**ДОДАТОК 3**

Опис програми

НТУУ «КПІ ім. Ігоря Сікорського» ТЕФ АПЕПС ТІ-01

Листів 2

Київ – 2021

В результаті виконання поставленого завдання для курсової роботи за курсом “Основи програмування – 3” створено антагоністичну консольну гру “Шахи”.

В першу чергу призначена ця програма для того, щоб розважити користувачів, проте може бути використана і в навчальних цілях: новачки можуть вивчити головні правила та навчитись грі, досвідчені гравці матимуть можливість повправлятись та потренуватись.

Логіка гри реалізована через великий набір функції, що безпосередньо не взаємодіють з користувачем. В основі алгоритму програми – генерації можливих ходів на основі тільки зробленого. Як і в реальних шахах, гра має початок – стартовий стан дошки та положення всіх фігур на ній, за це у програмі відповідає конструктор головного класу – Chess(). Він складається з чотирьох функцій: init\_move\_table(), unpack\_position(const char \*position), position\_handler() та update\_moves(). Функція init\_move\_table() відповідає за ініціалізацію масиву, який зберігає доступні для кожної фігури ходи; unpack\_position() – встановлює початкові стани та складається з трьох інших функцій:

* void unpack\_desk(const char \*position) – функція для отримання символів фігур та встановлення станів кожної клітинки;
* void unpack\_move(const char \*position) – відповідає за встановлення першого ходу (e8 e8 k);
* void unpack\_castling(const char \*position)– встановлює початкові стани фігур, що беруть участь в рокірвці.

Однією з найголовніших у програмі однозначно можна назвати функцію position\_handler(), яка є “збірною” функцією та відповідає за обробку дошки: при виклику індикатор шаху встановлюється як нуль – тобто його немає. Спочатку очищується дошка з ходами, потім особливі стани клітинок та фігур, після чого відбувається “встановлення”, реалізація обраного ходу (release\_current\_move()) та обробка рокіровок, лінії союзних фігур та ворожих за допомогою функції desk\_handler(cell\_handler\_t handl). У кожній функції-обробнику (анг. handler – обробник): ally\_figure\_handler(int x, int y) або enemy\_figure\_handler(int x, int y) відбувається “відбір” потрібних фігур та передача в окремі обробники, яких загалом дванадцять, тобто для кожної фігури, союзної та ворожої. Ці функції в свою чергу передають координати клітинок, які потрібно обробити в інші функції-обробники: X\_handler(int x, int y, int vh, int d, line\_handler\_t handl), наприклад, універсальний обробник, для перевірки координат по діагоналі та горизонталі, як для союзних, так і ворожих фігур. Це здійснюється за допомогою інших функцій: ally\_line\_handler (int x, int y, int dx, int dy) та enemy\_line\_handler (int x, int y, int dx, int dy), що відповідають за перевірку та додавання правильних ходів для союзних фігур та генерації ходів для ворожих.

Король є найважливішою фігурою, й тому на його перевірку відведено багато функцій: atack\_handler (int x, int y, int dx, int dy), is\_check(), is\_multicheck(), ally\_king\_handler(int x, int y), enemy\_king\_handler(int x, int y).

Як не дивно, але найбільшої уваги потребує не дуже важлива, на перший погляд, фігура – пішак. В коді програми передбачено багато функцій пов’язаних з ним, а саме: extra\_check(), pawn\_kill\_move(), enemy\_pawn\_handler(), ally\_pawn\_handler(). Функція extra\_check() відповідає за перевірку особливого ходу в шахах, відомого як “En passant” або “взяття на проході”, pawn\_kill\_move() – за визначення ходів-побиття пішаком та enemy\_pawn\_handler() й ally\_pawn\_handler() – обробники ворожого та союзного пішака відповідно.

До базисних функцій, які відповідно забезпечують основні перевірки та без яких неможлива коректна робота, належать get\_mobility(), is\_check(), is\_multicheck(), is\_correct(), is\_figure(), is\_ally(), is\_enemy(), is\_moveable(), is\_direction\_correct() та X\_handler().

Проте все описане вище недоступне оку користувача, він може побачити лише “відкриту” частину програми.

Запуск програми починається з виведення меню, де користувачу надається можливість обрати варіант розвитку подій. Для вибору опції, користувачу необхідно повністю ввести назву варіанту. В програмі також передбачена перевірка правильності користувацького вводу, отже після некоректного введення надається повторна можливість обрати варіант з меню.

Один із трьох варіантів меню – “rules”, обравши його, користувач зможе ознайомитись з основними правилами гри в шахи, правилами ходів для кожної з шести фігур, побачити вигляд стартової дошки та ознайомитись з основними особливими станами та ходами в грі.

Одразу після введення “exit” користувач зможе побачити повідомлення: “Thank you for the game! Hope to see you later.”, після чого програма завершить свою роботу, ця опція доступна і під час партії, що в шахах означає “здатися”. Вихід відбувається за допомогою стандартної бібліотечної функціії exit().

З вибором “play” розпочинається гра, користувачі зможуть побачити перед собою стартову шахову дошку, та панель з усіма варіантами ходів, що можуть здійснити фігури. За це відповідає функція Play(), в ній відбувається виклик функції output\_desk() для виведення дошки та вивід ходів та запит користувача вибрати один з них.

Вибраний варіант передається функції Turn(), де він запам’ятовується, та після цього викликається обробник позицій, відбувається обробка дошки: відбувається її аналіз та відповідно до нового стану генеруються можливі ходи. Після цього користувач зможе побачити оновлену дошку з

Звичайно ця реалізація гри в шахи не є “ідеальною” – програму можна вдосконалювати: передбачувати більше особливих випадків та ситуацій з урахуванням тонкощів гри, вдосконалювати інтерфейс, робити його зручнішим та на вигляд кращим. Проте, ця версія містить все, щоб дозволити користувачу зіграти в улюблену гру.