

ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ СИСТЕМИ ПОБУДОВИ ЛАБІРИНТІВ НА ОСНОВІ МЕТОДІВ ТЕОРІЇ ГРАФІВ

Пурхало М.О., науковий керівник Василюк-Зайцева С.В.

Актуальність. Алгоритмічні лабіринти давно перестали бути простою розвагою. У сучасній ігровій індустрії процедурно згенеровані рівні з непередбачуваними маршрутами знижують витрати на ручне проєктування. У робототехніці лабіринти слугують модельними середовищами для перевірки систем навігації, побудови карт і ухилення від перешкод, дозволяючи безпечно тестувати алгоритми у складних «кутових» конфігураціях. Середовища-лабіринти є стандартом у задачах навчання з підкріпленням і застосовуються для тренування агентів у DeepMind Lab і OpenAI Gym, оскільки дають змогу оцінити стратегічне планування та просторову пам'ять. Освітні платформи використовують наочну генерацію різних методів, аби зробити абстрактну теорію графів зрозумілішою студентам.

Крім того, лабіринти знаходять застосування у VR-квестах, тренажерах евакуації та навіть у 3D-друку декоративних елементів. Водночас більшість наявних інструментів підтримує лише один-два методи генерації. Розробка системи, що об'єднує п'ять різних алгоритмів, дозволяє паралельно створювати кілька лабіринтів, миттєво їх візуалізувати у 2D і експортувати у формати, придатні для інтеграції в ігрові рушії чи дослідницькі середовища, відповідає потребам індустрії, науки й освіти.

Об'єкт дослідження — процес процедурної генерації лабіринтів.

Предмет дослідження — алгоритмічні та програмні засоби реалізації зазначених п'яти методів.

Мета роботи. Метою роботи є створення системи, яка генерує лабіринти алгоритмами Прима, Крускала, DFS (пошук у глиб), Еллера та рекурсивного поділу, дозволяє одночасно запускати до трьох процесів генерації та за потреби перезапустити окремі з них. Система має забезпечувати двовимірний огляд. Передбачено експорт результатів у FBX, PDF, PNG та SVG, та проходження лабіринту за потреби користувача.

Ключові слова: лабіринт; теорія графів; алгоритм Прима; алгоритм Крускала; DFS (пошук у глибину); C#; 3D-експорт; паралельні обчислення.

Нижче наведено компонентну діаграму, що відображає логічну структуру програмної системи та потоки даних між шістьма ключовими підсистемами — інтерфейсом користувача, генерацією, проходженням, візуалізацією та експортом файлів. Діаграма допомагає чітко зрозуміти межі відповідальності кожного модуля, порядок взаємодії під час створення, візуалізації та використання лабіринту, а також демонструє, як користувацькі дії трансформуються у послідовність обчислень і результатів.

