Федеральное государственное автономное
образовательное учреждение
высшего образования
«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт космических и информационных технологий
институт

Кафедра «Информатика»
кафедра

ОТЧЕТ О ПРАКТИЧЕСКОЙ РАБОТЕ

Написание unit / integration тестов и упражнение в проектировании архитектуры приложения
Тема

Преподаватель _____ К. В. Богданов
подпись, дата | инициалы, фамилия

Студент КИ19-16/1б 031939175 _____ А. Д. Непомнящий
номер группы, зачетной нижки | подпись, дата | инициалы, фамилия

Красноярск 2022

1 Цель работы

Целью является знакомство с парадигмой разработки через тестирование (TDD).

2 Задачи

Задача – «TDD шахматы» подразумевает следующие действия.

1. Написать тесты для шахматных фигур Ферзь и Конь. В тестах учесть как негативные, так и позитивные варианты. Для указанных фигур учесть все возможные игровые ситуации. Отдельно протестировать ситуацию взятия одной фигурой другой.

2. Реализовать минимально функциональные классы выбранных фигур и продемонстрировать выполнение тестов.

3. Оформить отчет, в котором описать логику построения тестов и итоговые результаты выполнения. Код приложить отдельно

3 Ход работы

3.1 Разработка тестов и программы

Для обеих фигур тестируются следующие сценарии.

1. Ход на пустую клетку, достигаемую за одной ход фигурой (на пустой доске).

2. Ход на пустую клетку, не достигаемую за одной ход фигурой (на пустой доске).

3. Ход на клетку, занятую союзной фигурой.

4. Ход на клетку, занятую фигурой соперника.

5. Ход на клетку, занятую вражеским королем.

Ход на клетки за пределами поля не тестируется, так как, учитывая особенности архитектуры программы, такое действие возможно только при очевидно неправильно написанном вызове существующего кода.

Для ферзя сценарий 1 разбит на три – ход по диагонали, вертикали и горизонтали. Так же для ферзя добавлен сценарий хода на клетку, путь к которой заблокирован какой-либо фигурой.

Отдельно тестируются классы шахматного поля и клетки шахматного поля. В частности – тестируются возможность получить клетку за пределами поля и клетку в поле, корректность проверки наличия блокирующей фигуры на траектории хода.

При создании тестовых сценариев учитывалась архитектура приложения, в частности – предполагаемые действия по гипотетической дальнейшей разработке программы и игровые особенности выбранных фигур.

Исходный код программы и тестов прилагается отдельно от настоящего отчета и может быть найден на GitHub в репозитории https://github.com/durakin/QALabs .

3.2 Результаты

На рисунке 1 представлен отчет о прохождении тестов. На рисунке 2 представлен отчет о покрытии кода тестами, сгенерированный Rider.

