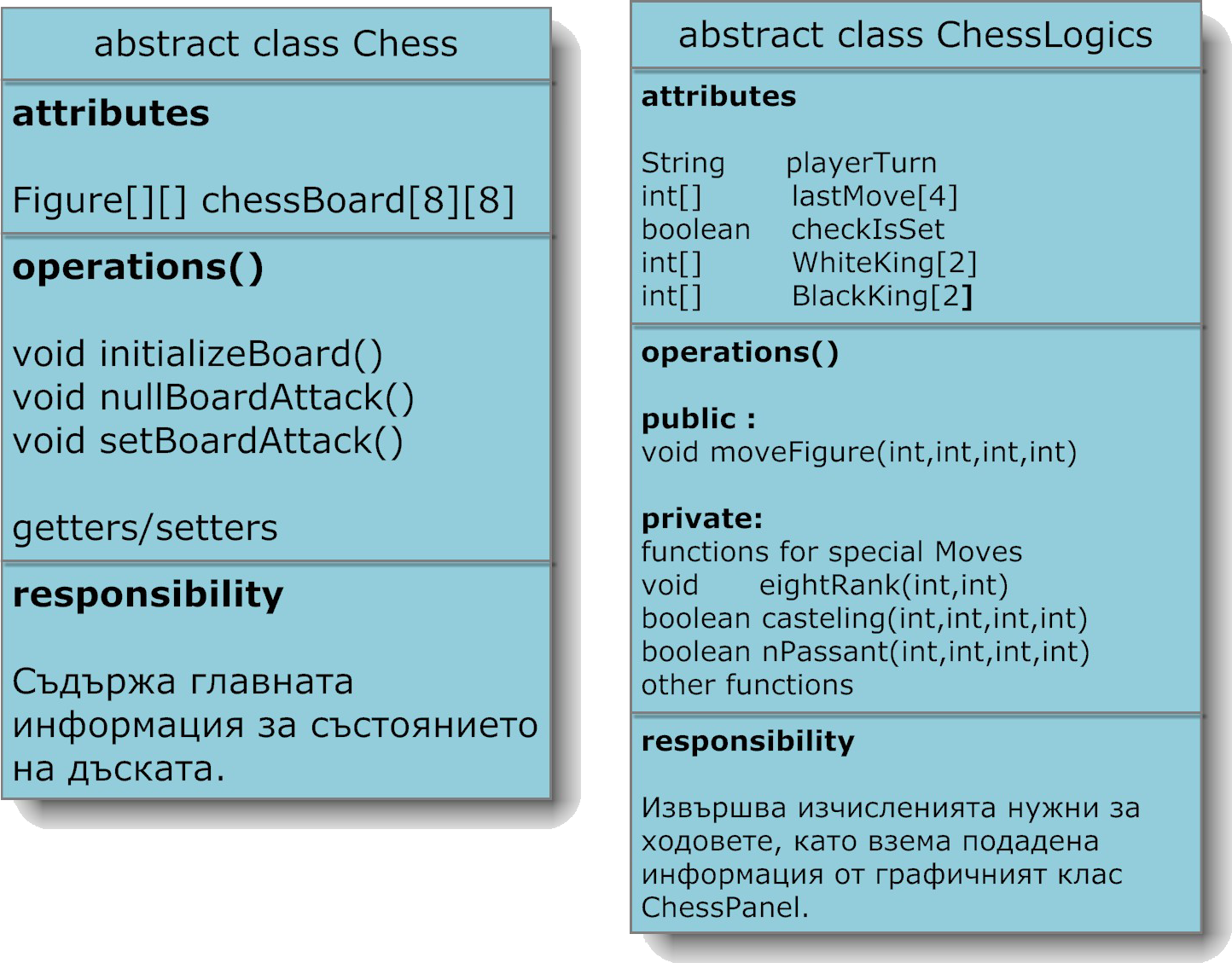
**Setters & Getters -** За промяна и извличане на съдържанието на полетата.

Класовете Queen, King, Rook, Bishop, Knight, Pawn наследяват класа Figure. Всеки от тях предефинира двата абстрактни класа isMoveValid и setAttacks спрямо начинът по който фигурата се движи.

Освен тези 6 класа за фигурите, има и седми клас – за поле. Class Field служи за да се използват неговите атрибути isAttByBlack и isAttByWhite при проверка за някои от специалните ходове (примерно Рокада, където трябва да се провери дали празните полета между царя и топа не са заплашени от противника).

Класът няма стойност за цвят и икона.

Пакет Chess



Пакетът съдържа два класа – class Chess & class ChessLogics.

**Class Chess :**

Класът съдържа главната информация за състоянието на дъската.

**Атрибути**

**Figure[][] chessBoard[8][8] –** Двумерен масив от Фигури (от пакетът Figures)

**Методи**

**void initializeBoard() –** Методът се пуска при стартирането на играта, и задава на масивът chessboard стойности на съотвените фигури. (Пример : chessboard[0][0] = new Rook())

**void nullBoardAttack() –** Всички полета от масива chessboard получават стойност на полетата си isAttByBlack isAttByWhite стойност false.

