

Алгоритмічні задачі

- A1.** Напишіть програми для сортування масиву чисел методом вставки та вибором. Додайте код для візуалізації процесу сортування, значення у комірках масиву подайте у вигляді нормованих вертикальних стовпців (гістограма).
- A2.** Розв'яжіть олімпіадну задачу:
<http://www.e-olymp.com/uk/problems/1070>.

Експериментальні задачі

- E1.** Підберіть масив із 100 елементів, який містить різні числа від 1 до 100, так щоб сортування Шела з кроками 1-4-13-40 виконувалось з максимальною кількістю порівнянь.
- E2.** Попросіть ваших знайомих відсортувати колоду карт. Зверніть увагу як вони це роблять і запрограмуйте ці методи. Проаналізуйте швидкість виконання отриманого способу сортування.
- E3.** Напишіть реалізацію сортування вставкою, яка усуває із внутрішнього циклу перевірку $j > 0$, перемістивши спочатку найменший елемент на початок. Оцініть ефективність такої модифікації.
- Таким чином можна уникнути перевірки на досягнення кінця масиву і в інших випадках. Елемент, який дозволяє усунути перевірку, називається маркером (sentinel).
- E4.** Експериментально визначте значення t , при якому отримується мінімальний час сортування Шела для випадкових масивів в послідовності кроків $1, [t], [t^2], [t^3], [t^4], \dots$ для $N=10^6$.
- E5.** Проаналізуйте швидкість роботи простих алгоритмів сортування для специфічних масивів, які можуть бути отримані в реальній ситуації:
- a) уже впорядковані масиви;
 - b) масиви у зворотному порядку;
 - c) масиви, що містять лише два різних елементи.
 - d) нерівномірні дані:
 - (1) половина даних рівна 0, інша – 1;
 - (2) половина даних рівна 0, чверть – 1, інші – 2.
 - (3) половина 0, інша половина випадкові числа.
- E6.** Напишіть клієнт для генерації частково-впорядкованих масивів:
- a) 95% впорядковані, інші містять випадкові дані;
 - b) всі елементи знаходяться в межах 10 позицій від кінцевого положення;
 - c) впорядковані дані, крім 5% елементів, які випадково розміщені в масиві.