

## ДОСЛІДЖЕННЯ ШВИДКОСТІ РОБОТИ РЕКУРЕНТНОГО МЕТОДУ ГЕНЕРУВАННЯ РОЗБИТТЯ МНОЖИН

Вінницький національний технічний університет;

### *Анотація*

У цій роботі досліджується залежність часу роботи рекурентного методу генерації розбиття множин, який є важливим для оптимізації комбінаторних задач, від потужності множини та кількості компонент розбиття. Методика базується на розподілі множини на підмножини із застосуванням рекурсивних викликів, що дозволяє створювати всі можливі розбиття. У ході дослідження було поставлено, що обчислювальна складність алгоритму зростає по експоненті, зі збільшенням розміру множини та кількості комірок для розкладу цієї множини. Це зумовлює суттєве збільшення часу виконання, що впливає на ефективність при роботі із задачами великого масштабу, які потребують швидкості та продуктивності.

**Ключові слова:** генерація розбиття множин, множини, рекурентний метод.

### *Abstract*

This paper examines the dependency of the runtime of a recursive method for generating set partitions, which is essential for optimizing combinatorial tasks, on the cardinality of the set and the number of partition components. The methodology is based on dividing a set into subsets through recursive calls, enabling the creation of all possible partitions. The study found that the computational complexity of the algorithm grows exponentially as the size of the set and the number of cells for its partition increase. This leads to a significant increase in execution time, which affects efficiency when dealing with large-scale tasks that require high speed and productivity.

**Keywords:** set partition generation, sets, recursive method.

### Вступ

У сучасних задачах комбінаторної оптимізації та дискретної математики проблема розбиття множин відіграє значну роль в розробці алгоритмів для технічних систем. Зокрема, дослідження, які націлені на пошук більш досконалих алгоритмів, що будуть здатні обробляти велику кількість даних за мінімальний часовий проміжок у великих обчислювальних системах [1]. Одним із таких методів є рекурентний метод генерації розбиття множин, що використовує досить простий механізм, за допомогою якого зменшується обчислювальна складність [2].

Метою цієї роботи є дослідити швидкість роботи рекурентного методу генерації множин, проаналізувавши його основні переваги, недоліки та напрями для оптимізації. Це дасть розуміння придатності методу для використання у різних практичних сценаріях та допоможе вдосконалити методи розв'язання задач, що потребують високої продуктивності та ефективності.

### Результати дослідження

Для початку описується алгоритм виконання методу рекурентного генерування розбиття числа. Розбиття числа  $n$  на  $k$  компонент можна продовжувати в зростаючому лексикографічному порядку [1,2], починаючи з розбиття, в якому  $l_1 = 1, l_2 = 2, \dots, l_{i-1} = n - k + 1$  і продовжуючи процес таким чином. Для отримання наступного розбиття з даного продивляємося елементи даного розбиття

в порядку з права наліво, знаходячи найправішу компоненту  $b_i$  таку, що  $l_i - l_k \geq 2$ . Далі замінюємо всі компоненти  $l_j (j = i, k - 1)$  на значення  $l_i + 1$  і остання компонента  $l_k$  тоді приймає значення:  $n - \sum_{j=1}^{k-1} l_j$ . Процедура побудови розбиттів закінчується, коли ні один з елементів розбиття не відрізняється від останнього більше ніж на одиницю.

Розглянемо результати роботи алгоритму, задавши стале значення кількості компонент 15 ( $n=15$ ), у той час як потужність збільшуватиметься із кроком 1. Для першого експерименту було отримано наступний графік:

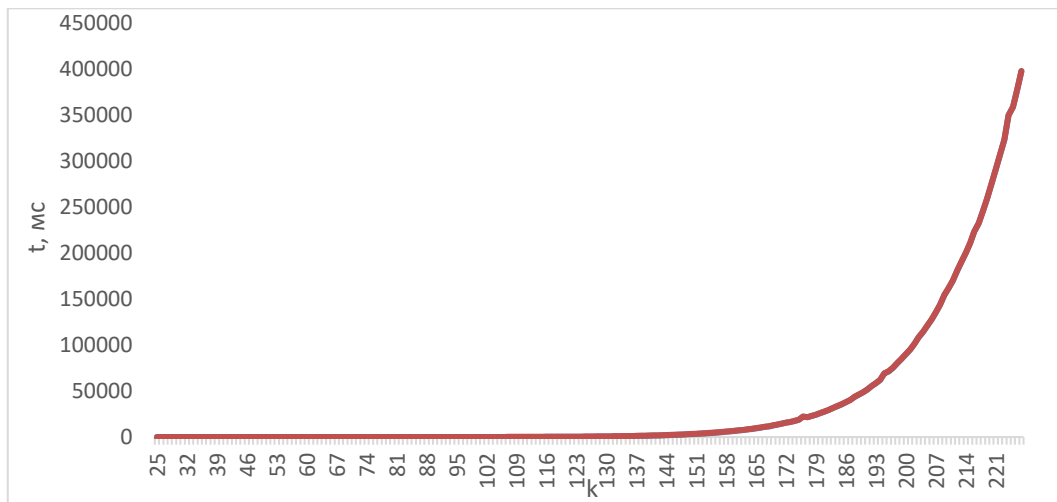


Рис. Графік залежності часу роботи алгоритму від потужності множини, при 15 компонентах (1)

