

Контрольні питання:

1. Порівняйте та назвіть переваги та недоліки таких методів сортування: сортування бульбашкою, сортування вибором, сортування вставкою.

а.

This algorithm has several advantages. It is simple to write, easy to understand and it only takes a few lines of code. The data is sorted in place so there is little memory overhead and, once sorted, the data is in memory, ready for processing. **The major disadvantage is the amount of time it takes to sort.**

2. Оцініть кожен з методів відповідно до наступних критеріїв:

- час роботи;
- потреби у додатковій пам'яті;
- стабільність.

За якими ще критеріями ви можете порівняти ці алгоритми?

**Sorting algorithm**

2. Який випадок є найкращім або найгіршим для роботи цих алгоритмів? (Наприклад: частково відсортований масив або масив, відсортований у зворотному порядку). Який з методів забезпечить у цих випадках найкращій/найгірший результат?

а.

**Timsort** is "an adaptive, stable, natural mergesort" with "supernatural performance on many kinds of partially ordered arrays (less than  $\lg(N!)$  comparisons needed, and as few as  $N-1$ )". Python's built-in `sort()` has used this algorithm for some time, apparently with good results.

4. Що означають позначення  $O(1)$ ,  $O(n)$ ,  $O(n^2)$ ?

Big O notation is a mathematical notation that describes the limiting behavior of a function when the argument tends towards a particular value or infinity. Big O is a member of a family of notations invented by Paul Bachmann,[1] Edmund Landau,[2] and others, collectively called Bachmann–Landau notation or asymptotic notation.

**$O(n^2)$  represents a function whose complexity is directly proportional to the square of the input size.** Adding more nested iterations through the input will increase the complexity which could then represent  $O(n^3)$  with 3 total iterations and  $O(n^4)$  with 4 total iterations