МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

ВІДОКРЕМЛЕНИЙ СТРУКТУРНИЙ ПІДРОЗДІЛ  
«ТЕХНІЧНИЙ ФАХОВИЙ КОЛЕДЖ

ЛУЦЬКОГО НАЦІОНАЛЬНОГО ТЕХНІЧНОГО УНІВЕРСИТЕТУ»

Випускова циклова комісія Компʼютерних систем та інформаційних технологій

**Звіт**

**з практичної роботи №3**

з дисципліни “Алгоритми та структури даних”

Виконав: студент **2** курсу, групи **22-ICT**

Синиця Назар

Викладач: Володимир Остапюк

Здано:

Луцьк – 2024

**Зміст**

**Теорія та дослідження:**

1. Опишіть принцип роботи алгоритму сортування бульбашкою. Як відбувається порівняння та обмін елементів?
2. Опишіть принцип роботи алгоритму сортування вибором. Чим він відрізняється від сортування бульбашкою?
3. Яка часова складність алгоритмів сортування бульбашкою та сортування вибором в найгіршому, середньому та найкращому випадках?
4. В яких випадках доцільно використовувати сортування бульбашкою, а в яких - сортування вибором? Які переваги та недоліки кожного алгоритму?

**Практичні завдання (Python):**

1. Реалізуйте функцію bubble\_sort(array\_to\_sort), яка сортує список array\_to\_sort за допомогою алгоритму сортування бульбашкою.
2. Реалізуйте функцію selection\_sort(array\_to\_sort), яка сортує список array\_to\_sort за допомогою алгоритму сортування вибором.
3. Напишіть програму, яка зчитує список чисел з клавіатури та сортує його за допомогою обох алгоритмів. Виведіть відсортований список та кількість операцій порівняння та обміну, виконаних кожним алгоритмом.
4. Порівняйте ефективність алгоритмів сортування бульбашкою та сортування вибором на списках різної довжини (наприклад, 10, 1k, 100k елементів). Виміряйте час або кількість операцій виконання кожного алгоритму та \*побудуйте графік залежності часу(кількості операцій) виконання від розміру списку \*за допомогою Python.
5. \*[Додатково] Реалізуйте модифікацію алгоритму сортування бульбашкою, яка зупиняється, якщо під час проходу по списку не було зроблено жодного обміну.

**Висновок**

**Теорія та дослідження**

1. Опишіть принцип роботи алгоритму сортування бульбашкою. Як відбувається порівняння та обмін елементів?

Сортування бульбашкою працює за принципом багаторазового проходження по списку, під час якого порівнюються пари сусідніх елементів і, якщо вони розташовані у неправильному порядку, здійснюється їхній обмін.

**Порівняння:** перевірка, чи потрібно міняти місцями два елементи.

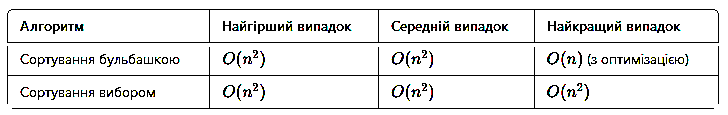
**Обмін:** перестановка двох елементів місцями.

1. Опишіть принцип роботи алгоритму сортування вибором. Чим він відрізняється від сортування бульбашкою?

**Алгоритм сортування вибором** знаходить найменший (або найбільший) елемент і переміщує його на початок (або кінець) списку.

**Відмінність від сортування бульбашкою:**

* **Сортування бульбашкою** виконує багато перестановок (кожного разу, коли сусідні елементи знаходяться у неправильному порядку).
* **Сортування вибором** мінімізує кількість перестановок (максимум одна на ітерацію), але виконує більше порівнянь.

1. Яка часова складність алгоритмів сортування бульбашкою та сортування вибором в найгіршому, середньому та найкращому випадках?
2. В яких випадках доцільно використовувати сортування бульбашкою, а в яких - сортування вибором? Які переваги та недоліки кожного алгоритму

**Сортування бульбашкою доцільне:**

* Коли список майже відсортований (з оптимізацією алгоритму).
* Для навчання або демонстрації основ алгоритмів сортування.
* Для невеликих списків (до кількох сотень елементів).

