

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
КРЕМЕНЧУЦЬКИЙ НАЦІОНАЛЬНИЙ
УНІВЕРСИТЕТ ІМЕНІ МИХАЙЛА
ОСТРОГРАДСЬКОГО

Кафедра комп'ютерної інженерії та електроніки

ЗВІТ З ПРАКТИЧНИХ РОБІТ

з навчальної дисципліни

«Алгоритми та методи обчислень»

Тема «Асимптотична складність алгоритмів. Інші нотації»

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

Викладач Сидоренко В. М.

Тема роботи: Асимптотична складність алгоритмів. Інші нотації

1.1 Постановка завдання

Мета роботи: набути практичних навичок у розв'язанні задач на оцінку асимптотичної складності алгоритмів у Ω , Θ , o , ω -нотаціях.

Завдання: розв'язати дві задачі, визначені за особистим номером у списку групи.

1.2 Розв'язання задачі

Завдання 3

Дано функції: $f(n) = 3n^3 - 8n^2 + 15$ та $g(n) = n^3$. Потрібно довести, що $f(n) = \Omega(g(n))$.

Розв'язання (рис. 1):

3. $f(n) = 3n^3 - 8n^2 + 15$
 $g(n) = n^3$
Довести, що $f(n) = \Omega(g(n))$
 $f(n) = n^3 + (2n^3 - 8n^2) + 15$
 $2n^3 - 8n^2 > 0$ для $\forall n > n_0 = 4$
 $3n^3 - 8n^2 + 15 = n^3 + (2n^3 - 8n^2) + 15$
 $\geq n^3 + 15$
 $\geq 1 \cdot n^3$
 $C = 1; n_0 = 4$

Рисунок 1 –Розв'язання завдання 3

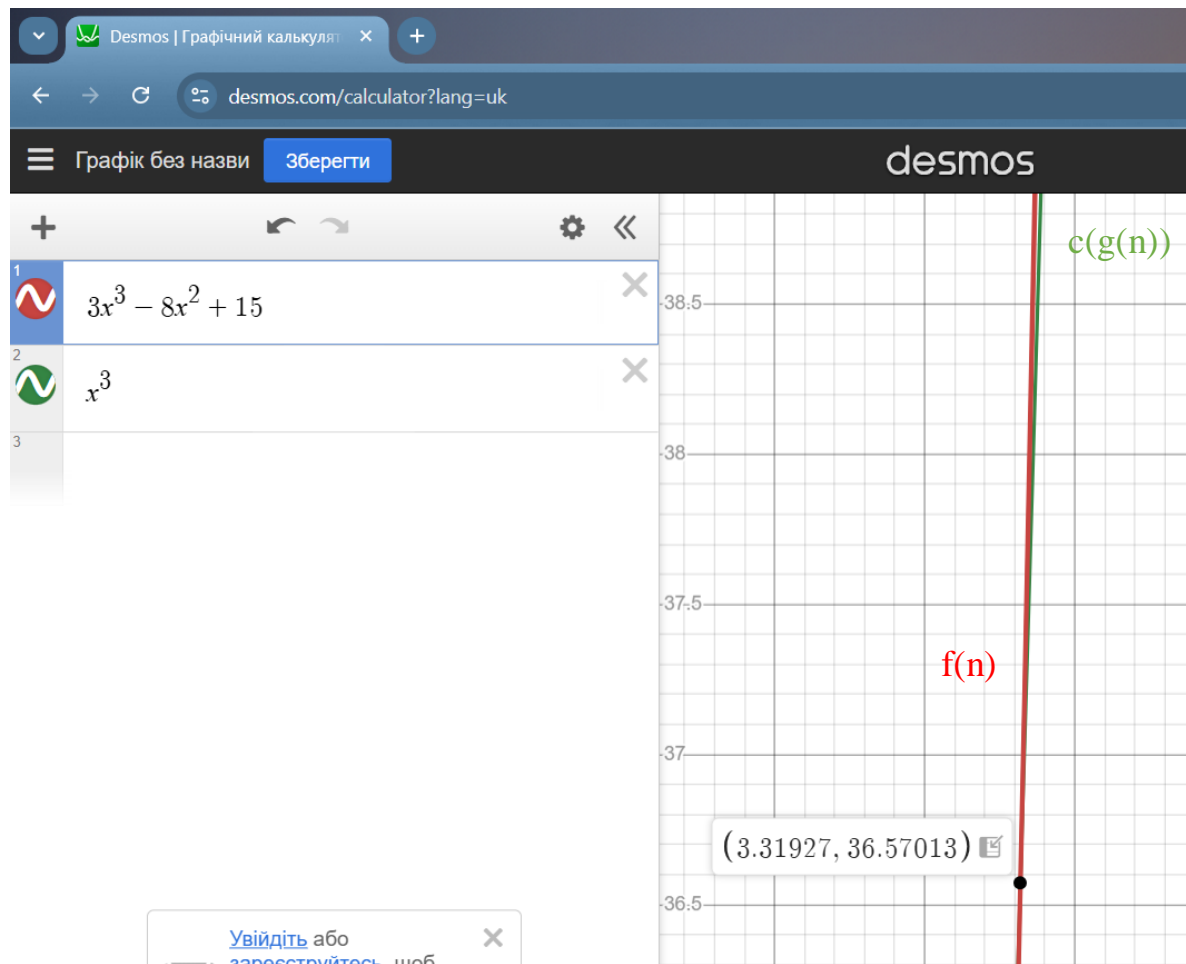


Рисунок 2 – Графік до завдання 3

Завдання 8

Дано функції: $f(n) = n^4 + 2n^3 - 5n^2 + 8$ та $g(n) = n^4$. Потрібно довести, що $f(n) = O(g(n))$, використовуючи метод меж.

