

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ
УКРАЇНИ
«Київський Політехнічний Інститут імені Ігоря
Сікорського»
кафедра Технічної кібернетики

Програма “Тетріс”

UA.IT.74020-01 01

*Курсова робота
дисципліна*

“Об'єктно-орієнтоване програмування - I”

Керівник:
ас. Дзінько Р. І.

Виконав:
Журба М.А.

“Захист дозволено”

гр. ІТ-74

“ ” _____ 2018р.

Підпис керівника: _____

Підпис студента: _____

Захищено з оцінкою

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»

Кафедра Технічної кібернетики
Дисципліна Об'єктно-орієнтоване програмування - 1
Курс II Група ІТ-74 Семестр III

ЗАВДАННЯ
на курсову роботу студента

Журба Микола Андрійович
(прізвище, ім'я, по батькові)

1. Тема роботи Розробка програми «Тетріс»

2. Строк здачі студентом закінченої роботи _____
3. Вихідні дані до роботи мова програмування
C++

4. Зміст розрахунково – пояснювальної записки (перелік питань, які підлягають розробці) _____
Розробка алгоритму програми
Архітектура програми

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень) _____
UML-діаграма класів

6. Дата видачі завдання 11.10.2018

КАЛЕНДАРНИЙ ПЛАН

[illegible]

Студент _____

Керівник _____

«_____» _____ 201_ г.

(прізвище, ім'я, по батькові студента)

Дзінько Р.І.

(прізвище, ім'я, по батькові викладача)

ЗМІСТ

ВСТУП.....	5
1. ОБГРУНТУВАННЯ І ВИБІР АЛГОРИТМУ.....	6
2. ПЕРЕЛІК ТА ПРИЗНАЧЕННЯ ІНСТРУМЕНТІВ	7
3. СТРУКТУРА ДАНИХ ТА РЕСУРСІВ ПРОГРАМИ	8
4. ОПИС ПРОГРАМИ	11
5. ІНСТРУКЦІЯ ПРОГРАМІСТА.....	14
6. КЕРІВНИЦТВО КОРИСТУВАЧА.....	16
ВИСНОВКИ	20
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	21

					UA.IT.74020-01 01	Лист
Вим.	Лист	№ докум.	Підпис	Дата		

ВСТУП

Тетріс — відеогра-головоломка, розроблена Олексієм Пажитновим та його колегами. Перша версія гри була представлена 6 червня 1984 року. Назву гри автор створив поєднавши грецький префікс «тетра-» зі словом «теніс» — улюбленою грою Пажитнова.

Метою розробки є демонстрація здобутих у курсі «Основи програмування - 2. Алгоритми та структури даних » навичок роботи з середовищем Qt.

Використані програмні засоби: Qt (середовище розробки), Advanced Installer (програма для створення інсталяційного пакету).

					UA.IT.74020-01 01	Лист
Вим.	Лист	№ докум.	Підпис	Дата		

1. ОБГРУНТУВАННЯ І ВИБІР АЛГОРИТМУ

Для реалізації інтерфейсу та інструментів растрового графічного редактора було використано багатоплатформовий фреймворк для розробки програмного забезпечення на мові програмування C ++.

У грі присутнє головне меню з навігацією, таблиця рекордів, саме вікно гри, у якому гравець може побачити нинішній рахунок, кількість знищених ліній та інше.

Основний алгоритм реалізовано у роботі зі шматочками тетрісу та полем, на якому вони знаходяться. З деяким інтервалом шматочок переміщується униз, після падіння з'являється новий. Також присутні бонусні шматки, які мають певні особливості, що робить гру більш захоплюючою.

					UA.IT.74020-01 01	Лист
Вим.	Лист	№ докум.	Підпис	Дата		

2. ПЕРЕЛІК ТА ПРИЗНАЧЕННЯ ІНСТРУМЕНТІВ

2.1 Перелік та функціонал інструментів та елементів меню

- «Start».Починає нову гру.
- «Leader Boards. Відображає таблицю рекордів.
- «Quit». Вихід з програми.
- «Pause». Ставить гру на паузу.
- «Main Menu». Повернення до головного меню.

					UA.IT.74020-01 01	Лист
Вим.	Лист	№ докум.	Підпис	Дата		

3. СТРУКТУРА ДАНИХ ТА РЕСУРСІВ ПРОГРАМИ

3.1 Розроблені класи

Для роботи програми був розроблений клас TetrixBoard. Він призначений для керування картою гри.