**void setBoardAttack() –** Маркира полетата заплашвани от играчите. Методът достъпва всеки един елемент от масива и използва неговата функция setAttacks().

**Setters & Getters -** За промяна и извличане на съдържанието на chessBoard.

**Class ChessLogics:**

Съдържа функции описващи правилата на играта. Повечето от тях проверяват дали извършеният ход е валиден.

**Атрибути**

**boolean checkIsSet –** Информация дали има поставен Шах.

**String playerTurn –** Информация кой играч е на ход.

**int LastMove[4] –** Масив който съдържа последният извършен ход.

int WhiteKing[2] - Масив който съдържа позицията на белия цар.

int BlackKing[2] - Масив който съдържа позицията на черния цар.

**Методи**

**void moveFigure (int,int,int,int)** – Основната функция в класа. Служи за преместване на фигурата от една позиция на друга. Използват се множество проверки дали за валидност на хода, (по-долу описани методи).

**boolean isPathClear(int,int,int,int)** – Използва се от moveFigure за проверка дали пътят от една позиция до друга е свободен.

**boolean checkTurn(int,int) –** Използва се от moveFigure за проверка дали фигурата която се опитва играчът да премести е на ход.

**boolean isFigureFriendly(int,int,int,int**) – Използва се от moveFigure за проверка дали фигурата която се опитва да премести играча не отива върхуя приятелска фигура (нарушение на правилата).

**boolean kingChecked(int,int) -** Проверява след преместването на фигура дали царят е заплашен. Ако е заплашен се извежда съобщение „Check!”.

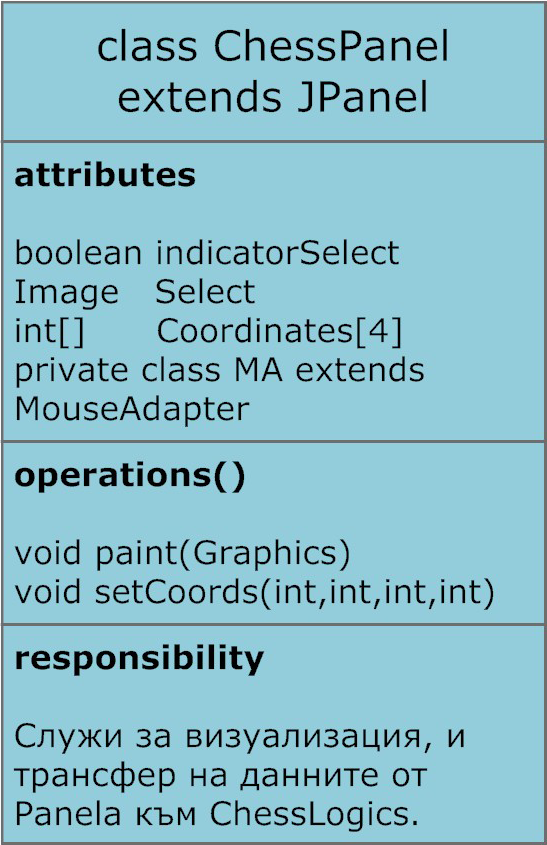
Методи за специалните ходове :

**boolean casteling(int,int,int,int)** – Проверява дали е възможно да бъде извършена рокада (дали топът е местен, царят е местен, заплашван, или някое от полетата между царя и топа е заплашвано). Ако е възможно – извършва хода.

**boolean nPassant(int,int,int,int) –** Проверява дали е възможно да бъде извършен хода Ен Пасант. Ако е възможно – извършва хода.

**void eightRank(int,int) –** Метод реализиращ правилото за пешка на 8ми ранк. Ако пешката е на 8ми ранк извежда прозорец със 4 бутона (Queen, Rook, Bishop, Knight) от които играчът може да си избере за какво да замени пешката си.

Пакет Graphics



Пакетът съдържа два класа.

Класът ChessFrame наследява JFrame (клас от Java библиотеката създаващ рамка) и използва класът ChessPanel за създаване на панел.

Класът ChessPanel наследяа JPanel (клас от Java библиотеката създаващ панел).Той служи за визуализация и трансфер на данните от панела, към функцията moveFigure(int,int,int,int) от класа ChessLogics.

Един от атрибутите на ChessPanel е вътрешния class MA който наследява класът MouseAdapter. Чрез класът MA при кликване върху панела, програмата получава координатите на мишката, от там те се обработват до подходящи числа които да бъдат предадени за достъп до масива ChessBoard (пример : x=420, y=250 предават се x=6, y=4).

Друг важен елемент от класът ChessPanel е предефинираната функция на JPanel – void paint(Graphics). С предефинирането и се казва как ще се визуализира панела. Функцията се пуска всеки път при успешен ход.

**Възможно бъдещо развитие на проекта :**

* Добавяне на опция за игра в мрежа
* Добавяне на опция за игра срещу компютър

**Използвана литература :**

* Въведение в програмирането с Java. Автори : Светлин Наков и коектив.
* <http://docs.oracle.com> документация за Java класове.
* <http://en.wikipedia.org/wiki/Rules_of_chess> правила на шаха.