Розв'язання (рис. 3):

$$\begin{aligned}
 & 8. \quad f(n) = n^4 + 2n^3 - 5n^2 + 8 \\
 & \quad g(n) = n^4 \\
 & \text{Показати } f(n) = O(g(n)) \\
 & \lim_{n \rightarrow \infty} \frac{n^4 + 2n^3 - 5n^2 + 8}{n^4} = \lim_{n \rightarrow \infty} \frac{n^4}{n^4} = 1 \\
 & 1 > 0 \quad \text{і} \quad 1 \neq \infty
 \end{aligned}$$

Рисунок 3 – Розв'язання завдання 8

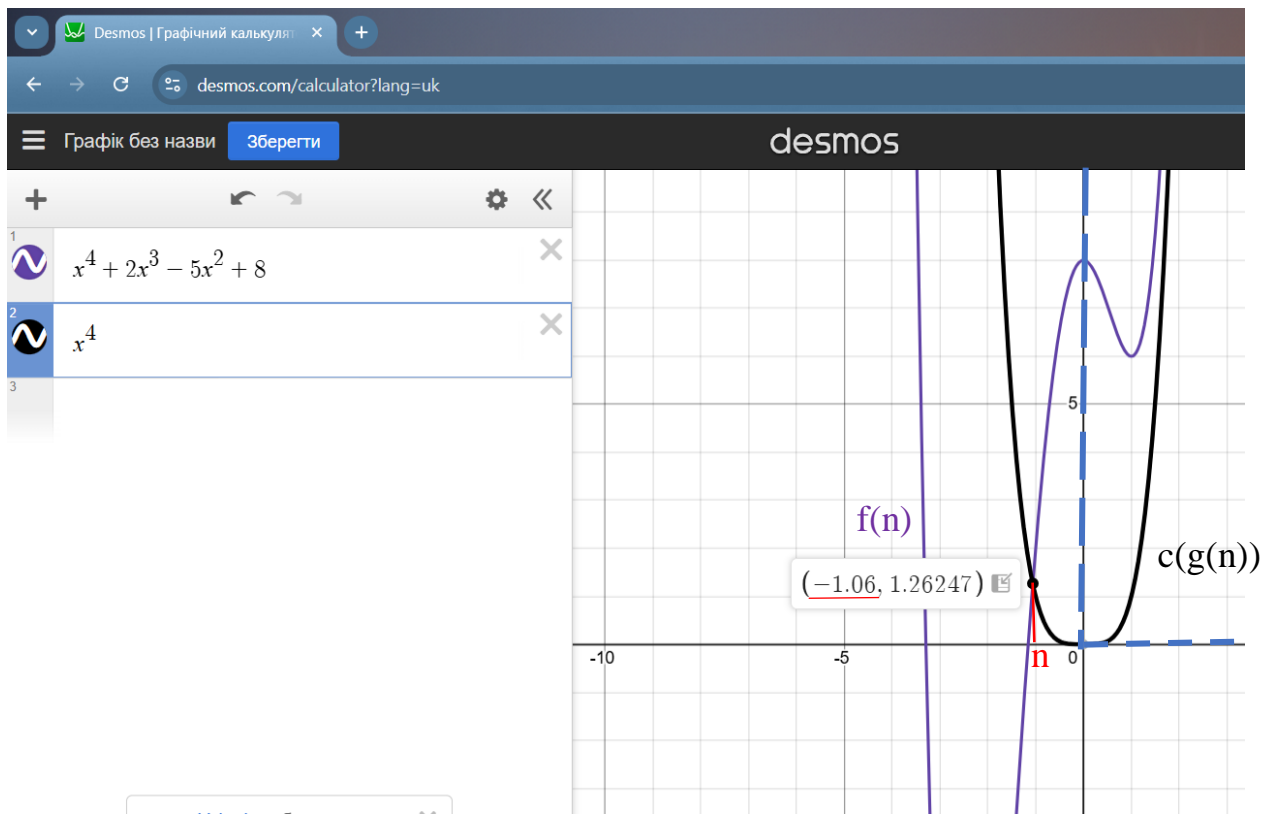


Рисунок 4 – Графік до завдання 8

1.3 Отримані результати

Доведено, що $f(n) = \Omega(g(n))$ у завданні 3.

Доведено, що $f(n) = O(g(n))$ у завданні 8, використовуючи метод меж.

1.4 Відповіді на контрольні питання

1. Що таке асимптотична складність алгоритму?

Асимптотична складність алгоритму – це характеристика швидкості його виконання залежно від розміру вхідних даних у граничному випадку (наприклад, при $n \rightarrow \infty$).

2. Які інші нотації, крім O-нотації, використовуються для вираження асимптотичної складності?

Крім O-нотації, використовують нотації Ω (велика омега), Θ (тета), o (маленьке o), ω (маленька омега).

3. Як визначити асимптотичну складність алгоритму за допомогою символів Θ і Ω ?

$\Omega(n)$ вказує, що функція має нижню межу асимптотичної складності.

$\Theta(n)$ означає, що функція має як верхню, так і нижню межу, тобто точну асимптотичну складність.

4. Яка різниця між O -нотацією, Θ -нотацією і Ω -нотацією?

$O(n)$ вказує на верхню межу складності.

$\Omega(n)$ вказує на нижню межу.

$\Theta(n)$ означає, що функція точно знаходиться в цих межах.

5. Які основні властивості інших нотацій, таких як o (маленьке o) та ω (маленька ω)?

$o(n)$ означає, що функція зростає повільніше за іншу (жорсткіше обмеження, ніж $O(n)$).

$\omega(n)$ означає, що функція зростає швидше за іншу (жорсткіше обмеження, ніж $\Omega(n)$).