

Технические науки

Пахомова Євгенія Олександрівна

студент

Факультету прикладної математики

Національного технічного університету України

«КПІ ім. Ігоря Сікорського»

Пахомова Евгения Александровна

студент

Факультета прикладной математики

Национального технического университета Украины

«КПИ им. Игоря Сикорского»

Rakhomova Yevheniia

Student of the

Faculty of Applied Mathematics

National Technical University of Ukraine

«Igor Sikorsky KPI»

**ІМПЛЕМЕНТАЦІЯ ШАБЛОНІВ ПРОЕКТУВАННЯ У ПРОГРАМНУ
СТРУКТУРУ ІГРОВОГО ПРОЦЕСУ**

**ИМПЛЕМЕНТАЦИЯ ШАБЛОНОВ ПРОЕКТИРОВАНИЯ В
ПРОГРАММНУЮ СТРУКТУРУ ИГРОВОГО ПРОЦЕССА**

**IMPLEMENTATION OF DESIGN PATTERNS TO THE PROGRAM
STRUCTURE OF THE GAME PROCESS**

Анотація. У роботі наведено аналіз обраних різномісних шаблонів проектування та обґрунтування їх імплементації для створення оптимальної за стійкістю програмної структури, що описує логічний ігровий процес.

Ключові слова: шаблони проектування, об'єктно-орієнтоване програмування, логічна гра.

Аннотация. В работе представлен анализ выбранных шаблонов проектирования и обоснование их имплементации для создания оптимальной по стойкости программной структуры, описывающей логичный игровой процесс.

Ключевые слова: шаблоны проектирования, объектно-ориентированное программирование, логическая игра.

Summary. The paper presents an analysis of selected design patterns and rationale for their implementation for creating the optimal stable software structure that describes the logical game.

Key words: design patterns, object-oriented programming, logical game.

При створенні стійкого завдяки різнотипним шаблонам проектування програмного продукту доцільно спочатку абстрагувати об'єкти предметної галузі. Потім пропонується проаналізувати та обґрунтувати вибір шаблонів проектування та імплементувати їх у програмну структуру. На основі отриманих даних виконати реалізацію програмного засобу логічного ігрового процесу, а саме популярної головоломки «Гра у 15».

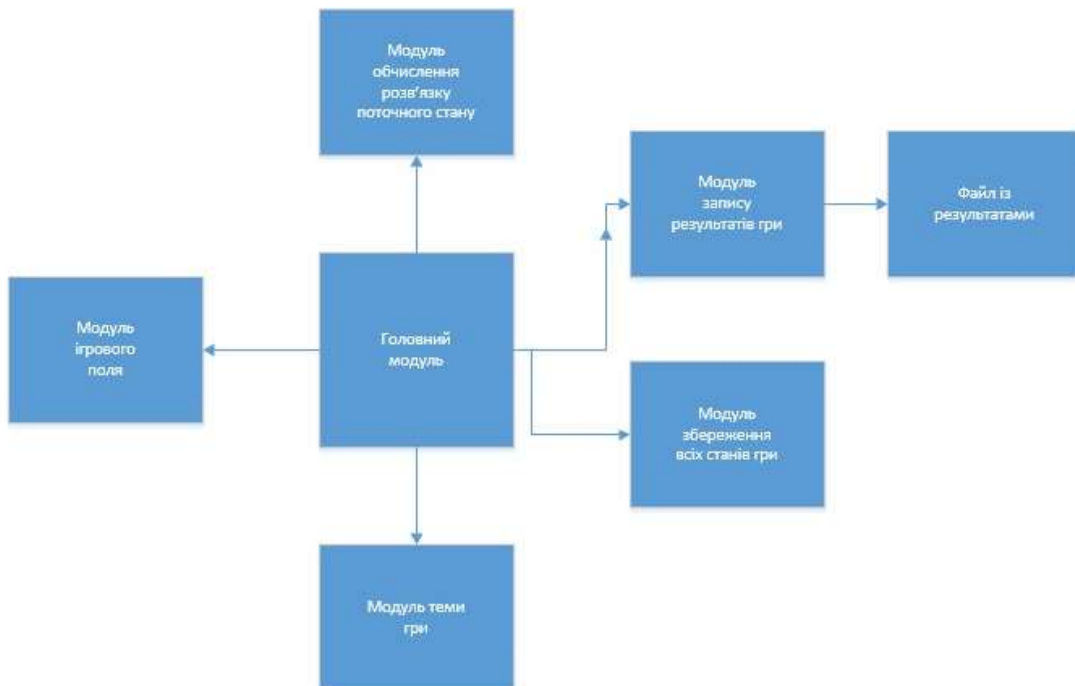
«Гра у 15» - головоломка що складається з 15 однакових квадратних пластин з числами від 1 до 15. Пластинки поміщаються в квадратне поле, довжина сторони якого в чотири рази більша довжини сторони пластинок, відповідно в полі залишається незаповненим одне місце. Мета гри — переміщаючи пластинки по полю добитися впорядкування їх по номерах (як зображено на рисунку), бажано зробивши якомога менше переміщень.