**Переваги:** проста реалізація, можливість оптимізації.

**Недоліки:** низька ефективність для великих списків.

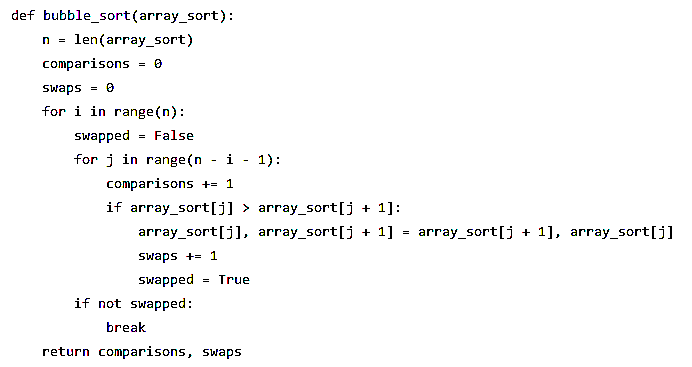
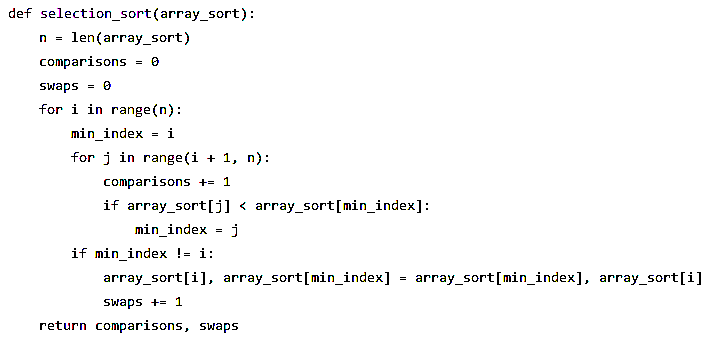
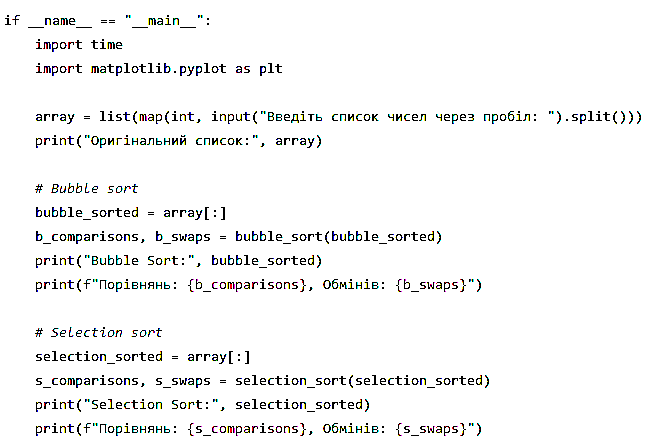
**Сортування вибором доцільне:**

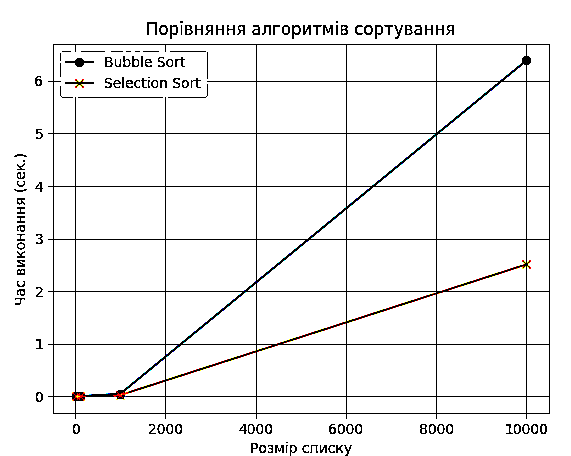
* Коли важливо мінімізувати кількість обмінів (наприклад, при роботі з пам’яттю, де обмін дорогий).
* Для невеликих списків.

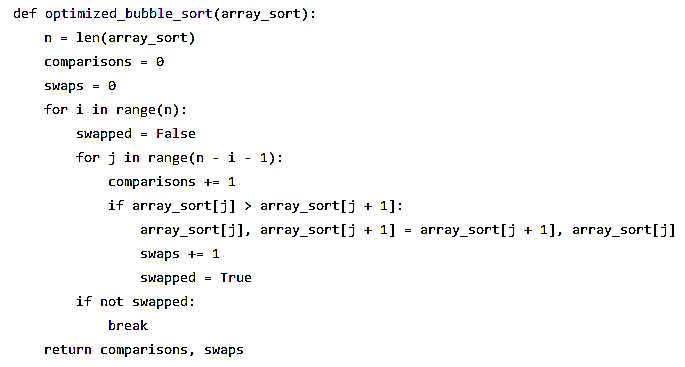
**Переваги:** менша кількість обмінів.

**Недоліки:** більше порівнянь, однакова часова складність у всіх випадках.

**Практичні завдання (Python):**

1. ****Ось функція bubble\_sort(array\_sort), яка сортує список array\_sort за допомогою алгоритму сортування бульбашкою.
2. Ось функція selection\_sort(array\_sort), яка сортує список array\_sort за допомогою алгоритму сортування вибором.
3. Це програма, яка зчитує список чисел з клавіатури та сортує його за допомогою обох алгоритмів. Виведіть відсортований список та кількість операцій порівняння та обміну, виконаних кожним алгоритмом.
4. Порівняйте ефективність алгоритмів сортування бульбашкою та сортування вибором на списках різної довжини (наприклад, 10, 1k, 100k елементів). Виміряйте час або кількість операцій виконання кожного алгоритму та \*побудуйте графік залежності часу(кількості операцій) виконання від розміру списку \*за допомогою Python.



1. [Додатково] Реалізуйте модифікацію алгоритму сортування бульбашкою, яка зупиняється, якщо під час проходу по списку не було зроблено жодного обміну.

**Висновок**

У дослідженні було розглянуто два класичних алгоритми сортування: сортування бульбашкою та сортування вибором.

Сортування бульбашкою ефективне для малих або майже відсортованих списків завдяки можливості оптимізації (переривання, якщо немає обмінів). Однак для великих наборів даних він має квадратичну складність O(n2)O(n2) і стає неефективним.

Сортування вибором також має складність O(n2)O(n2), але з мінімальною кількістю обмінів. Він підходить, коли важлива мінімізація обмінів, наприклад, для даних з обмеженими ресурсами пам'яті.

Таким чином, обидва алгоритми мають свої переваги та недоліки, і вибір між ними залежить від специфіки задачі. У загальному випадку, для малих або частково відсортованих даних можна використовувати сортування бульбашкою, а для великих — більш продуктивні алгоритми, такі як швидке сортування чи сортування злиттям.