Перелік функцій класу TetrixBoard:

- int timeoutTime. Поле числового типу. Функція для обчислення швидкості оновлення поля.
- void clearBoard. Функція для виклику очистки поля.
- void oneLineDown. Функція зсуву елементів вниз.
- void pieceDropped. Функція, що викликається при падінні шматочка.
- void removeFullLines. Функція для перевірки повних ліній.
- void newPiece. Функція для створення нового шматочка.
- bool tryMove. Поле логічного типу. Функція для перевірки можливості зсуву шматочка.
- конструктор класу TetrixBoard.

Також був розроблений клас Piece та дочірні від нього класи squareBombPiece, horizontalLinePiece, lineBombPiece. Вони призначені для роботи безпосередньо зі шматочками (звичайними та бонусними).

Перелік функцій класу TetrixPiece:

- void droppedAction. Функція для виконання певних дій при падінні шматочка на полі.
- void setRandomShape. Функція для задання випадкової форми для шматочка.
- void updateBoard. Функція для оновлення стану поля після падіння шматочка.

					UA.IT.74020-01 01	Лист
Вим.	Лист	№ докум.	Підпис	Дата		

- TetrixPiece* returnCopyOfSelf. Функція, що повертає копію шматочка.
- TetrixPiece* GeneratePiece. Функція, що реалізує фабричний метод. Повертає екземпляр класу TetrixPiece.
- TetrixPiece* rotatedLeft. Функція, що перевертає шматочок вліво.
- TetrixPiece* rotatedRight. Функція, що перевертає шматочок вправо.
- конструктор класу TetrixPiece.

Найголовніший клас – це клас TetrixWindow. Він відповідає за інтерфейс усього, за зчитування та запис даних у файл та інше.

Перелік функцій класу TetrixWindow:

- void layoutClear. Функція для очищення інтерфейсу, приховання всіх віджетів на формі.
- void layoutToMainMenu. Функція для додання на форму тих віджетів інтерфейсу, які потрібні у головному меню.
- void layoutLeaderBoards. Функція для додання на форму тих віджетів інтерфейсу, які потрібні у вікні перегляду рекордів.
- void layoutInGame. Функція для додання на форму тих віджетів інтерфейсу, які потрібні під час гри.
- Void saveLog. Функція для збереження усієї необхідної інформації у файл.

3.2 Перелік використаних файлів

- main.cpp.
- board.h

					UA.IT.74020-01 01	Лист
Вим.	Лист	№ докум.	Підпис	Дата		

- drawable.h
- horizontalbomb.h
- linebombpiece.h
- piece.h
- squarebombpiece.h
- window.h
- windowstate.h
- board.cpp
- drawable.cpp
- horizontalbomb.cpp
- linebombpiece.cpp
- piece.cpp
- squarebombpiece.cpp
- window.cpp
- windowstate.cpp
- lastplayer.txt
- records.txt

					UA.IT.74020-01 01	Лист
Вим.	Лист	№ докум.	Підпис	Дата		

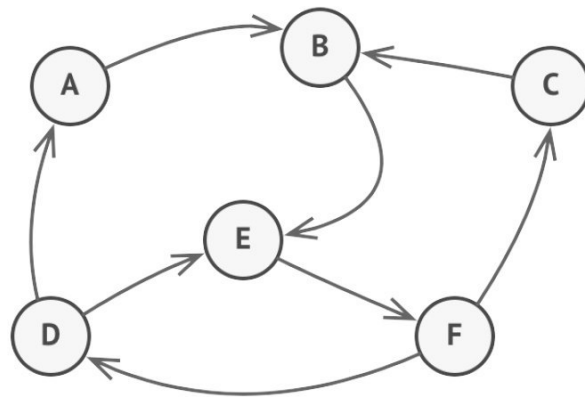
4. ОПИС ПРОГРАМИ

4.1 Опис функцій програми

Вся програма була написана з дотриманням принципів ООП, тому безпосередньо функцій самої програми немає. Я тільки функції-члени класів, які були створені задля створення гри. Окрім вище зазначених класів, також були створені класи, що реалізують деякі шаблони проектування, зокрема Factory Method, Singleton й State.

State(Стан) – це поведінковий патерн проектування, який дозволяє об'єктам змінювати поведінку залежно від свого стану. Ззовні створюється враження, що змінився клас об'єкта.

