МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ КРЕМЕНЧУЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМЕНІ МИХАЙЛА ОСТРОГРАДСЬКОГО

Кафедра комп’ютерної інженерії та електроніки

ЗВІТ З ПРАКТИЧНОЇ РОБОТИ №1

з навчальної дисципліни

**«**Алгоритми та методи обчислень**»**

Тема «Асимптотична складність алгоритмів. O-нотація»

Студентка гр. КН-24-1 Бояринцова П. С.

Викладач Сидоренко В. М.

Кременчук 2025

**Тема роботи:** Асимптотична складність алгоритмів. O-нотація

## Постановка завдання.

Ознайомитися з теоретичними основами асимптотичного аналізу алгоритмів. Виконати індивідуальні завдання згідно з варіантом. Відповісти на контрольні питання.

## Розв’язання задачі згідно зі своїм варіантом.

**Задача 3:** Дано функцію . Знайти асимптотичну складність у O-нотації.

Розв’язання:

Не зважаємо на константи:

За правилом спрощення, при великому *n* домінуючим членом є *n2*, оскільки він зростає швидше за log(n) (рис. Рисунок 1). Отже, асимптотична складність:.

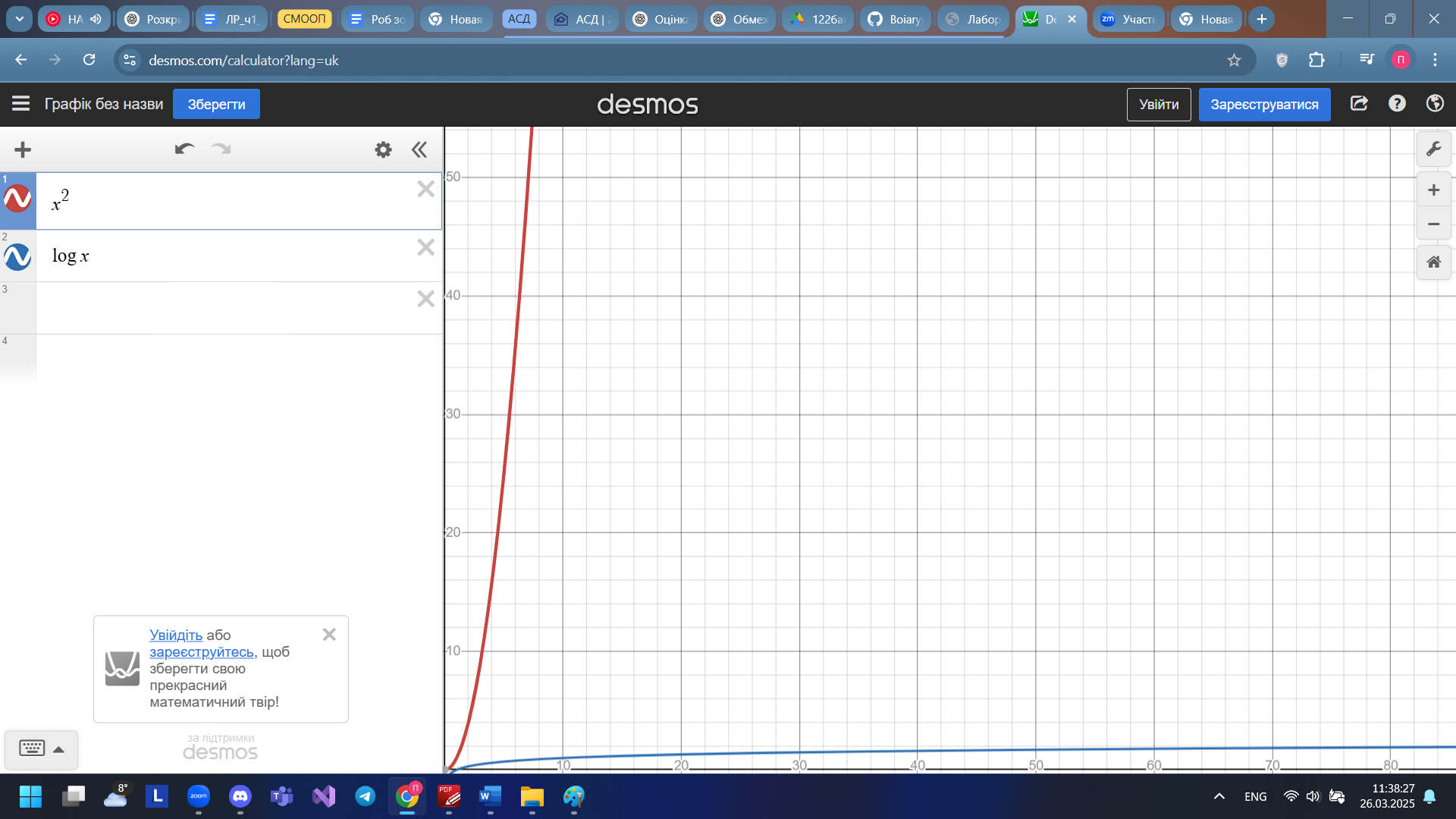


Рисунок 1 – Поведінка функцій

**Задача 8:** Довести, що .

Розв’язання:

Згідно з визначенням O-нотації, потрібно знайти такі константи 𝑐 і 𝑛₀, що виконується: для .

Маємо:

, для .

.

Отже, можна взяти 𝑐 = 15, 𝑛₀ = 1. Це доводить, що .

