# **МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**

КРЕМЕНЧУЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМЕНІ МИХАЙЛА ОСТРОГРАДСЬКОГО

НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ЕЛЕКТРИЧНОЇ ІНЖЕНЕРІЇ ТА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

КАФЕДРА АВТОМАТИЗАЦІЇ ТА ІНФОРМАЦІЙНИХ СИСТЕМ

НАВЧАЛЬНА ДИСЦИПЛІНА

«**АЛГОРИТМИ І СТРУКТУРИ ДАНИХ**»

# **ЗВІТ З ПРАКТИЧНОЇ РОБОТИ №1**

Виконав:

студент групи КН-24-1

Дон.А.А

Перевірив:

доцент кафедри АІС

Сидоренко В.М.

# **Кременчук 2025**

Тема: Асимптотична складність алгоритмів. 𝐎-нотація

Мета: набути практичних навичок у розв’язанні задач на оцінку асимптотичної складності алгоритмів у 𝑂.

# **Хід роботи**

1. Ознайомитися з теоретичним матеріалом та прикладами виконання завдання
2. Визначити своє завдання.

Довести, щоf(n) = 3n^3 + 5 = O(n^3)

1. Розв’язати завдання.

Щоб довести, щоf(n) = 3n^3 + 5 = O(n^3),

скористаємося означенням О-нотації.

Визначення:Функція f(n) = O(g(n)), якщо існують додатні константи c і n₀,

такі що для всіх n ≥ n₀ виконується:f(n) ≤ c ⋅ g(n)

У нашому випадку:- f(n) = 3n^3 + 5- g(n) = n^3

Покажемо, що існують такі c і n₀, що:3n^3 + 5 ≤ c ⋅ n^3 для всіх n ≥ n₀

Поділимо обидві частини нерівності на n^3 (для n > 0):3 + 5/n^3 ≤ c

При збільшенні n, дробовий доданок 5/n^3 → 0.Візьмемо n₀ = 2.

Тоді при n ≥ 2:3 + 5/8 = 3.625Отже, можна взяти c = 4.

Таким чином, при n ≥ 2:f(n) = 3n^3 + 5 ≤ 4n^3

Висновок:Отже, за означенням,f(n) = 3n^3 + 5 = O(n^3)(при c = 4, n₀ = 2)

Надати відповідь на контрольні запитання.

1.Асимптотична складність алгоритму це характеристика , яка визначає відношення часу виконання певної дії до кількості вхідних даних у алгоритм , подається у вигляді математичних виразів з використанням нотації , як наприклад велика О-нотація яку було використано у даній практичній роботі.

2.О-нотація визначається так , що певна f зростає з такою ж швидкістю або не швидше ніж певна g , фактично О-нотація являє собою асимптотичну верхню межу часу виконання певної дії.

3.Перше правило – член із найвищим темпом зростання домінує , роблячи незначним внесок решти членів у темп зростання всієї функції.

Друге правило – фактори , які не залежать від аргументу функції або константи можна відкидати.

4.Вони вказують на те , як залежить час на виконання завдання від кількості вхідних даних , наприклад O(1) в ньому час не залежить від кількості вхідних даних , а у O(n) залежить.

5.Для того , щоб визначити асимптотичну складність алгоритму треба проаналізувати функцію за правилами , що наведені вищі , а саме відкинути не значущі елементи , такі як константи та члени з меншим темпом зростання , і тоді ми отримаємо результат , як приклад можна використати виконані завдання з цього звіту.

Висновки : На цій практичній роботі зрозумів , що таке асимптотична складність алгоритму та , що таке О-нотація . Навчився розв’язувати завдання на оцінку асимптотичної складності алгоритмів.