Основна ідея в тому, що програма може знаходитися в одному з декількох станів, які весь час змінюють один одного. Набір цих станів, а також переходів між ними, зумовлений і кінцевий. Перебуваючи в різних станах, програма може по-різному реагувати на одні і ті ж події, які відбуваються з нею.

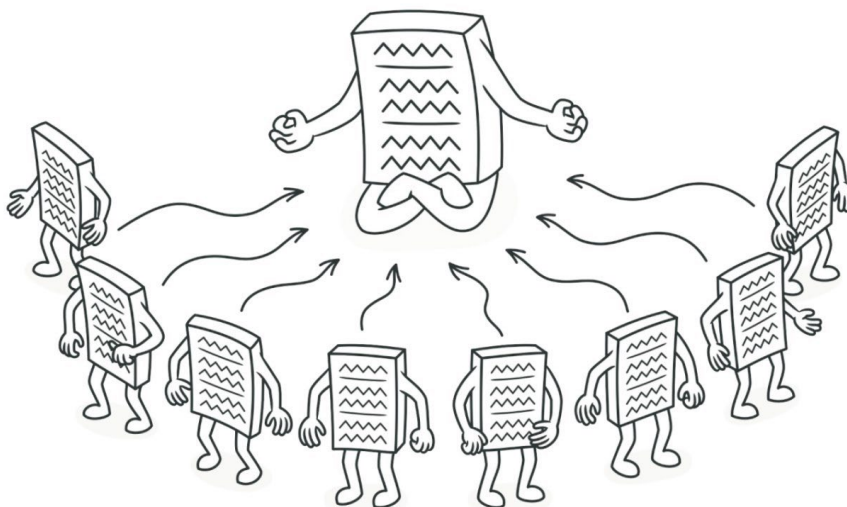


У програмі було реалізовано клас-стан WindowState, від якого наслідувалися класи MainMenuState, InGameState, ShowLeaderBoardsState. Вони мали віртуальну функцію windowAction(), яка була реалізована по різному, в залежності від стану, у якому знаходиться вікно програми. Це

вдалося зробити а допомогою використання поліморфізму на показників на класи.

Клас TetrixWindow, згаданий раніше, має у собі показчик на стан, у якому він знаходиться, і змінює його, коли користувач обирає певний пункт меню.

Singleton - це породжуючий патерн проектування, який гарантує, що у класу є тільки один екземпляр, і надає до нього глобальну точку доступу.



Singleton вирішує одразу ж дві проблеми:

1. Чи гарантує наявність єдиного екземпляра класу, найчастіше за все це корисно для доступу до якогось спільного ресурсу, наприклад, бази даних.

2. Надає глобальну точку доступу. Це не просто глобальна змінна, через яку можна достукатися до певного об'єкту. Глобальні змінні не захищені від запису, тому будь-який код може підміняти їх значення без вашого відома.

Всі реалізації Singleton-а зводяться до того, щоб приховати конструктор за замовчуванням і створити публічний статичний метод, який і контролюватиме життєвий цикл об'єкта- Singleton-а.

Якщо у вас є доступ до класу Singleton-a, значить, буде доступ і до цього статичного методу. З якої точки коду ви б його ні викликали, він завжди буде віддавати один і той же об'єкт.

Factory Method (Фабричний метод) - це породжуючий патерн проектування, який визначає загальний інтерфейс для створення об'єктів в суперкласі, дозволяючи підкласам змінювати тип створюваних об'єктів.

4.2 Вхідні та вихідні дані програми

- Вхідними даними є сигнали натискання кнопок миші та клавіш клавіатури.
- Файл з ім'ям останнього гравця
- Файл з рекордами
- Оновлена таблиця рекордів після гри

					UA.IT.74020-01 01	Лист
Вим.	Лист	№ докум.	Підпис	Дата		

5. ІНСТРУКЦІЯ ПРОГРАМІСТА

5.1 Умови проектування

Додаткове програмне забезпечення під час проектування програми не використовувалося. Середовище програмування Qt.

Мова програмування C++.

5.1.1 Характеристики платформи.

Програма проектувалася на ПК з наступними характеристиками:

- процесор Intel Core i5-6200U 2.40GHZ
- кількість оперативної пам'яті 16 ГБ.
- операційна система Windows 10 Pro x64

Для проектування програми не потрібні особливі периферійні пристрої. Достатньо комп'ютерної миші, клавіатури та монітору.