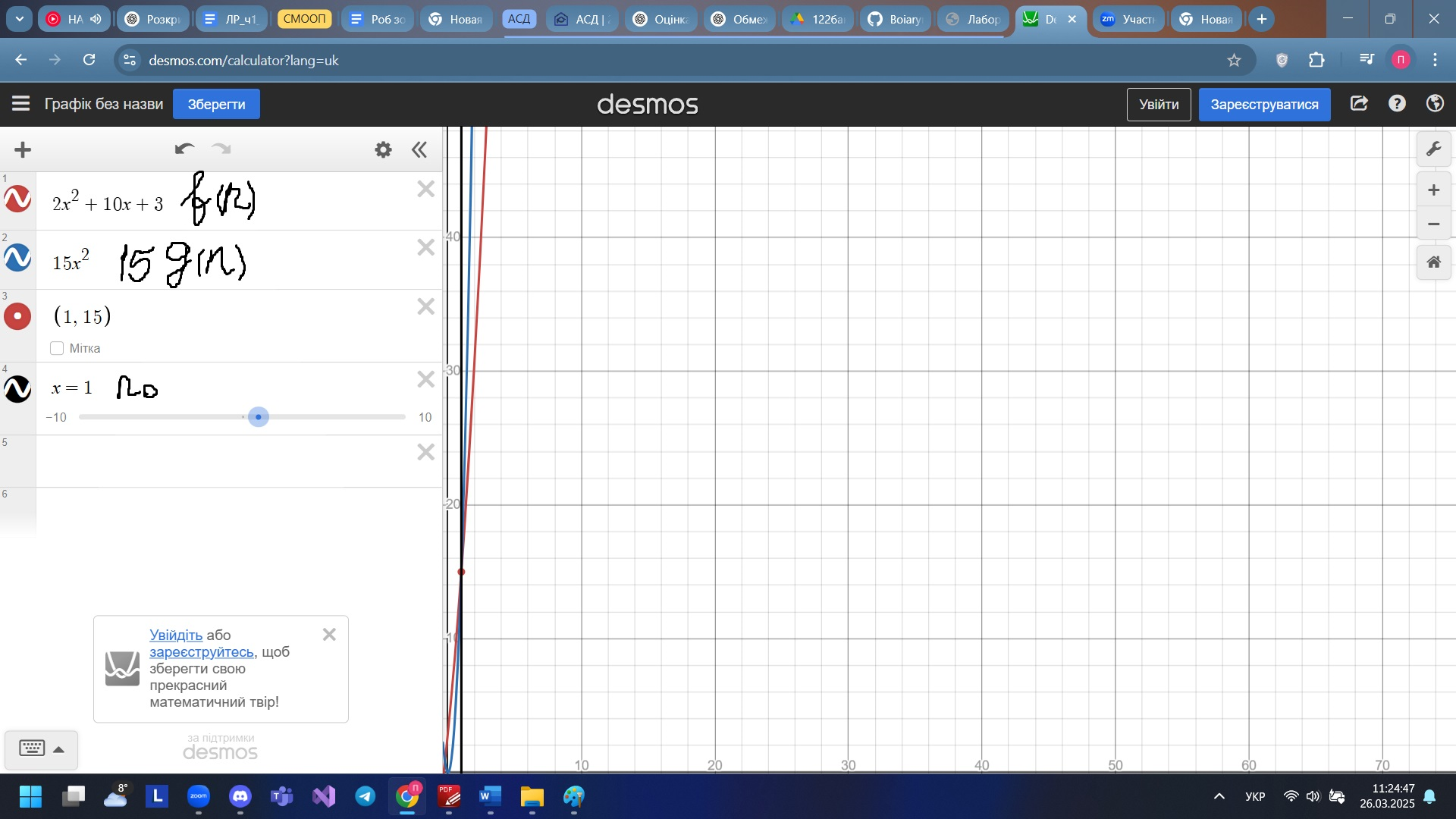


Рисунок 2 – Графік до задачі 8

## Отримані результати.

Визначено асимптотичну складність заданих функцій.

Доведено правильність обчислень згідно з визначенням O-нотації.

## Відповіді на контрольні питання.

1. Що таке асимптотична складність алгоритму?

Асимптотична складність алгоритму – це міра зростання часу виконання або використання пам’яті алгоритмом при збільшенні вхідних даних.

1. Яким чином визначається O-нотація і яка її сутність?

O-нотація використовується для оцінки верхньої межі зростання функції. Вона визначається як множина функцій, які не зростають швидше за певну граничну функцію з точністю до константи.

1. Які основні правила використання O-нотації при аналізі алгоритмів?

Залишаємо тільки найшвидше зростаючий член.

Ігноруємо константні множники.

Використовуємо визначення O-нотації для формального доведення.

1. Що означають вирази O(1), O(n), O(n²) в контексті асимптотичної складності?

O(1) – алгоритм виконується за сталий час, незалежно від розміру вхідних даних.

O(n) – час виконання алгоритму зростає лінійно зі збільшенням розміру вхідних даних.