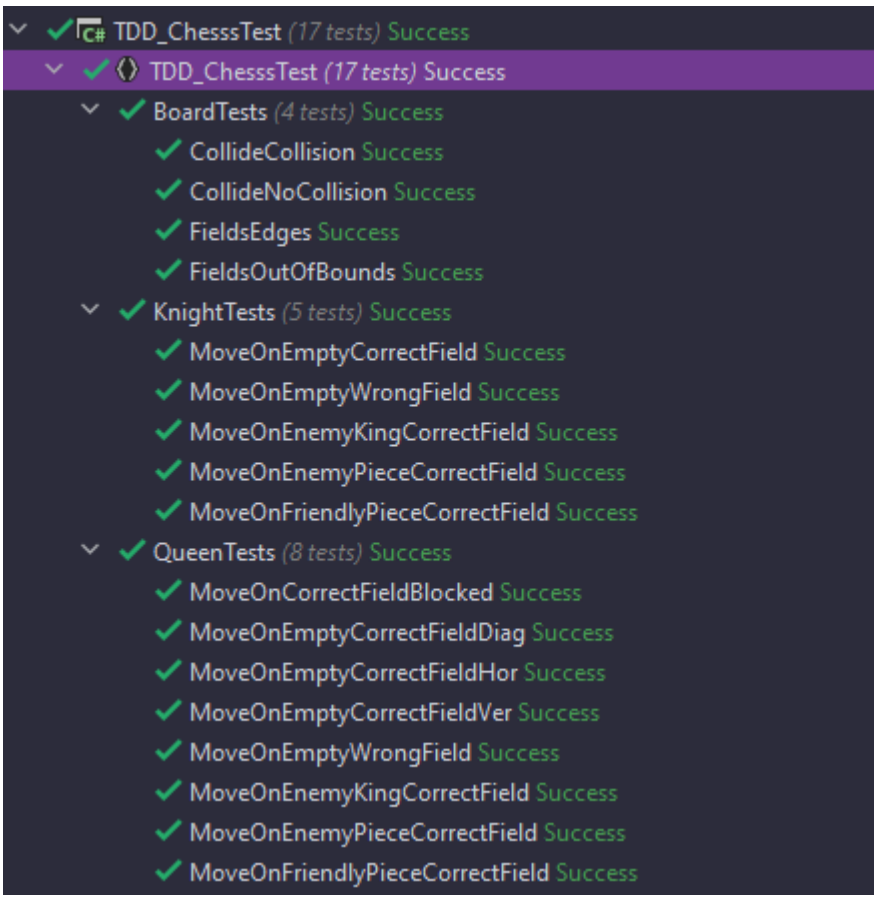

Рисунок 1 – Отчет о прохождении тестов

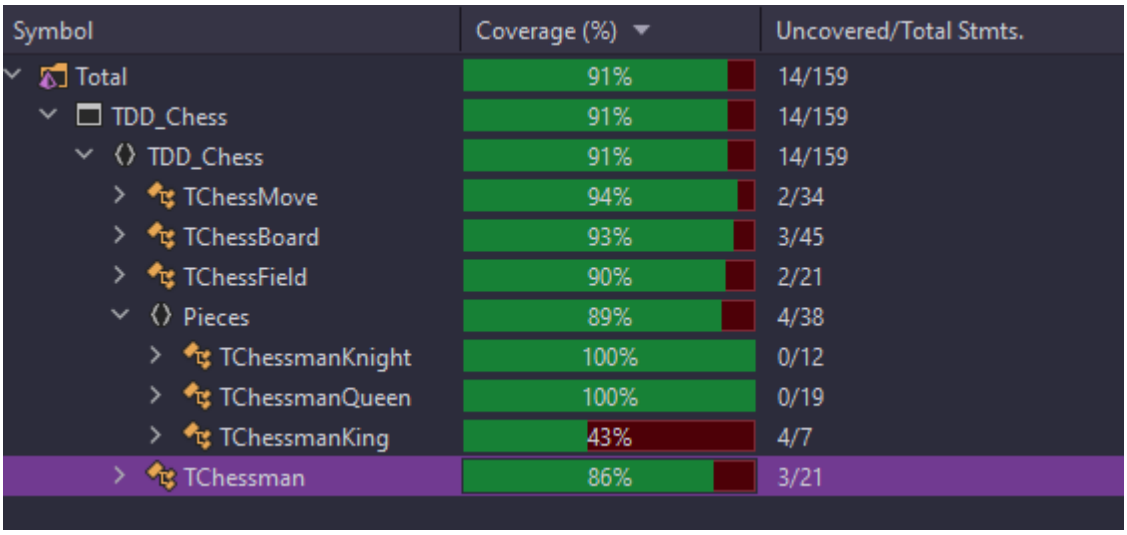

Рисунок 2 – Отчет о покрытии кода

Таким образом, в результате исходный код оказался покрыт написанными тестами почти полностью. При детальном рассмотрении отчета (приложен вместе с исходным кодом) видно, что непокрытые фрагменты исходного кода по

большей части отвечают за отслеживание обращений из гипотетического продолжения кода программы.

4 Выводы

В ходе работы было осуществлено знакомство с парадигмой TDD. На основе нее была разработана простая программа.