Для другого експерименту змінимо стале значення кількості компонент на 25 ( $n=25$ ). Потужність множини, як і у першому експерименті, збільшуватиметься із кроком 1. Отримано наступний результат:

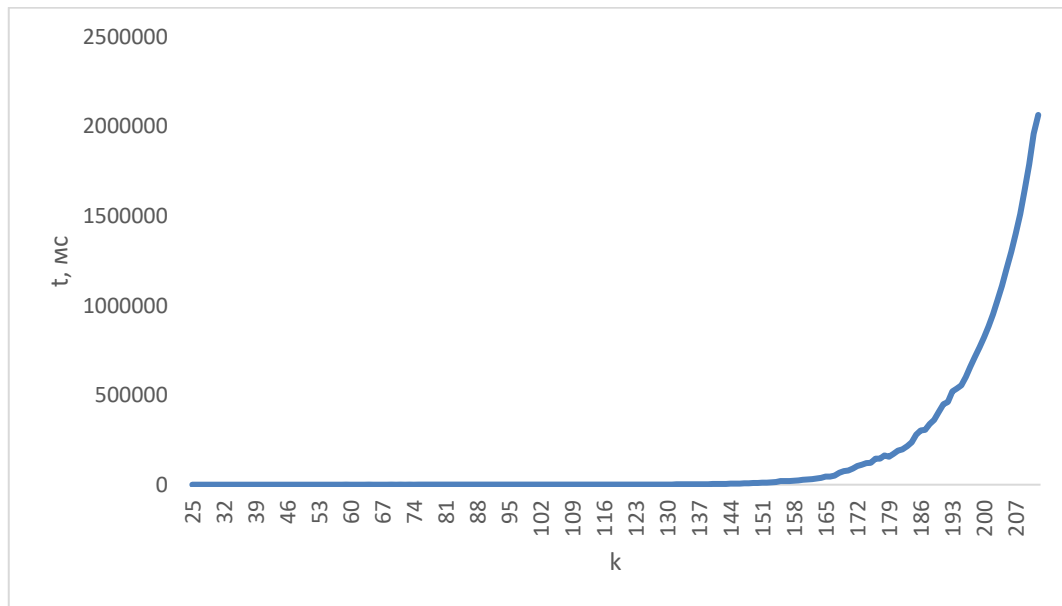


Рис. Графік залежності часу роботи алгоритму від потужності множини, при 25 компонентах (2)

### Висновки

Результати дослідження показали, що час роботи рекурентного методу генерації розбиття множин збільшується експоненціально зі збільшенням потужності множини. Також відповідно до результатів дослідження, кількість компонент також значно впливає швидкість роботи алгоритму. Це свідчить

про те, що при роботі із потужними множинами, алгоритм має високу обчислювальну складність, що може бути критичним для розв’язання певних задач, що задають обмеження у часі.

#### СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Бондаренко М.Ф. Комп’ютерна дискретна математика – Харків, “Кампанія СМІТ”, 2004. 480с.
2. Капітонова Ю.В. Основи дискретної математики – Київ, наукова думка, 2002. 573с.

**Буняк Богдан Ю.** — студент факультету інформаційних технологій та комп’ютерної інженерії, Вінницький національний технічний університет, Вінниця, e-mail: [buniak.bohdan2016@gmail.com](mailto:buniak.bohdan2016@gmail.com)

Науковий керівник: **Кондратенко Наталія Р.** – кандидат технічних наук, професор каф. Захисту інформації, Вінницький національний технічний університет, м. Вінниця, e-mail: [kondratenko.natalia@vntu.edu.ua](mailto:kondratenko.natalia@vntu.edu.ua)

**Bunyak Bohdan Y.** — Department of Information Technologies and Computer Engineering, Vinnytsia National Technical University, Vinnytsia, email : [buniak.bohdan2016@gmail.com](mailto:buniak.bohdan2016@gmail.com)

Supervisor: **Kondratenko Natalia R.** — Candidate of Technical Sciences, Professor of the Department of Information Security, Vinnytsia National Technical University, Vinnytsia, [kondratenko.natalia@vntu.edu.ua](mailto:kondratenko.natalia@vntu.edu.ua)