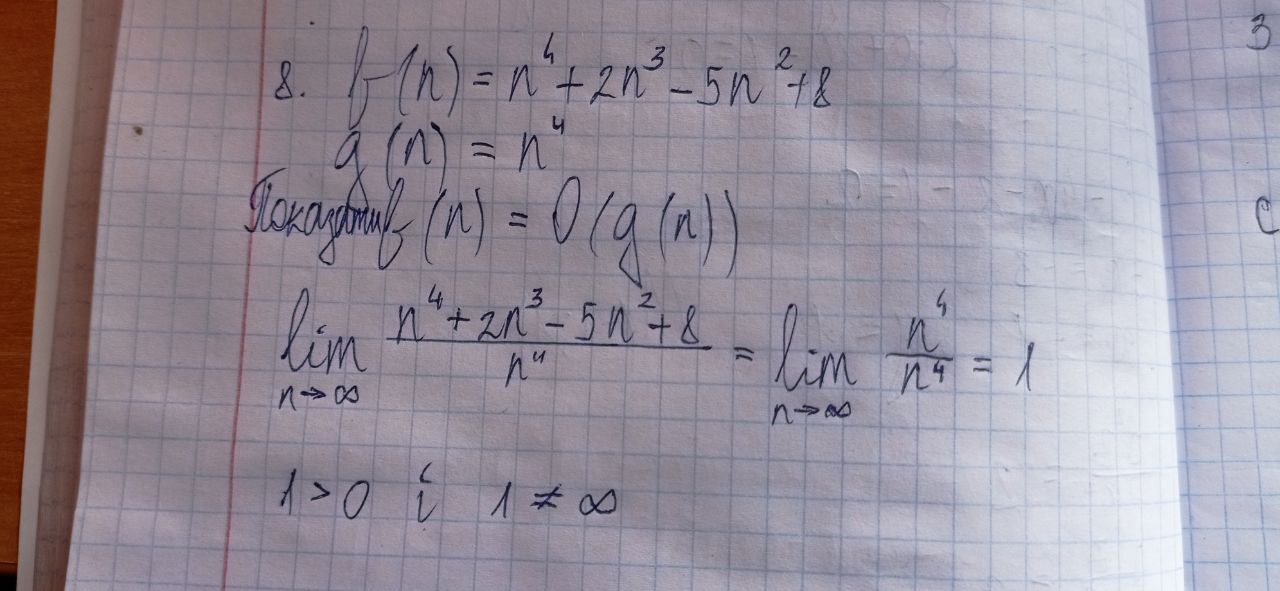
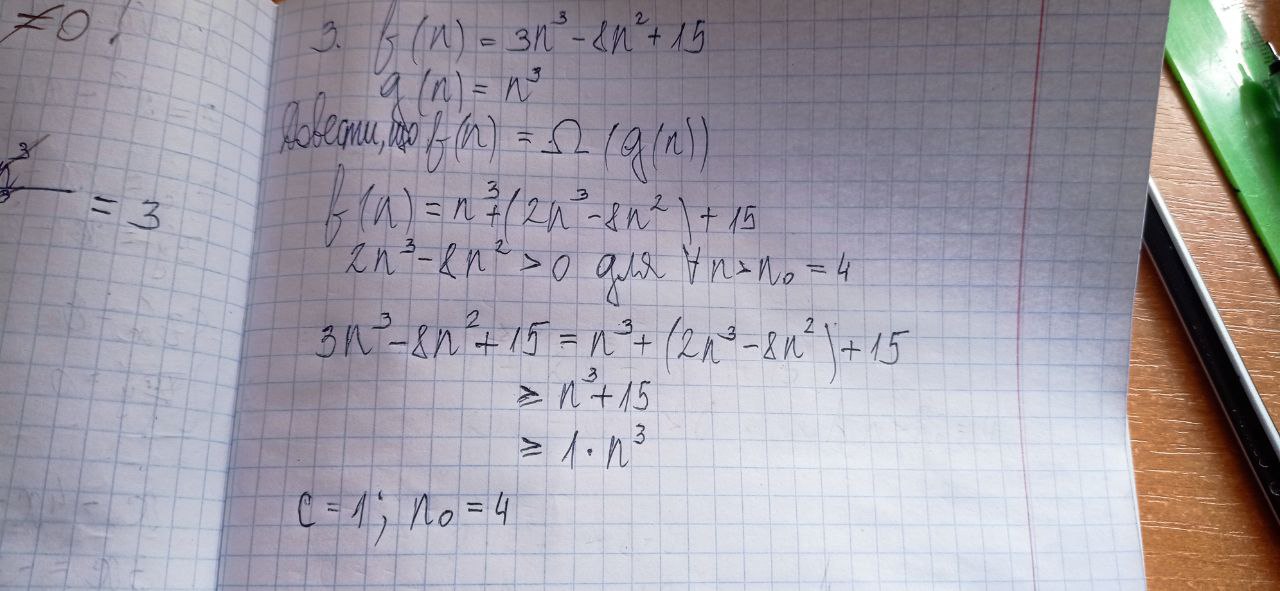
O(n²) – час виконання алгоритму зростає квадратично.

