**Шахмат**

Курсов проект по ООП-практикум,

ФМИ, СУ, летен семестър 2020/2021

### Съдържание

### Описание на проекта

Проектът реализира конзолно приложение, симулиращо игра на шах между двама играчи. Дава възможност за придвижване на фигурите, взимане на противникови фигури, обявяване на шах и разпознаване на мат.

Целта на проекта е успешна симулация на разиграването на партия шах.

### Как се играе

На всеки вход се изпечатва игровото поле. Белите фигури са отбелязани с главни латински букви, а черните – с малки. Играчите се редуват да извършват ходове, като всеки ход се отбелязва като кода на фигурата с малка буква (независимо от цвета й), полето на което се намира (символ-цифра нотация) и полето, на което искаме да я преместим (пример: ***r a3 a5***). Извеждат се различни съобщения, ако ходът е невалиден, успешен или е взета противникова фигура. Играта завършва с разпознат мат на единия играч или с команда *exit*.

### Архитектура

Архитектурата на проекта разпознава различните видове фигури – реализирани като отделни наследяващи класове на основен клас *Piece*. Също така се включва отделен клас, представящ дъската, върху която се осъществява игра, както и един отделен клас, поддържащ логиката на играта.

(диаграма)

**Position**

Представя една позиция на дъската. Състои се от координати, предтавени като символ (a-h) и цифра (1-8).

int toInt() const - представя еднозначно позицията като цяло число.

bool inLine(const Position&) const - проверява дали позицията се намира в същата колона или ред от текущата.

bool inDiagonal(const Position&) const - проверява дали позицията се намира в същия диагонал от текущата.

**Piece**

Базов клас, представящ една фигура от играта на шах. Съдържа информация за нейната позиция, цвят и символно представяне.

virtual void move(const Position&) - проверява дали фигурата може да се премести на съответното поле.

TBD

**King: Piece**

Клас, представляващ фигурата цар в една игра на шах. Наследява базовия клас *Piece*. При взимането му от противникова фигура, играта свършва с мат, победител е противника. Ако е заплашен от взимане, това се нарича “шах” и трябва задължително да бъде или преместен, или заплахата да бъде премахната (противниковата фигура да бъде взета или да се постави друга фигура между царя и противниковата фигура). Символното му представяне е ‘k’.

void move(const Position&) - проверява дали може да се премести на дадена позиция. Цярят може да се премества с една позиция в права линия (същия ред и колона) или по диагонал.

### Обобщение на реализацията

Реализацията покрива по-голямата част от базовите изисквания на проекта – дава възможност за извършване на ход – местене на фигура и взимане на противникова фигура, разпозвнава сигуациите шах и мат.

### Възможности за разширение

Една възможност за допълнителна функционалност е реализацията на механизъм за запазване на информация във файлове, както и за възпроизвеждане на историята на играта.

### Заключение

Проектът е реалиализиран успешно, спайзвайки основните ООП принципи.

### Използвана литература и външни препратки

[https://blog.chesshouse.com/how-to-read-and-write-algebraic-chess-notation/#:~:text=Chess%20Notation%20Explained](https://blog.chesshouse.com/how-to-read-and-write-algebraic-chess-notation/" \l ":~:text=Chess Notation Explained)