Санкт-Петербургский политехнический университет Институт компьютерных наук и технологий

**Кафедра «Компьютерные системы и программные технологии»**

**КУ Р С О В ОЙ П Р О Е К Т**

**Разработка игры "Английские шашки"**

по дисциплине «Технологии программирования»

Выполнил студент

гр. 3530901/10001 Шмелёв К.В.

Преподаватель Алексюк А.О.

20 мая 2022 г.

Санкт-Петербург

2022

Санкт-Петербургский политехнический университет

**ЗАДАНИЕ**

**НА ВЫПОЛНЕНИЕ КУРСОВОГО ПРОЕКТА**

студенту группы 3530901/10001 Шмелёву Кириллу Валентиновичу

1. Тема проекта: создание игры «Английские шашки» с графическим интерфейсом.
2. Срок сдачи законченного проекта: 18 июня
3. Исходные данные к проекту: требования к реализовываемому проекту
4. Содержание пояснительной записки: введение с правилами игры, основная часть (технологии JavaFX, их применение в приложении), описание тестов, заключение, список используемых источников.

Дата получения задания: «2» апреля 2022 г.

Руководитель Алексюк А.О.

Задание принял к исполнению Шмелёв К.В. 2 апреля 2022

**СОДЕРЖАНИЕ**

[Введение 4](#_TOC_250004)

[Описание решения 5](#_TOC_250003)

[Тестирование программы](#_TOC_250002) 7

[Заключение](#_TOC_250001) 8

[Список используемых источников](#_TOC_250000) 9

Введение

Цель работы: создать и протестировать игру Английские шашки на двух игроков за одним компьютером

Правила игры:

Английские шашки (в британском варианте английского языка «English draughts», в США и Канаде «чекерс») — одна из разновидностей игры в шашки

Доска 8×8 клеток располагается между партнерами таким образом, чтобы слева от играющего находилось тёмное угловое поле. В начальной позиции у каждого игрока по 12 шашек, расположенных в первых трёх рядах на чёрных клетках.

Первый ход делают чёрные шашки. «Простые» шашки могут ходить по диагонали на одно поле вперёд и бить только вперёд. Дамка может ходить на одно поле по диагонали вперёд или назад, при взятии ходит только через одно поле в любую сторону, а не на любое поле диагонали, как в русских или международных шашках. Взятие шашки соперника является обязательным. При нескольких вариантах взятия игрок выбирает вариант взятия по своему усмотрению, и в выбранном варианте необходимо бить все доступные для взятия шашки.

При достижении последнего (восьмого от себя) горизонтального ряда простая шашка превращается в дамку. Если простая достигла последнего ряда во время взятия, то она превращается в дамку и останавливается, даже при возможности продолжить взятие.

Описание решения

Мной была использована библиотека JavaFX для создания графического пользовательского интерфейса (GUI). Данная библиотека обладает большим числом заготовок элементов интерфейса. В частности, были использованы: текст (Text), кнопка (Button), линия (Line), а также Pane и его потомки. Взаимодействие с графическим элементом описывается в событиях для него. Все события базируются на нажатии кнопок мыши по кнопкам или игровым фигурам

Код поделен на 2 пакета и класс Main. В пакете logic находятся классы DiagonalChecker, GameStatistic, Turner, Utils, VerifierTurns. В пакете ui находятся классы CheckersBoard, Constants, GameStage, InfoCenter, Main, TileBoard, WelcomeStage.

В классе Main содержится метод main, она запускает метод main в классе Main из пакета ui.

Пакет ui. CheckersBoard содержит в себе игровые фигуры игроков и запускает событие в результате нажатия пользователя мышкой по ним.

Пакет ui. Класс Constants содержит private final static переменные, которые являются константами для всей программы для удобства их использования.

Пакет ui. Класс GameStage описывает инициализацию поля и фигур.

Пакет ui. Класс InfoCenter содержит кнопки инициализации поля и фигур, а также рестарта и отображает информацию о ходе и победителе

Пакет ui. Класс Main запускает программу

Пакет ui. Класс TileBoard содержит в себе игровое поле(клетки)

Пакет ui. Класс WelcomeStage описывает инициализацию центра информации (InfoCenter)

Пакет logic. Класс DiagonalChecker является вспомогательным классом для класса VerifierTurns. Содержит в себе методы, позволяющие проверить клетки по диагонали от шашки на тип возможного хода

Пакет logic. Класс GameStatistic содержит в себе поля и методы, касающиеся глобальных данных вроде количества фигур у игроков или объявления победителей.

Пакет logic. Класс Turner отвечает за выбор шашки или совершение хода после клика пользователя

Пакет logic. Класс Utils содержит в себе разные методы, отвечающие за разные задачи. Например, за превращение шашки в дамку

Пакет logic. Класс VerifierTurn отвечает за проверку шашки на возможность совершения желаемого хода, а также на наличие возможных ходов или взятий.

Тестирование программы

С использованием библиотеки JUnit было написано 8 тестов, используемых для проверки правильности работы приложения.

Первый тест проверяет правильность работы функций из класса Utils. Второй тест проверяет правильность изменения игровых данных в GameStatistic и правильность окончания игры. Третий тест проверяет VerifierTurns на отсутствие ошибок в определении хода и его возможности. Четвертый тест проверяет Turner на правильность выбора фигуры, которой пользователь собирается походить. Пятый тесть проверяет правильность смены активной шашки на другую. Шестой тест проверяет правильность обычного хода шашки. Седьмой тест проверяет правильность взятия. Восьмой тест проверяет правильность взятия 2 шашек сразу.

Заключение

Было создано приложение с графическим интерфейсом, для игры в «Английские шашки». Были разработаны тесты для проверки правильности кода. В ходе выполнения задания научился работать с библиотекой JavaFX и связывать логику и графику с действиями пользователя.

Исходные файлы приложения лежат в репозитории на GitHub: <https://github.com/ristavor/ProgrammingLabSummer2022Task3>

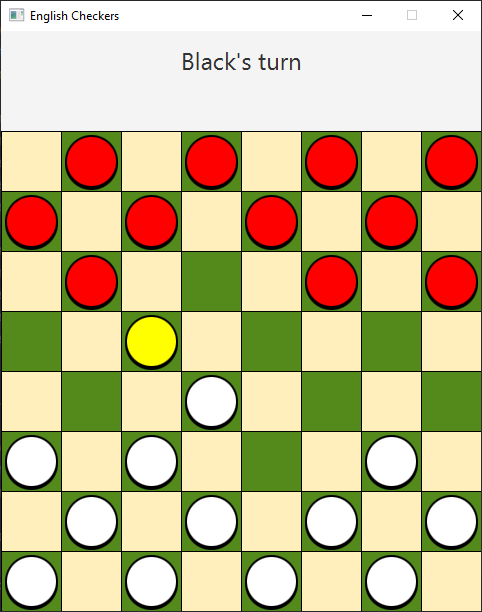


Рис 1. Скриншот приложения

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. https://metanit.com/java/tutorial/ – описание языка Java
2. https://habr.com/ru/post/474498/ - руководство по JavaFX
3. https://metanit.com/java/javafx/ – описание JavaFX
4. https://github.com/Kotlin-Polytech/FromKotlinToJava - конспект по созданию GUI от лектора