5.2 Керівництво розробника програми

5.2.1 Опис етапів проектування програми.

Етапи розробки програми:

- Qt треба завантажити з офіційного сайту Qt. Потім треба слідувати простій програмі встановлення середовища програмування.
- створюється каркас проекту, а саме форма головного меню
- тепер необхідно реалізувати усі необхідні функції на обробники подій для елементів раніше створеного інтерфейсу. Під час цього етапу доцільно користуватися офіційною документацією Qt [1].

					UA.IT.74020-01 01	Лист
Вим.	Лист	№ докум.	Підпис	Дата		

- як результат необхідно скомпілювати програму та зібрати виконавчий файл за допомогою стандартного інтерфейсу Qt.

5.2.2 Оформлення інсталяції.

Інсталяційний пакет оформлюється за допомогою програми Advanced Installer 14.8. У ньому треба обрати назву програми, вказати серед необхідного програмного забезпечення .Net Framework версії не менше ніж 4.5. Далі треба додати усі файли, які мають міститися у установчому пакеті.

					UA.IT.74020-01 01	Лист
Вим.	Лист	№ докум.	Підпис	Дата		

6. КЕРІВНИЦТВО КОРИСТУВАЧА

6.1 Загальні відомості

Програма призначена для дозвілля, вбивання часу, тренування реакції, концентрації.

6.2 Умови застосування програми

Єдиним необхідним програмним забезпеченням є .Net Framework версії не старше 4.5. Програма вимагає платформи на ОС Windows не старішій ніж Windows Vista. Для роботи з програмою необхідна комп'ютерна миша, клавіатура та монітор.

6.3 Характеристики програми

Дана реалізація Тетрісу має особливі шматочки, кожен з яких діє по різному. Також присутня таблиця рекордів, щоб додати у гру елемент змагання.

					UA.IT.74020-01 01	Лист
Вим.	Лист	№ докум.	Підпис	Дата		

6.4 Інсталяція та видалення

Для інсталяції програми треба запустити інсталяційний файл та слідувати простій інструкції інсталяції. Для видалення програми достатньо скористатися Панеллю керування, яка є вбудованою у ОС Windows.

6.5 Послідовність дій користування програмою

Після запуску програми потрапляє у головне меню, де йому пропонується ввести свій нікнейм. Потім користувач може переглянути таблицю рекордів або розпочати гру.

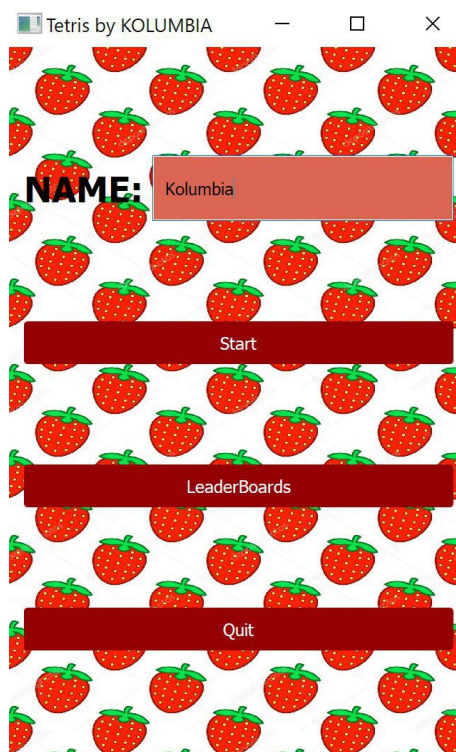


Рис. 6.1 – Головне меню.

					UA.IT.74020-01 01	Лист
Вим.	Лист	№ докум.	Підпис	Дата		

Обравши перше, користувач побачить таблицю найкращих десяти рекордів.



	Name	Score
1	KolumbiA	999999
2	San4ez	888888
3	Moonsinger	845638
4	msaliuta	685768
5	Strawberry	666666
6	gloomikon	123123
7	arpupsan	100100
8	PRO100PRO	45863
9	Kolumbia	1285
10	Nicole_Gloom	873

Main Menu

Рис. 6.2 – Рекорди

Під таблицею знаходиться кнопка “Main Menu”, що дозволяє користувачеві повернутися до головного меню.

Під час гри користувач може бачити свій рахунок, кількість знищених ліній на рівень складності. Чим вищий рівень, тим швидше падають шматочки. Також є кнопки, що дозволяють розпочати нову гру, повернутися до головного меню, поставити гру на паузу та вийти з гри.

