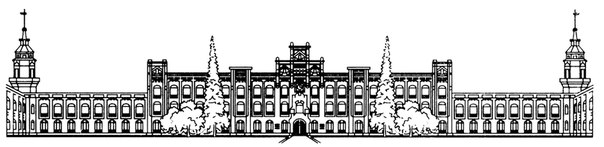
МІНІСТЕРСТВО ОСВІТИ ТА НАУКИ УКРАЇНИ

Національний технічний університет України

"Київський політехнічний інститут імені Ігоря Сікорського"



Факультет інформатики та обчислювальної техніки

Кафедра інформаційних систем та технологій

Програмування. Частина 2. Основи програмування

Практична робота №6

Тема: Алгоритми сортування

Варіант №11

Виконав: Коваль Богдан

студент групи ІС-31

Перевірив:

Майер І.С.

Київ-2024

**Тема:** Масиви 2

**Хід роботи:**

**Хід роботи:**

1. Ознайомитись з алгоритмами сортування:

1.1. Сортування обміном (сортування бульбашкою, Bubble sort)

https://uk.wikipedia.org/wiki/Сортування\_бульбашкою

https://en.wikipedia.org/wiki/Bubble\_sort

1.2. Сортування вибором (Selection sort)

https://uk.wikipedia.org/wiki/Сортування\_вибором

https://en.wikipedia.org/wiki/Selection\_sort

1.3. Сортування включенням (сортування вставкою, Insertion sort)

https://uk.wikipedia.org/wiki/Сортування\_включенням

https://en.wikipedia.org/wiki/Insertion\_sort

2. Реалізувати два методи сортування відповідно до свого варіанту з таблиці 1.

3. Відповісти на контрольні питання.



**Відповіді на контрольні питання**

1. **Порівняння методів сортування:**

- Сортування бульбашкою (Bubble Sort):

- Переваги: Простий для реалізації, не потребує додаткової пам'яті (сортує масив "на місці").

- Недоліки: Має квадратичну складність часу у найгіршому та середньому випадках, що робить його неефективним для великих масивів.

- Сортування вибором (Selection Sort):

- Переваги: Простий для реалізації, не потребує багато додаткової пам'яті.

- Недоліки: Має квадратичну складність часу незалежно від вхідних даних, що робить його неефективним для великих масивів.

- Сортування вставкою (Insertion Sort):

- Переваги: Ефективний для невеликих масивів, простий для реалізації.

- Недоліки: Має квадратичну складність часу у найгіршому та середньому випадках, що робить його менш ефективним для великих масивів.

2. Оцінка за критеріями:

- Час роботи: Усі три методи мають квадратичну складність у найгіршому випадку (O(n2)), проте для невеликих масивів сортування вставкою може бути найшвидшим.

- Потреби у додатковій пам'яті: Усі три методи мають постійну потребу у додатковій пам'яті (O(1)).

- Стабільність: Сортування вставкою зазвичай є стабільним, в той час як сортування бульбашкою та сортування вибором можуть порушувати порядок однакових елементів.

Інші критерії можуть включати:

- Ефективність у випадку частково відсортованих даних.

- Ефективність у випадку великих обсягів даних.

- Легкість реалізації та зрозумілість коду.

3. Найкращий та найгірший випадок:

- Найкращий випадок для всіх трьох методів - коли масив вже відсортований. У цьому випадку сортування вставкою зазвичай працює найкраще.

- Найгірший випадок - коли масив відсортований у зворотному порядку або майже відсортований. У цьому випадку сортування бульбашкою та сортування вибором можуть працювати дуже повільно.

4. Означення позначень O(1), O(n), O(n2):

- O(1): Постійний час. Час виконання алгоритму не змінюється зі збільшенням обсягу вхідних даних.

- O(n): Лінійний час. Час виконання алгоритму зростає лінійно зі збільшенням обсягу вхідних даних.

- O(n2): Квадратичний час. Час виконання алгоритму зростає квадратично зі збільшенням обсягу вхідних даних.