



РЕПУБЛИКА МАКЕДОНИЈА
УНИВЕРЗИТЕТ “Св. Кирил и Методиј,, Скопје
Факултет за информатички науки и компјутерско
инженерство



Документација
Развој на апликација – Шах
предмет: Визуелно Програмирање
github link : <https://github.com/MihailVladimirVP/Chess>

Михаил Петков

111161

Владимир Николовски

111150

проф. Д-р Ѓорѓи Маџаров

асс. м-р Томче Делев

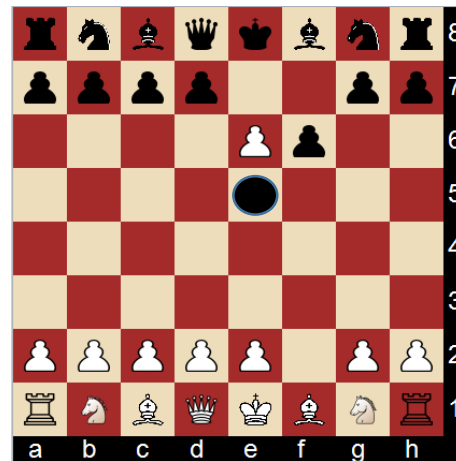
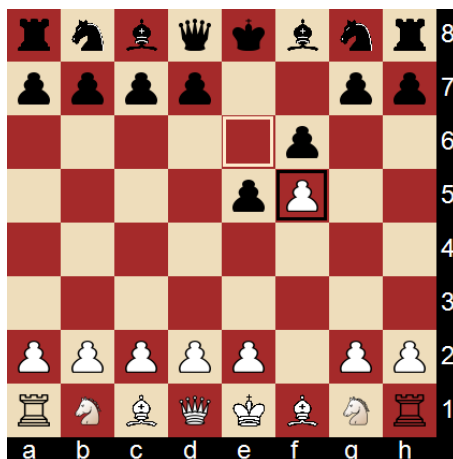
Скопје
Мај, 2013

Како тема на проектот ние одбравме развој на добро познатата игра шах. Во истата постојат 6 различни фигури, распределени на два играчи (црн и бел). Секоја од фигурите има свои сопствени валидни движења, фигурите се борат и се одземаат едни против други, а крајната цел на играта е да се освои кралот на противничкиот тим со што се доаѓа до состојба шах мат. Валидни движења на секоја од фигурите се:

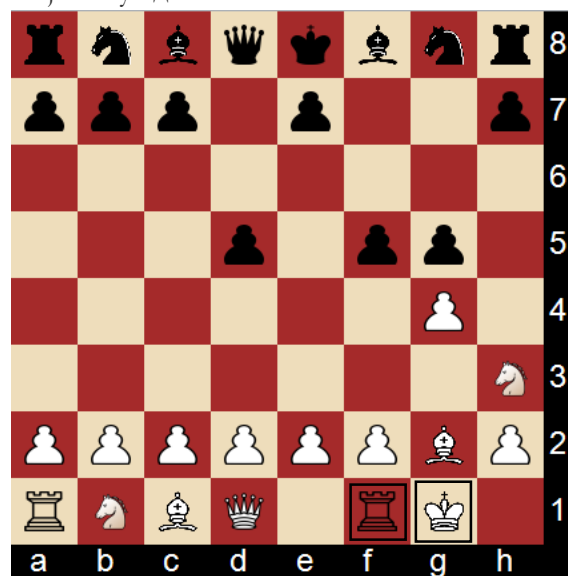
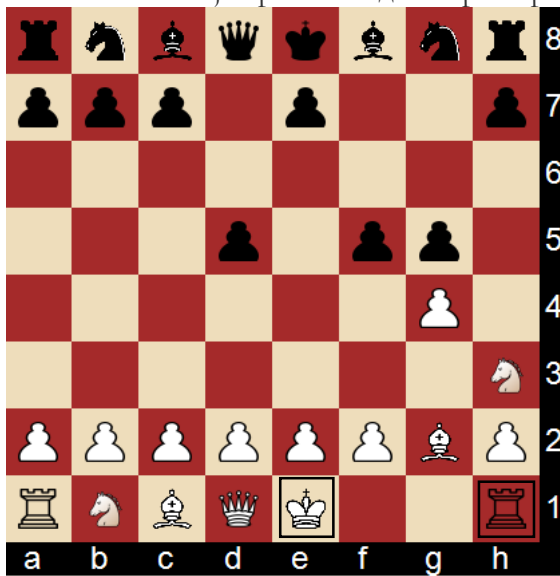
- Пионите се движат единствено во спротивната насока од својата, и прават потези од едно поле напред, доколку полето пред нив е слободно, односно косо напред (лево и десно) доколку полето пред нив е зафатено од противничка фигура (со што противничката фигура се одзема). Пионите во својата почетна состојба, може да се придвижат две полиња напред, доколку полињата не се зафатени.
- Топот се движи само со вертикална и хоризонтална насока до првата фигура којашто му го попречува патот.
- Коњот прави потег Г, во било која од насоките на таблата, без да се земе во обзир дали има фигури на патот.
- Ловецот се движи во коса насока до првата фигура којашто му попречува пат.
- Кралицата ги наследува сите валидни потези на топот и ловецот, со што се движи во сите можни насоки, се додека фигури не и го попречат патот.
- Кралот се движи во сите можни насоки, но само по едно поле.