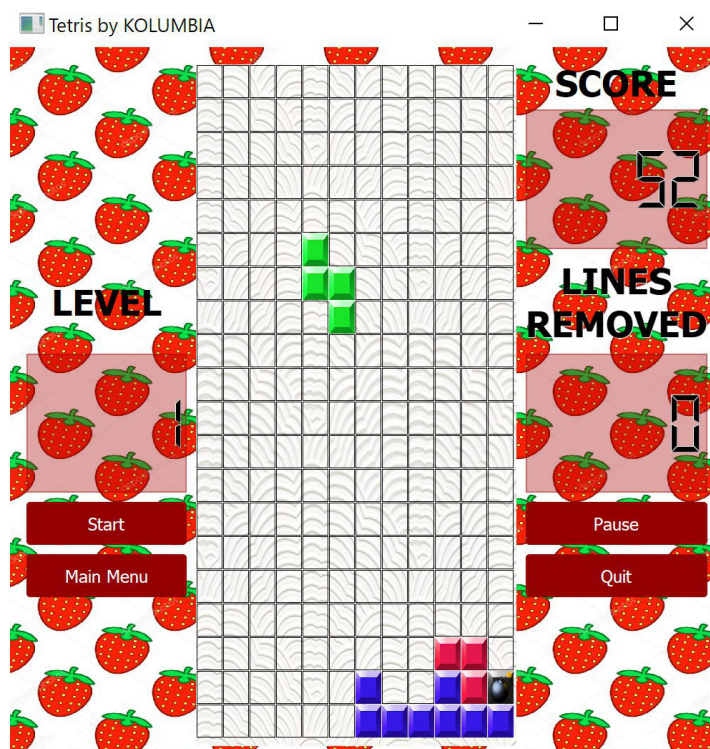


Рис 6.3 – Гра

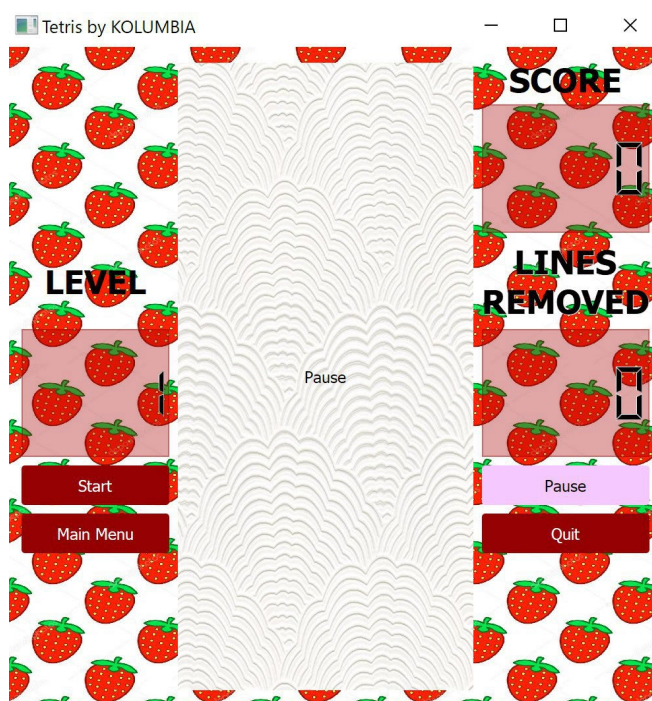


Рис. 6.4 – Пауза

ВИСНОВКИ

Під час виконання курсової роботи була спроектована нова версія Тетрісу, використовуючи знання та навички, здобуті під час проходження курсу Програмування – 2. Алгоритми та структури даних». Для реалізації був використаний фреймворк Qt. Був розроблений дружелюбний та зручний інтерфейс з усіма необхідними елементами управління. Програма була оформлена як установчий пакет з інсталяційним інтерфейсом а також можливістю видалити її через стандартну панель керування Windows у разі необхідності.

					UA.IT.74020-01 01	Лист
Вим.	Лист	№ докум.	Підпис	Дата		

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Офіційна документація Qt.

URL: <http://doc.qt.io/qt-5/reference-overview.html>

					UA.IT.74020-01 01	Лист
Вим.	Лист	№ докум.	Підпис	Дата		

ДОДАТОК UML

діаграма класів