Розв'язок головоломки

1. Модульна організація програми

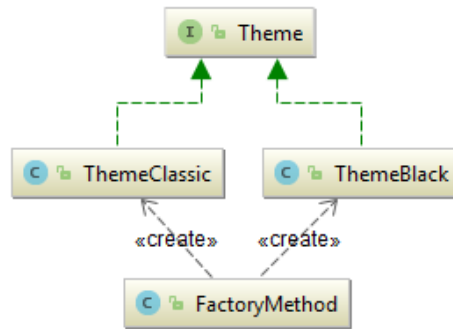
Абстрагування об'єктів предметної галузі дає змогу створити модульну організацію програми:



Модульна організація програми

2. Аналіз та обґрунтування використання різнотипних шаблонів проектування

2.1. Factory method



Визначає інтерфейс для створення об'єкта, але залишає підкласам рішення про те, який саме клас інстанціювати. Фабричний метод дозволяє класу делегувати інстанціювання підкласам.

Даний шаблон використовується для створення єдиного інтерфейсу через фабрику в залежності від натиснутої кнопки теми.

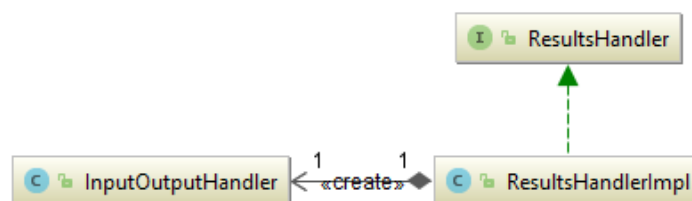
2.2. Singleton



Відноситься до класу твірних шаблонів. Гарантує, що клас матиме тільки один екземпляр, і забезпечує глобальну точку доступу до цього екземпляра.

Ігрова панель має бути лише одна аж до припинення роботи з програмою. Якщо в якомусь місці буде створено ще хоча б один екземпляр, будуть великі зайві втрати пам'яті. Шаблон Singleton створений, щоб не допустити цього.

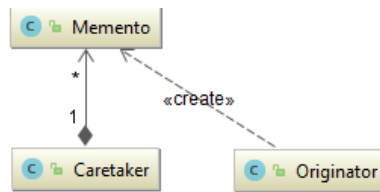
2.3. Adapter



Структурний шаблон проектування, призначений для організації використання функцій об'єкта, недоступного для модифікації, через спеціально створений інтерфейс.

Користувачу байдуже, куди будуть записуватися і звідки зчитуватися результати. Для цього йому наданий інтерфейс ResultsHandler, що дозволяє це робити. Для даної версії програми запис і зчитування йде з файлу. Але із модифікаціями програми цього вже може бути недостатньо і записи потрібно буде виконувати до бази даних. Не змінюючи інтерфейс і реалізацію ResultsHandler можливо тільки замінити методи InputOutputHandler на ті, що будуть записувати до бази.

