Міністерство науки і освіти України

ВІННИЦЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ

Кафедра обчислювальної техніки

**Системне програмне забезпечення**

**Лабораторна робота № 4**

Виконав: Свіріпа С.М.

Перевірив: Семеренко В.П.

Вінниця 2019

Сортування вибором — простий алгоритм сортування лінійного [масиву](https://uk.wikipedia.org/wiki/%D0%9C%D0%B0%D1%81%D0%B8%D0%B2_(%D1%81%D1%82%D1%80%D1%83%D0%BA%D1%82%D1%83%D1%80%D0%B0_%D0%B4%D0%B0%D0%BD%D0%B8%D1%85)" \o "Масив (структура даних)), на [основі вставок](https://uk.wikipedia.org/wiki/%D0%A1%D0%BE%D1%80%D1%82%D1%83%D0%B2%D0%B0%D0%BD%D0%BD%D1%8F_%D0%BD%D0%B0_%D0%BE%D1%81%D0%BD%D0%BE%D0%B2%D1%96_%D0%BF%D0%BE%D1%80%D1%96%D0%B2%D0%BD%D1%8F%D0%BD%D1%8C" \o "Сортування на основі порівнянь). Має ефективність n2, що робить його неефективним при сортування великих масивів, і в цілому, менш ефективним за подібний алгоритм [сортування включенням](https://uk.wikipedia.org/wiki/%D0%A1%D0%BE%D1%80%D1%82%D1%83%D0%B2%D0%B0%D0%BD%D0%BD%D1%8F_%D0%B2%D0%BA%D0%BB%D1%8E%D1%87%D0%B5%D0%BD%D0%BD%D1%8F%D0%BC" \o "Сортування включенням). Сортування вибором вирізняється більшою простотою, ніж [сортування включенням](https://uk.wikipedia.org/wiki/%D0%A1%D0%BE%D1%80%D1%82%D1%83%D0%B2%D0%B0%D0%BD%D0%BD%D1%8F_%D0%B2%D0%BA%D0%BB%D1%8E%D1%87%D0%B5%D0%BD%D0%BD%D1%8F%D0%BC" \o "Сортування включенням), і в деяких випадках, вищою продуктивністю.

Алгоритм працює таким чином:

1. Знаходить у списку найменше значення
2. Міняє його місцями із першим значеннями у списку
3. Повторює два попередніх кроки, доки список не завершиться (починаючи з наступної позиції)

Фактично, таким чином ми поділили список на дві частини: перша (ліва) — повністю відсортована, а друга (права) — ні.

Ось приклад сортування масиву з п'яти елементів за даним алгоритмом:

64 25 12 22 11

11 25 12 22 64

11 12 25 22 64

11 12 22 25 64

Сортування вставками також може працювати зі списками, які підтримують операції додавання і видалення, як то [зв'язаний список](https://uk.wikipedia.org/wiki/%D0%97%D0%B2%27%D1%8F%D0%B7%D0%B0%D0%BD%D0%B8%D0%B9_%D1%81%D0%BF%D0%B8%D1%81%D0%BE%D0%BA" \o "Зв'язаний список). У такому разі, більш зручно видаляти зі списку найменший елемент, і вставляти його в кінець відсортованої частини масиву. Наприклад:

64 25 12 22 11

11 64 25 12 22

11 12 64 25 22

11 12 22 64 25

11 12 22 25 64

Сортування вибором не є складним в аналізі та порівнянні його з іншими алгоритмами, оскільки жоден з циклів не залежить від даних у списку. Знаходження найменшого елементу вимагає перегляду усіх n елементів (у цьому випадку n − 1 порівняння), і після цього, перестановки його до першої позиції. Знаходження наступного найменшого елементу вимагає перегляду n − 1 елементів, і так далі, для (n − 1) + (n − 2) + … + 2 + 1 = n(n − 1) / 2 ∈ Θ(n2) порівнянь (дивіться [арифметична прогресія](https://uk.wikipedia.org/wiki/%D0%90%D1%80%D0%B8%D1%84%D0%BC%D0%B5%D1%82%D0%B8%D1%87%D0%BD%D0%B0_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B5%D1%81%D1%96%D1%8F" \o "Арифметична прогресія)). Кожне сканування вимагає однієї перестановки для n − 1 елементів (останній елемент знаходитиметься на своєму місці).

Двостороння сортування вибором :: Double selection sort

Схожа ідея використовується в шейкерні сортуванні, яка є варіантом бульбашкового сортування. Проходячи по невідсортоване частини масиву, ми крім максимуму також попутно знаходимо і мінімум. Мінімум ставимо на перше місце, максимум на останнє. Таким чином, несортованими частина при кожній ітерації зменшується відразу на два елементи.

На перший погляд здається, що це прискорює алгоритм в 2 рази - після кожного проходу невідсортоване подмассів зменшується не з одного, а відразу з двох сторін. Але при цьому в 2 рази збільшилася кількість порівнянь, а число свопів залишилося незмінним. Подвійний вибір лише незначно збільшує швидкість алгоритму, а на деяких мовах навіть чомусь працює повільніше.