

АЛГОРИТМІЧНА СКЛАДНІСТЬ ДИСКРЕТНИХ ЗАДАЧ

Вінницький національний технічний університет

Анотація

У роботі розглянуто поняття алгоритмічної складності як ключового інструменту аналізу дискретних задач. Показано роль часової та просторової складності алгоритмів у теорії алгоритмів і комп'ютерних науках. Розглянуто основні класи складності та наведено приклади їх використання при розв'язанні дискретних задач.

Ключові слова

алгоритмічна складність, дискретна математика, алгоритми, обчислювальна складність, Big O.

Abstract

The paper considers the concept of algorithmic complexity as a key tool for analyzing discrete problems. The role of time and space complexity of algorithms in algorithm theory and computer science is shown. The main complexity classes are considered and examples of their application in solving discrete problems are presented.

Keywords

algorithmic complexity, discrete mathematics, algorithms, computational complexity, Big O.

Вступ

У дискретній математиці значну роль відіграє аналіз ефективності алгоритмів, які використовуються для розв'язання задач. Алгоритмічна складність дозволяє оцінити ресурси, необхідні для виконання алгоритму, та порівнювати різні підходи до розв'язання однієї і тієї ж задачі.

Результати дослідження

Алгоритмічна складність є важливою характеристикою дискретних задач, оскільки вона дає змогу оцінити ефективність алгоритмів, що застосовуються для їх розв'язання у сфері інформатики, теорії графів, комбінаторики та криптографії [1]. Зі збільшенням обсягу вхідних даних вибір алгоритму з прийнятною складністю безпосередньо впливає на швидкодію та практичну реалізованість обчислень [2].

Під алгоритмічною складністю розуміють залежність витрат обчислювальних ресурсів від розміру вхідних даних, насамперед часу виконання та обсягу використаної пам'яті [1]. Для порівняння алгоритмів застосовується асимптотичний аналіз, який дозволяє оцінити їх поведінку при великих обсягах даних незалежно від конкретної реалізації та апаратного забезпечення [3].

У межах асимптотичного аналізу широко використовуються позначення $O()$, $\Omega()$ та $\Theta()$, які відповідно описують верхню, нижню та точну оцінки складності алгоритмів. Застосування цих позначень дає змогу формалізувати процес порівняння алгоритмів і визначити їх ефективність у найгіршому, найкращому або середньому випадках.

У дискретних задачах складність алгоритмів може зростати з різною швидкістю — від константної, логарифмічної та лінійної до поліноміальної й експоненційної [2]. Алгоритми з експоненційною складністю, як правило, є непридатними для практичного використання при великих обсягах даних, що зумовлює необхідність пошуку ефективніших підходів або обмеження розмірів задач.

Особливе місце займають NP-повні задачі, для яких не відомі алгоритми з поліноміальною складністю. До таких задач належать, зокрема, задача комівояжера, задача розфарбування графів та задача покриття множин. У практичних застосуваннях для їх розв'язання використовуються наближені, евристичні або ймовірнісні алгоритми, які забезпечують прийнятний результат за обмежений час [4].

Типовими прикладами дискретних задач є задачі сортування, пошуку та аналізу графів, для яких існує декілька алгоритмів із різною алгоритмічною складністю [3]. Наприклад, порівняння алгоритмів сортування дозволяє обрати оптимальний метод залежно від розміру та структури вхідних даних, а аналіз графових алгоритмів — забезпечити ефективну обробку мережових та логістичних задач.

Таким чином, аналіз алгоритмічної складності дискретних задач є необхідним для оцінки ефективності алгоритмів і прогнозування їх роботи при зростанні обсягу даних [4]. Розуміння принципів оцінки складності сприяє розробці продуктивних і надійних програмних систем та є важливою складовою підготовки фахівців у галузі комп'ютерних наук [2].

Висновки

Алгоритмічна складність є важливим поняттям дискретної математики та теорії алгоритмів. Вона дозволяє оцінювати ефективність алгоритмів, порівнювати різні методи розв'язання задач та обирати оптимальні рішення для практичних застосувань у комп'ютерних системах.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Кормен Т. Алгоритми: побудова та аналіз / Пер. з англ. – К.: Видавництво «Вільна думка», 2012.
2. Гросс Д., Манкузе П. Дискретна математика та алгоритми – М.: ДМК Пресс, 2015.
3. Лафоре Р. Структури даних і алгоритми в C++ – К.: BHV, 2013.
4. Вихристюк В. Комп'ютерні алгоритми та оцінка їх складності – Львів: Видавництво ЛНУ, 2018.

Червінський Владислав Олегович — студент групи 1KI-25МС, факультет інформаційних технологій та комп'ютерної інженерії, Вінницький національний технічний університет, Вінниця, e-mail: ychervinsky2006@gmail.com

Науковий керівник: Добровольська Наталя Вікторівна — канд. технічних наук, доцент, Вінницький національний технічний університет, м. Вінниця

Chervinskyi Vladyslav Olehovych — student of group 1KI-25MS, Faculty of Information Technology and Computer Engineering, Vinnytsia National Technical University, Vinnytsia, e-mail: ychervinsky2006@gmail.com

Scientific supervisor: **Dobrovol'ska Nataliia Viktorivna** — Candidate of Technical Sciences, Associate Professor, Vinnytsia National Technical University, Vinnytsia.