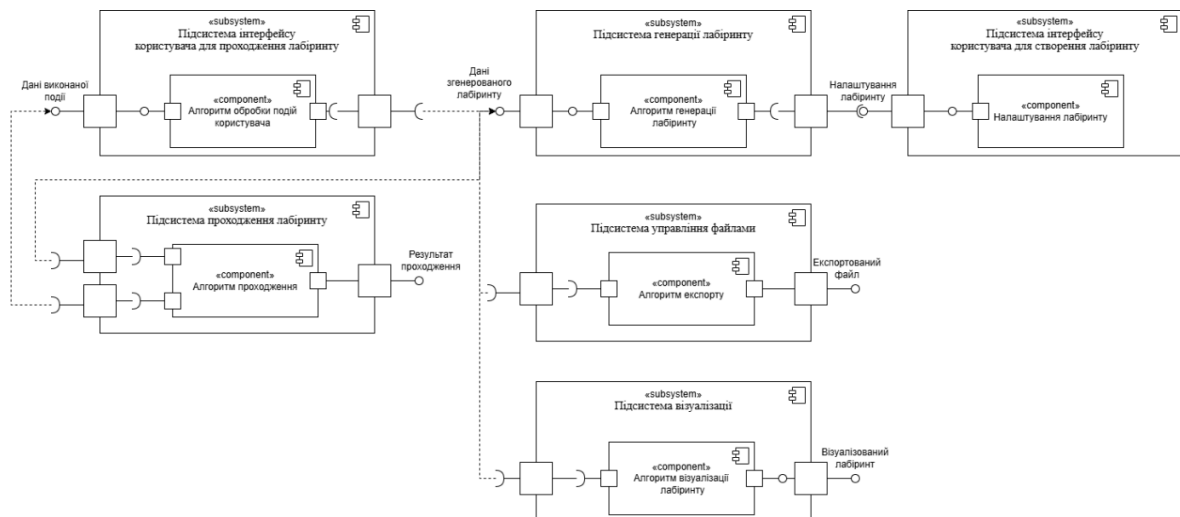


Рис. 6 Діаграма компонентів

1. Підсистема інтерфейсу користувача для проходження лабіринту

Користувач ініціює подію — «Дані виконаної події» надходять у компонент Алгоритм обробки подій користувача. Компонент перетворює подію на зрозумілу системі команду й передає її далі двома шляхами:

- в Підсистему проходження лабіринту;

2. Підсистема генерації лабіринту

Компонент Алгоритм генерації лабіринту отримує запит та параметри (висота, ширина, метод) і створює структуру лабіринту. Результат у вигляді «Даних згенерованого лабіринту» відправляється чотирма напрямками:

- у Підсистему інтерфейсу користувача — для відображення;
- у Підсистему проходження лабіринту — для проходження
- у Підсистему управління файлами — якщо користувач натиснув «Експорт»;
- у Підсистему візуалізації, де формується 2D-вигляд.

3. Підсистема інтерфейсу користувача для створення лабіринту

Блок, де задаються «Налаштування лабіринту» (метод, розміри, анімація). Після підтвердження параметри потрапляють у Алгоритм генерації лабіринту.

4. Підсистема проходження лабіринту

Складається з компонента Алгоритм проходження, який приймає:

- «Дані згенерованого лабіринту»;
- «Дані виконаної події» — кроки користувача.

Алгоритм обчислює нову позицію і повертає «Результат проходження» назад у UI для оновлення відображення.

5. Підсистема управління файлами

Компонент Алгоритм експорту отримує готовий лабіринт і, врахувавши вибраний формат (FBX/PNG тощо), формує файл, який зберігається у вибраній директорії.

6. Підсистема візуалізації

Алгоритм візуалізації лабіринту приймає структурні дані й генерує «Візуалізований лабіринт» (2D-рисунок), який повертається у вікно перегляду.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. «Алгоритм створення лабіринту» Вікіпедія. Режим доступу: https://uk.wikipedia.org/wiki/Алгоритм_створення_лабіринту
2. Nakka S. M. Reddy (2024) A Comparative Study of Maze-Solving Algorithms: Performance, Complexity, and Practical Applications in AI and Robotics. Available at: https://ijariie.com/AdminUploadPdf/A_Comparative_Study_of_Maze_Solving_Algorithms

[Performance_Complexity_and_Practical_Applications_in_AI_and_Robotics_ijariie25179.pdf](#)

**НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ БІОРЕСУРСІВ І
ПРИРОДОКОРИСТУВАННЯ УКРАЇНИ
ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ**



**ТЕОРЕТИЧНІ ТА ПРИКЛАДНІ
АСПЕКТИ РОЗРОБКИ
КОМП'ЮТЕРНИХ СИСТЕМ**

**ЗБІРНИК НАУКОВИХ ПРАЦЬ ЗА МАТЕРІАЛАМИ
VII ВСЕУКРАЇНСЬКОЇ НАУКОВО-ПРАКТИЧНОЇ КОНФЕРЕНЦІЇ
СТУДЕНТІВ І АСПІРАНТІВ
*24 квітня 2025 року***

Київ 2025

УДК 004

Відповідальна за випуск: М.І. Лендел

Збірник наукових праць за матеріалами VII Всеукраїнської науково-практичної конференції студентів і аспірантів «ТЕОРЕТИЧНІ ТА ПРИКЛАДНІ АСПЕКТИ РОЗРОБКИ КОМП'ЮТЕРНИХ СИСТЕМ '2025», 24 квітня 2025 року, НУБіП України, Київ. – 452 с. (електронне видання)

Відповідальність за зміст публікацій несуть автори.

Передрук матеріалів, а також використання їх будь-якій формі допускається лише з дозволу авторів