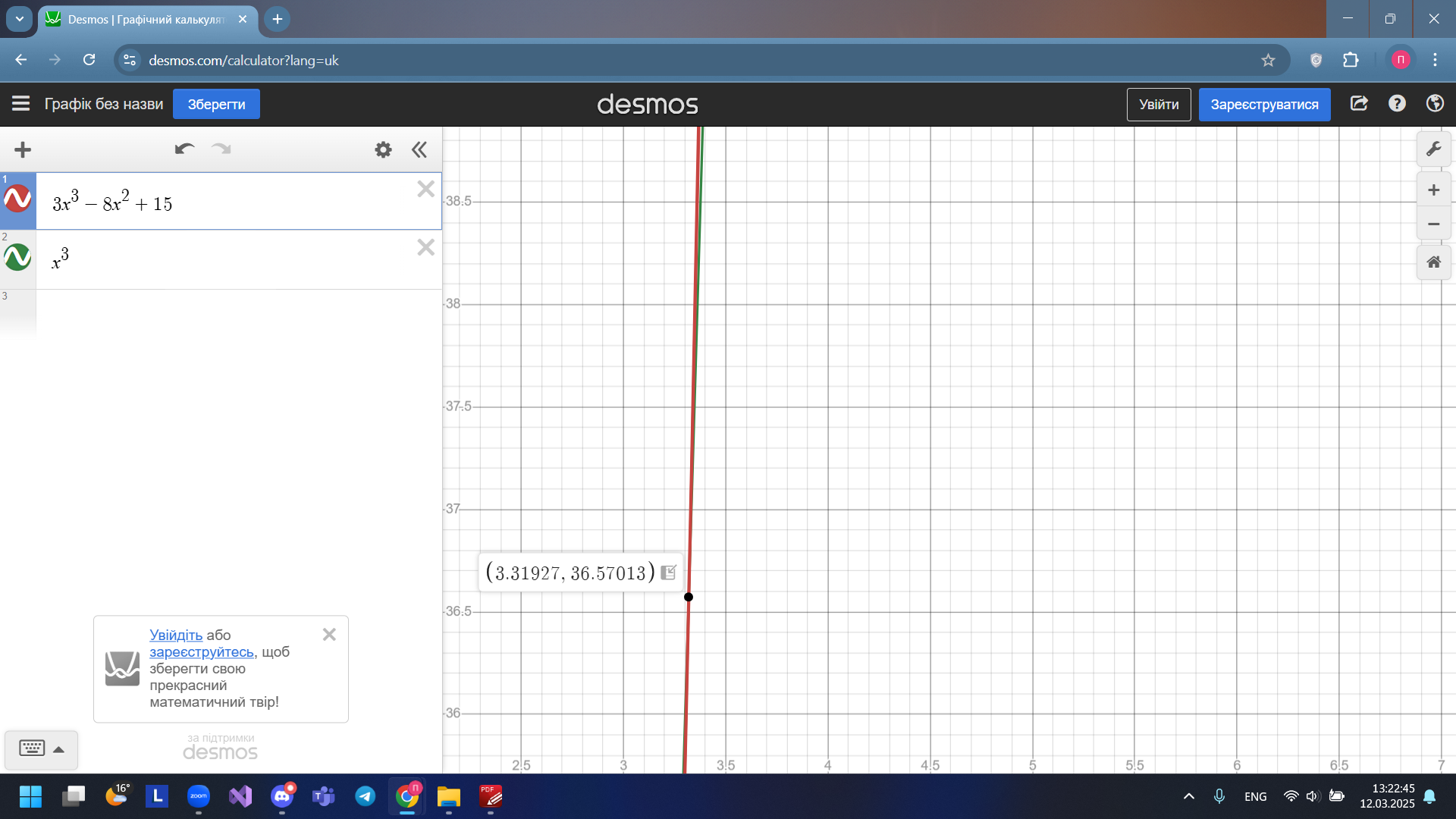
1. Яким чином визначити асимптотичну складність алгоритму за його кодом або математичним виразом?

Аналізувати вкладені цикли та рекурсію.

Використовувати правила спрощення O-нотації.