Специјални движења на фигурите се:

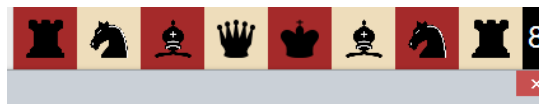
- En passant – движење на пионот во насока косо-напред (лево или десно) доколку се исполнети следните услови
 - Пионот којшто го прави ова движење е на 5ти ранг од својата гледна точка
 - Претходниот потег, противникот го направил првото движење напред со својот пион, правејќи дупли потег напред. Неговиот пион се наоѓа веднаш до пионот на играчот којшто го прави en passant движењето.
 - Со ова се исполнети условите за валиден потег en passant, со којшто се зема фигурата на противникот која се наоѓа веднаш до фигурата на играчот кој го прави движењето.



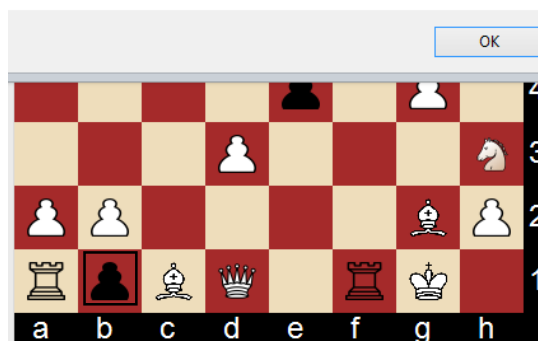
- Рокада (castling) во којашто кралот оди две места накај топот со којшто прави рокада, а топот оди од другата страна на кралот, доколку се исполнети следните услови:
 - Кралот на играчот досега не направил потег
 - Топот кој ќе се замени со кралот досега не направил потег
 - Меѓу кралот и топот нема фигури
 - Секој играч смее да направи рокада најмногу еднаш



- Унапредување пион (Pawn Promotion) е специјално движење на пион, во којшто пионот може да се замени со друга фигура по избор (различна од крал) кога тој пион ќе стигне на другиот крај од таблата.



Pawn Promotion! Одберете друга фигура којашто сакате да ја замените наместо пионот



Опис на решението:

За имплементација на шах, се чуваат следните податоци, во следен формат:

- Фигурите во шахот се чуваат во `Ostanati` како низа од 64 полиња енумерирани со вредности `King`, `Queen`, `Rook`, `Bishop`, `Knight` и `Pawn` (односно Крал, Кралица, Топ, Ловец и Пион) за секоја од различните фигури, односно `None` (ништо) доколку на тоа поле нема застаната фигура. Секоја фигура има булова променлива `White` за бели (`true`), односно црни (`false`) фигури.
- Секоја фигура, следејќи се по 64те индекси на `Ostanati` има сопствена слика сместена во соодветниот индекс во низата `figures`.
- Низа од 4 булови `PlayerTurn`: 1 за прв клик на бел играч, 2 за место на поместување на фигурата на бел играч, 3 и 4 исто, соодветно за црн играч.
- `FinishedPlayer`, доколку потегот на играчот е завршен.
- `BlackTaken`, `WhiteTaken`, листи на одземени фигури, коишто потоа се сместуваат во соодветни `listBoxes` со формат опишан од класата `TakenFigures`. Истата опишува која фигура, во колкав број е земена.
- Низа од булови `Enpassant`, за водење евиденција дали тоа специјално движење е дозволено.
- Низа од булови `Castling`(0, 1, 2 и 3) резервирани за црните фигури (`Validate.Castling`, крал, лев топ, десен топ), односно `Castling`(4, 5, 6 и 7) за белите фигури. Преку повик на `Castling` класата, се води евиденција дали рокадата е валидна преку доделување на `Castling[0]`, односно `Castling[4]`. Оваа класа како параметри прима дали се работи за црн или бел, лев или десен топ.
- `TakenPlaces matrica`, од класата `TakenPlaces` којашто чува дводимензионална матрица [8][8], дали местото е зафатено од фигура, и доколку да, од кој играч. Истата е дополнителна структура којашто обезбедува полесен пристап од `Ostanati[64]`, за пресметување на модуларната аритметика на класата `ValidMoves` којашто укажува кои потези на фигурите се валидни во даден момент.
- Класата `ValidMoves` како параметар прима фигура и нејзиниот индекс, и за истата, во зависност од типот и боја, распределноста на останатите фигури, буловите `Enpassant` и `Castling`, користејќи модулална аритметика одредува на кои полиња смее да се придвижи.
- `SahMat` е класа којашто после потегот на спротивниот играч, го предупредува другиот доколку неговиот крал е во шах позиција, односно е нападнат од некоја фигура од спротивниот тим. Во исто време, класата содржи функција којашто му ги оневозможува потезите на кралот кон местата во коишто повторно би бил под напад. Имплементирана е и шах мат проверка, т.е. ако некој од играчите е во шах мат играта ќе биде завршена.

