

ПРОГРАМНА СИСТЕМА ПРОЦЕДУРНОЇ ПАРАМЕТРИЧНОЇ ГЕНЕРАЦІЇ ВІРТУАЛЬНИХ ЛОКАЦІЙ ДЛЯ ГРИ В ЖАНРІ ПЛАТФОРМЕР

Ольшанський Д.В., науковий керівник Назаренко В.А.

Актуальність. Комп'ютерні ігри з моменту своєї появи, завоювали серця користувачів персональних комп'ютерів, однак розробка ігор ніколи не була простим завданням. Перш за все, варто згадати, що ігри, як і будь-який ІТ-продукт, створюється в результаті складного, багаторівневого процесу та проходить через масу етапів, починаючи формування ідеї та концепції, завершуючи оптимізацією та підтримкою готового продукту. Сучасний рівень розвитку індустрії комп'ютерних ігор вимагає швидкого створення великої кількості унікального ігрового контенту, що особливо стосується інді-розробки, де часто розробники мають велику кількість обмежень в ресурсах. Однією з найскладніших частин є створення ігрових локацій що потребують балансу між цікавим дизайном, можливістю використовувати їх в грі та зацікавленістю гравцями. Процедурна генерація рівнів дозволяє автоматизувати цей процес, а створення інструменту безпосередньо у середовищі рушія підвищує ефективність і дозволяє інтегрувати його в будь-який проєкт. Актуальність дослідження зумовлена потребою в таких інструментах у галузі ігрової розробки, особливо для 2D-платформерів, які потребують численних рівнів.

Мета дослідження є розробка плагіну для Unity Editor, що дозволяє генерувати 2D-локації для гри в жанрі платформер за допомогою методу random walk, з можливістю параметричного налаштування.

Методи дослідження. Дослідження проводилося в середовищі Unity 6 (2023.1 і вище) із використанням мови програмування C#. Для реалізації інтерфейсу було використано класичний редакторський API — EditorWindow у поєднанні з GUILayout, що дозволило створити повноцінне вікно з параметрами генерації, інтегроване в Unity Editor, яке має всі властивості вікон редактору Unity з можливістю змінювати розмір, «прикріплювати вікно до панелей», «Drag and drop» та інші. Основним алгоритмом генерації карт обрано Random Walk — метод випадкового переміщення по сітці, який дозволяє створювати звивисті коридори та лабіринтоподібні структури. Алгоритм адаптовано під логіку платформерних рівнів: він враховує межі карти, заповнює внутрішній простір «платформами», що відповідає основам жанру гри, запобігає виходу за межі, а також дозволяє гнучко задавати кількість кроків та стартову точку, властивості генерації оточення, імовірності появи елементів генерації, а також плагін містить потужну систему візуалізації, для якої використовувалась система Tilemap, а саме TilemapRenderer та Tilemap, які були інтегровані в редакторську сцену автоматично після натискання кнопки "Згенерувати", або призначені користувачем плагіну самостійно, чи автоматично визначені на сцені у випадку, якщо вони були присутні. Результат відображався одразу, без потреби запуску Play Mode, що забезпечило зручність для розробника, та можливість додавати готові ігрові об'єкти відповідно до ігрових сценаріїв. Дані про параметри генерації зберігались через ScriptableObject, що дозволяє зберігати шаблони генерації та використовувати їх повторно на інших сценах або в інших проєктах. Тестування відбувалось у межах кількох тестових сцен Unity, де перевірялась стабільність генерації, уникнення зайвого навантаження на редактор, а також відповідність структури карти умовам платформеру.

Результати. У ході розробки програмної системи було здійснено дослідження алгоритмів процедурної генерації, зокрема методу random walk, який

було обрано як основний підхід до побудови віртуальних рівнів. На основі цього дослідження розроблено архітектуру плагіна з підтримкою EditorWindow, що дозволяє взаємодіяти з користувачем безпосередньо в Unity Editor. Окрему увагу було приділено реалізації візуалізації результату генерації, яка здійснюється через Tilemap, а також забезпеченню збереження конфігурацій за допомогою ScriptableObject. Завершальним етапом стала перевірка працездатності плагіна на серії тестових сцен Unity, з метою виявлення помилок і перевірки відповідності очікуваному функціоналу. У результаті проведеного дослідження отримано вагомий практичний результат. Розроблено повноцінний плагін для Unity Editor, який реалізовано у вигляді окремого вікна на базі EditorWindow. Плагін дозволяє генерувати 2D-рівні для платформера методом random walk із гнучкими параметрами: розміри карти, кількість кроків, стартова позиція тощо. Інтерфейс користувача побудовано з використанням GUILayout, що забезпечує простий і зручний механізм введення параметрів. Генерація карти відбувається без необхідності запуску Play Mode — результат з'являється одразу в сцені, а також може бути очищений за потреби. Сам алгоритм random walk було адаптовано до потреб платформерного жанру. Було реалізовано контроль меж карти, щоб уникнути виходу за її межі, а також механізм перевірки унікальності відвіданих клітин для уникнення надлишкового повторення. Додано можливість задавати різні початкові seed-параметри для отримання унікальних результатів. Візуалізація карти відбувається через Tilemap і TilemapRenderer, що дозволяє бачити структуру рівня в реальному часі та миттєво оцінювати результат генерації. Збереження параметрів генерації здійснюється за допомогою ScriptableObject, що дає змогу створювати шаблони конфігурацій і використовувати їх повторно в інших сценах або проєктах. Це значно спрощує роботу над великими ігровими світами, робить плагін масштабованим і гнучким, а сам процес генерації — контрольованим і стабільним.

Висновки. Враховуючи розвиток Unity, та сфери розробки ігор, плагін, що був створений в результаті дослідження, матиме позитивний вплив на продуктивність створення ігрових програмних продуктів. Відкритий код даного розширення дозволить розробникам ігор гнучко налаштовувати та змінювати функціонал під потреби своїх проєктів, а також наявний потенціал інтеграції з такими новими технологіями, як UI Toolkit, Graph-based Editor Tools, або навіть DOTS для швидшої генерації у великих масштабах. Він може бути розширений шляхом додавання нових алгоритмів, що дозволить створювати локації не лише для платформерів, але й для 2D ігор в інших жанрах.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Shaker, N., Togelius, J., & Nelson, M. J. (2016). Procedural content generation in games. Springer. <https://doi.org/10.1007/978-3-319-42716-4>
2. Kerssemakers, M., Tuxen, J., Togelius, J., & Yannakakis, G. N. (2012, September). A procedural procedural level generator generator. In 2012 IEEE Conference on Computational Intelligence and Games (CIG) (pp. 335–341). IEEE. <https://doi.org/10.1109/CIG.2012.6374167>
3. Yadav, A. (2019, June 25). Exploring procedural generation in Unity: Tools and techniques. Medium. <https://medium.com/@yadavaman/exploring-procedural-generation-in-unity-tools-and-techniques-e0f16e88e2f2>
4. Nwankwo, G., Mohammed, S., & Fiadhi, J. (2017). Procedural content generation for dynamic level design and difficulty in a 2D game using Unity. International Journal of Multimedia and Ubiquitous Engineering, 12(9), 41–52. <https://doi.org/10.14257/ijmue.2017.12.9.04>

**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