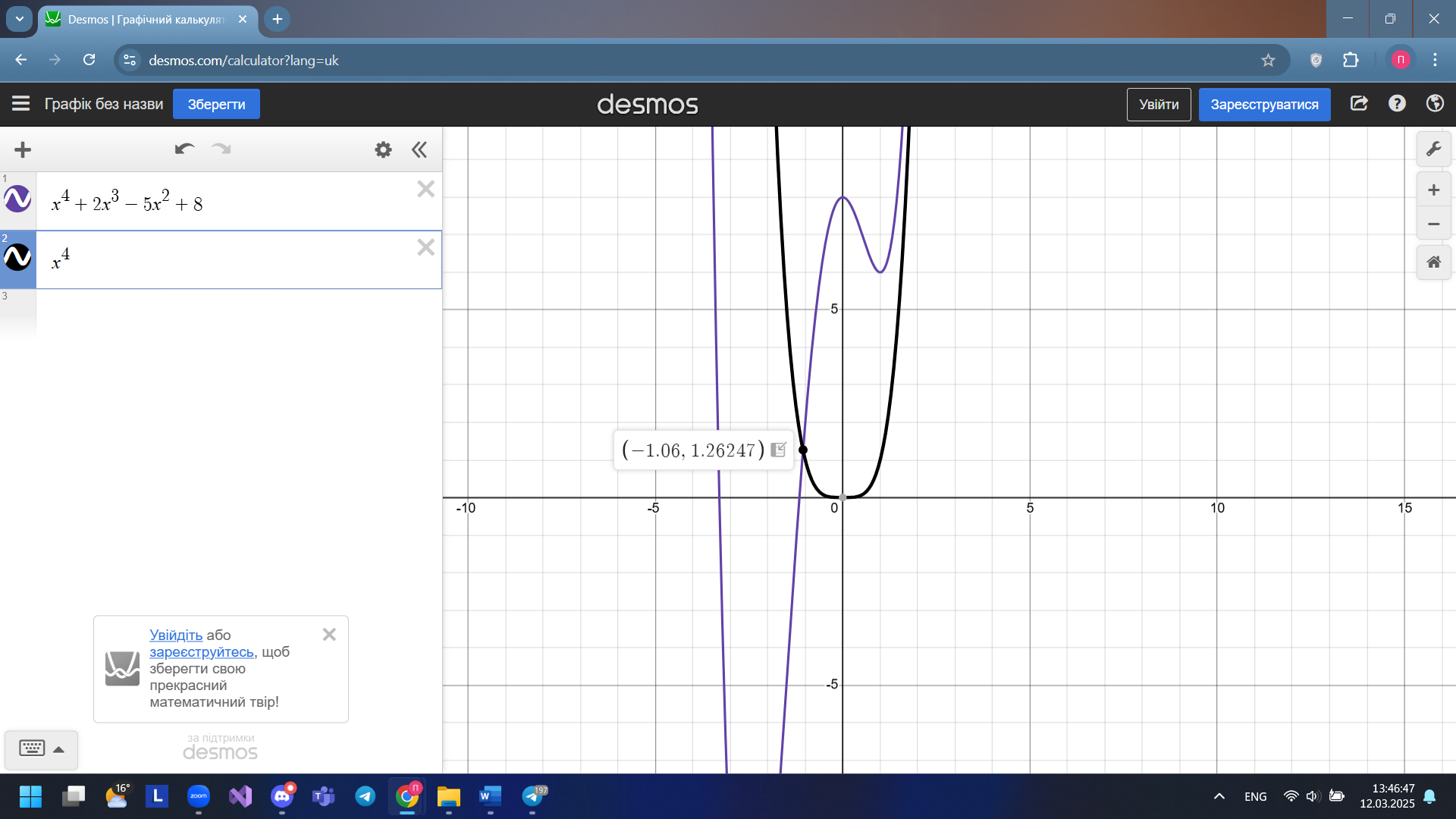
Виділяти найшвидше зростаючий член у виразі складності.





f(n)

c(g(n))



n

f(n)

c(g(n))