МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

КРЕМЕНЧУЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

ІМЕНІ МИХАЙЛА ОСТРОГРАДСЬКОГО

Кафедра комп’ютерної інженерії та електроніки

Виконала: Групи

Пугачова Д. В. КІ-23-1

Кременчук 2024

ПРАКТИЧНА РОБОТА №2

Тема. Асимптотична складність алгоритмів. Інші нотації

Мета: набути практичних навичок у розв’язанні задач на оцінку асимптотичної складності алгоритмів у Ω, Θ, 𝜊, 𝜃, 𝜔-нотаціях.

ЗАВДАННЯ:

№5

**𝑓(𝑛) = 𝛺(𝑔(𝑛))**

**c=4,**

**n0​=3.**

**Ω-Oмега-нотації виконується.**

№15

𝑓(𝑛) = 𝑛 4 − 3𝑛 2 + 4𝑛 + 9 𝑔(𝑛) = 𝑛4 𝑓(𝑛) = 𝑂(𝑔(𝑛))

==1-++=1

n

n

f(n)=

отже, 𝑓(𝑛) = 𝑂(𝑔(𝑛))

**Контрольні питання**

**1. Що таке асимптотична складність алгоритму?**

Асимптоти́чна складність алгоритму — це характеристика, яка описує, як змінюється час виконання або обсяг пам'яті алгоритму в залежності від розміру вхідних даних при великих значеннях цього розміру. Вона використовується для оцінки ефективності алгоритму в умовах великих вхідних даних і дозволяє порівнювати алгоритми за їхньою ефективністю в довгостроковій перспективі.

Основною метою асимптотичної складності є усунення впливу констант і незначних чинників, щоб зосередитися на основних аспектах алгоритму, таких як його масштабованість.

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

*Θ-нотація: точна оцінка зростання.*

*Ω-нотація: нижня межа зростання.*

*o-нотація: строга верхня межа.*

*ω-нотація: строга нижня межа.*

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

***Θ-нотація****: Щоб визначити асимптотичну складність за допомогою Θ-нотації, потрібно оцінити алгоритм з обох боків — як верхню, так і нижню межу його часу виконання. Якщо алгоритм виконується в межах деякої константи, то його складність можна точно виразити через Θ-нотацію.*

***Ω-нотація****: Для визначення нижньої межі складності алгоритму, потрібно оцінити його швидкість виконання у найгіршому випадку. Якщо алгоритм має обмеження на час виконання (наприклад, мінімальний час, який він потребує для виконання певної операції), то можна застосувати Ω-нотацію.*

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

**O-нотація (Велике O)** описує **верхню межу** складності. Вона використовується для того, щоб показати, що час виконання алгоритму не буде більшим за певну функцію

**Θ-нотація (Тета-нотація)** дає **точну оцінку** складності, показуючи, що час виконання алгоритму буде зростати в межах двох функцій для великих значень n. Тобто, це як верхня, так і нижня межа

**Ω-нотація (Омега-нотація)** описує **нижню межу** складності. Вона використовується для визначення мінімального часу, необхідного для виконання алгоритму. Тобто, це мінімальний час, який алгоритм обов'язково витрачає на виконання задачі

**5. Які основні властивості інших нотацій, таких як o (маленька о), ω (маленька омега) та 𝑜 (маленька о з верхнім індексом)?**

***o-нотація (маленька о)*** *— використовується для позначення строго меншої асимптотичної верхньої межі. Якщо f(n)=o(g(n))f(n) = o(g(n))f(n)=o(g(n)), це означає, що функція f(n)f(n)f(n) зростає значно повільніше, ніж g(n)g(n)g(n), і при великих значеннях nnn функція f(n)f(n)f(n) стає майже незначною порівняно з g(n)g(n)g(n).*

***ω-нотація (маленька омега) —*** *описує строго більшу асимптотичну нижню межу. Якщо f(n)=ω(g(n))f(n) = \omega(g(n))f(n)=ω(g(n)), це означає, що функція f(n)f(n)f(n) зростає швидше за g(n)g(n)g(n), і для великих значень nnn функція g(n)g(n)g(n) стає незначною порівняно з f(n)f(n)f(n).*

***o-нотація\**** *— рідше використовується і зазвичай застосовується для позначення асимптотичної поведінки функцій без урахування певних множників. Вона часто використовується в теорії чисел або для опису складності алгоритмів, коли потрібно враховувати тільки основні аспекти зростання функцій без додаткових поліноміальних множників.*