- Draw Valid ги исцртува со црни правоаголници кликнатите фигури, и ги исцртува валидните потези на истата фигура.
- HighScores е DataGridView во која се чуваат колони од имињата на играчите и нивното време на играње. Времето на играње се определува со имплементирање на два тајмери, за секој играч посебно, кои се вклучуваат со клик на копчето “Старт Игра”
- Поради природата на шах играта, Undo копче не беше имплементирано, иако истото би можело да се имплементира со користење на стек структурата.

Опис на функцијата figureHandlers:void на Form1 (шах таблата)

Функцијата, објектот којшто е кликнат (picturebox) го сместува во променлива. Се повикува класата ValidMoves да пресмета валидни потези за фигурата според индексот којшто е кликнат. Според низата од булови PlayerTurn се одредува кој потег треба да биде проследен. Доколку се работи за одбирање на бела фигура, тоа е Player[0] (бел прв потег):

- Фигурата ќе биде сместена во помошен PictureBox
- Индексот којшто е кликнат ќе биде сместен во integer
- Доколку е кликнато надвор од шаховската табла или на picturebox на којшто нема фигура, или фигурата е од спротивниот тим, тогаш се рестартира истиот потег

Ако се работи за поместување на белата фигура на друго поле, тоа е Player[1] (бел втор потег):

- Ако е кликната невалидна позиција, се рестартира потегот назад кон одбирање на фигура којашто треба да биде поместена.
- Доколку се кликнати индекси коишто овозможуваат enpassant на другиот играч, се променуваат буловите на true за тие индекси, односно false за сите останати
- Доколку Castling буловите се true и доколку кликнатите индекси соодветствуваат на Castling движење, преку Castling.ValidirajCastle() се проверува дали фигури помеѓу кликнатиот топ и коњ постојат, и доколку да, движењето се извршува, се променуваат типовите на индексите на фигурите, се апдејтираат зафатените места и се променува лабелот којшто укажува на тоа кој играч е следен на потег.
- Се врши проверка дали е селектиран крал, и доколку е, дали на вториот клик играчот се обидува да го помести кон поле кон којшто тој би бил во позиција шах. Доколку тоа е случајот, истиот потег се оневозможува.
- Доколку е валиден потегот, смени ги сликите и индексите на местата на фигурите.
- Провери дали фигурата се наоѓа во листата на земени, и ако не се наоѓа додади ја. Ако фигурата е пронајдена, зголеми го бројот на земени фигури од таков тип и апдејтирај ја листата.

- Ако играчот се` уште не е завршен, се проверува дали е поместена топ или крал, и доколку да, се оневозможува Castling движење на тој специфичен топ, или воопшто на тој тим, доколку се работи за поместен крал.
- Се проверува дали се работи за enpassante движење, и доколку истото е валидно и извршено, се менуваат индексите на фигурите, како и апдејтираат зафатените места.
- Се врши проверка дали кралот на спротивниот тим е се` уште во игра. Доколку не е пронајден кралот, играта завршува. Доколку е се` уште во игра, се проверува дали е во шах положба, и спротивниот играч се предупредува.
- Доколку пионот дошол до крајната позиција, се прави претходно објаснетиот потег Pawn Promotion.
- После секој потег се повикува функцијата this.DrawIt за постојано гледање на валидни потези на селектирани фигури.
- На ист начин функционираат и PlayerTurn[2] (црн прв потег) и PlayerTurn[3] (црн втор потег)

