

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  
Клінген К. В.

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.

8. Довести, що  $f(n) = 2n^2 + 10n + 3 = O(n^2)$ .

13. Задано функції  $f(n) = 100n^3 + 8$  та  $g(n) = n^3$ . Доведіть, що  $f(n) = O(g(n))$ .

3. Розв'язати завдання.

8. Довести, що  $f(n) = 2n^2 + 10n + 3 = O(n^2)$ .

$$f = O(n^2)$$

$$2n^2 + 10n + 3 \leq C \cdot n^2 \quad \forall n \geq n_0$$

$$2n^2 + 10n + 3 \leq 2n^2 + 10n^2 + 3n^2 \quad \forall n \geq 1$$

$$15n^2$$

$$C = 15 \quad n_0 = 1$$

13. Задано функції  $f(n) = 100n^3 + 8$  та  $g(n) = n^3$ . Доведіть, що  $f(n) = O(g(n))$ .

$$\begin{aligned} 100n^3 + 8 &\leq C \cdot n^3 \quad \forall n \geq n_0 \\ 100n^3 + 8 &\leq 100n^3 + 8n^3 \quad \forall n \geq 1 \\ 108n^3 \\ C &= 108 \quad n = 1 \end{aligned}$$

4. 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 вище, 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  $O$ -нотація . Навчився розв'язувати завдання на оцінку асимптотичної складності алгоритмів.