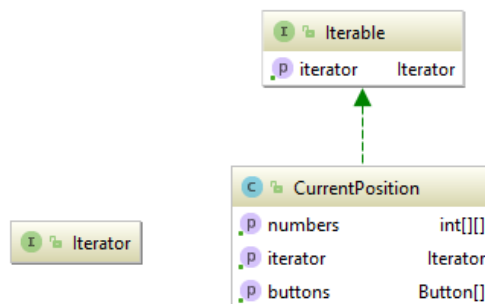
2.4. Memento



Шаблон проектування, відноситься до класу шаблонів поведінки. Не порушуючи інкапсуляції, фіксує та виносить за межі об'єкта його внутрішній стан так, щоб пізніше можна було відновити з нього об'єкт.

Шаблон був використаний для зберігання усіх станів ігрового поля аж від самого початку. Це дає користувачу змогу повернутися до будь-якої позиції і перепланувати свої кроки, адже і кількість кроків при цьому віднімається назад.

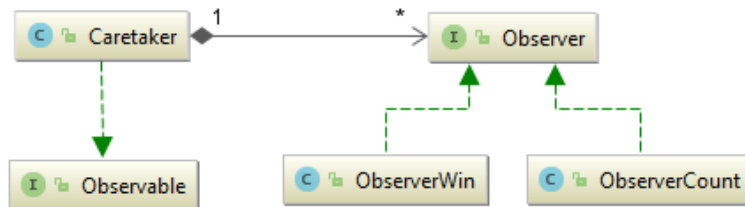
2.5. Iterator



Шаблон проектування, належить до класу шаблонів поведінки. Надає спосіб послідовного доступу до всіх елементів складеного об'єкта, не розкриваючи його внутрішнього улаштування.

Шаблон використаний для зручної ітерації всіх кнопок поточного стану. Так як в основному взаємодія вимагає всіх кнопок, а не окремих, шаблон є доречним до використання.

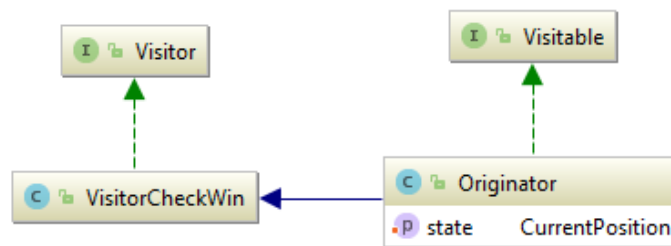
2.6. Observer



Поведінковий шаблон проектування. Визначає залежність типу «один до багатьох» між об'єктами таким чином, що при зміні стану одного об'єкту всіх залежних від нього сповіщають про цю подію.

Шаблон використаний для спостереження за станами гри. ObserverWin при кожному новому перевірляє, чи не є він виграшним, щоб завершити дану гру. ObserverCount виконує підрахунок кроків, що зробив користувач. Використані для того, щоб не нагромаджувати логіку класу CareTaker.

2.7. Visitor



Шаблон проектування, який дозволяє відділити певний алгоритм від елементів, на яких алгоритм має бути виконаний, таким чином можливо легко додати або ж змінити алгоритм без змін щодо елементів системи. Практичним результатом є можливість додавання нових операцій в існуючі структури об'єкта без зміни цих структур.

Шаблон використаний для перевірки поточного стану на виграш. Якщо стан таким не є, гра продовжується, якщо навпаки – створює віконце, що пропонує записати свої результати та ім'я.

2. Фримен Э., Фримен Э., Сьерра К., Бейтс Б. Паттерны проектирования. — СПб.: Питер, 2011. — 656 с.: ил.
3. Design Patterns: Elements of Reusable Object-Oriented Software / Erich Gamma, Richard Helm, Ralph Johnson, John M. Vlissides // Шаблоны проектирования - 2012. - С.30-69.