

УДК 004.42: 004.43

Ушаков М.О.¹, Дейнега Л.Ю.²,

¹ студ. гр. КНТ-133 НУ «Запорізька політехніка»

² старш. викл. НУ «Запорізька політехніка»

РОЗВИТОК МОВ ПРОГРАМУВАННЯ І ГРАФІЧНИХ БІБЛІОТЕК

Початки програмування були надзвичайно складними, оскільки перші програми писалися безпосередньо на машинній мові, тобто у вигляді довгих послідовностей нулів і одиниць [1]. Будь-яка помилка могла призвести до збоїв, а сам процес програмування був дуже громіздким і повільним. Щоб спростити роботу програмістів, з'явилися асемблери — мови низького рівня, які використовували мнемонічні коди замість числових інструкцій процесора [1]. Наприклад, у Assembly програміст міг написати MOV AX, BX замість довгих двійкових кодів. Проте асемблери все ще вимагали глибокого розуміння архітектури комп'ютера і не давали суттєвої абстракції для складних завдань.

У 1950–1960-х роках виникли перші високорівневі мови програмування, які дозволяли програмістам мислити більш абстрактно [1]. Важливу роль у розвитку високорівневих мов відіграла українка Катерина Ющенко, яка працювала над формалізацією структур даних, алгоритмів і методів абстрактного програмування. Створена нею в 1955 році Адресна мова стала першою в світі мовою програмування високого рівня і прототипом інших алгоритмічних мов. Найважливішою та революційною особливістю Адресної мови була реалізація непрямої адресації вищих рангів, концепції для динамічного виділення пам'яті, складних структур даних та об'єктно-орієнтованого програмування. Ця інновація дозволяє використовувати реляційні бази даних та розпізнавання зображень на основі машинного навчання [2].

Мова Fortran, створена у 1957 році, була орієнтована на наукові та інженерні розрахунки, проте мала обмежену підтримку структурованого коду та роботу з пам'яттю. Мова Cobol, розроблена у 1959 році для бізнес-логіки та бухгалтерії, відрізнялася громіздким синтаксисом і обмеженою гнучкістю. Мова Lisp, з'явившись у 1958 році, була спрямована на штучний інтелект і обробку символів, проте її специфічний синтаксис і складність оптимізації обмежували широку популярність [1]. Ці мови відкрили нові можливості для програмістів, але залишали багато проблем у роботі з великими системами.

У 1970-х і 1980-х роках відбувся перехід до структурованого програмування та більш універсальних мов. C, створена у 1972 році, дозволила писати ефективні системні програми і стала основою багатьох операційних систем, включно з Unix [1]. Pascal, розроблена у 1970 році, популяризувала навчання структурованому програмуванню та правильному організованому коду.

Поява високорівневих мов дала потужний імпульс розвитку комп'ютерної графіки та ігрової індустрії. Високорівневі мови дозволили створювати складні програми, не занурюючись у деталі машинного коду. Концепція gamedev з'явилася як спосіб використання мов високого рівня для швидкої розробки ігор та генерації контенту. Це також стало основою для розвитку графічних API, таких як OpenGL, кросплатформенний стандарт для 2D і 3D-графіки; DirectX від Microsoft для Windows і ігрових консолей; PhysX для реалізації фізики в іграх; Vulkan, який забезпечує низькорівневий доступ до графічного і обчислювального апаратного забезпечення [3]. Всі ці технології спираються на високорівневі мови програмування, такі як C++ або C#, що дозволяє програмістам створювати візуально складні ігрові світи, анімацію, освітлення та фізичні симуляції без написання сотень тисяч рядків низькорівневого коду.

Розвиток високорівневих мов і графічних API призвів до створення сучасних ігрових рушіїв, які значно спрощують розробку ігор. Unity використовує C# і дозволяє створювати кросплатформенні ігри швидко і ефективно. Unreal Engine побудований на C++ та має власну систему Blueprints для візуального скриптингу. Godot використовує GDScript, високорівневу мову, схожу на Python, що спрощує процес навчання та розробки 2D і 3D ігор. Ці рушії дозволяють інтегрувати фізику, освітлення, шейдери та анімацію, роблячи створення ігор значно швидшим і доступнішим.

Таким чином, еволюція мов програмування пройшла шлях від машинного коду і асемблерів до високорівневих мов, які дозволяють програмістам мислити абстрактно і створювати складні системи [1]. Високорівневі мови стали основою для графічних API і ігрових рушіїв, що відкрили двері для сучасної комп'ютерної графіки, симуляцій і глобальної

ігрової індустрії, а внесок Катерини Ющенко заклав фундамент для формування цих потужних технологій.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Arusoai A. Principles of Programming Languages: A little bit of history. URL: https://edu.info.uaic.ro/principii-ale-limbajelor-de-programare/2025/week1/1_intro.pdf.

2. Катерина Ющенко: донька "ворога народу" і першопрохідниця у світі програмування. URL: <https://perciaslav.city/articles/380788/katerina-yuschenko-donka-voroga-narodu-i-persha-programistka>.

3. Catanese S. A. et al. Rendering of 3D Dynamic Virtual Environments. 2011. URL: https://www.researchgate.net/profile/Francesco-Pagano-5/publication/50435101_Rendering_of_3D_Dynamic_Virtual_Environments/links/6734da9da78ba469f060acde/Rendering-of-3D-Dynamic-Virtual-Environments.pdf.