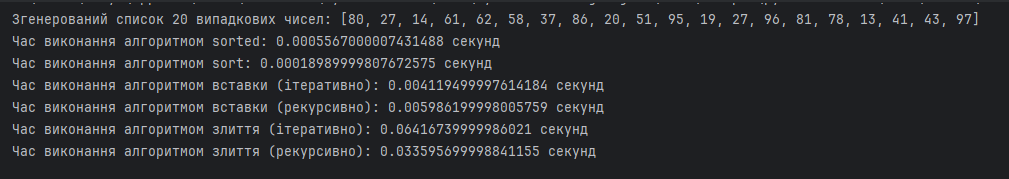
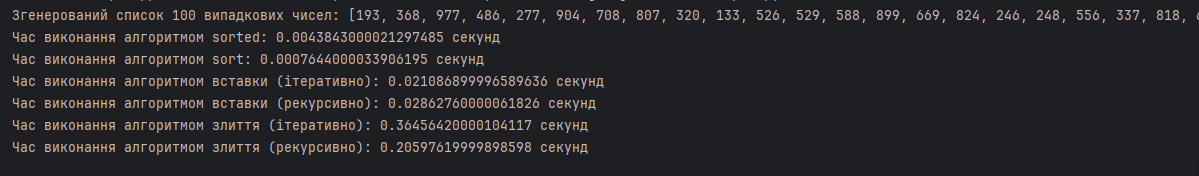
Результати вимірювань часу сортування по списку з 20 випадкових чисел



Результати вимірювань часу сортування по списку з 100 випадкових чисел



**Аналіз результатів вимірювань часу сортування по списку з 100 випадкових чисел:**

1. **Алгоритми Timesort (sort і sorted)**:
   * Час виконання: 0.004384 секунд (при використанні алгоритму sorted()).
   * Час виконання для алгоритму sort() (метод списку): 0.000764 секунд. Це найбільш оптимізований алгоритм, що демонструє найшвидший час виконання.
2. **Алгоритм вставки**:
   * **Ітеративна версія**: 0.021 секунд.
   * **Рекурсивна версія**: 0.028 секунд. Обидва варіанти працюють повільніше в порівнянні з вбудованими функціями.
3. **Алгоритм злиття**:
   * **Ітеративна версія**: 0.364 секунди.
   * **Рекурсивна версія**: 0.205 секунд. Алгоритм злиття у цьому прикладі демонструє більший час виконання, ніж вбудовані функції Python. Ітеративна версія працює повільніше, ніж рекурсивна.

**Висновки:**

* Вбудовані методи сортування sorted() і sort() є найбільш ефективними для цього списку, оскільки вони оптимізовані.
* Алгоритм вставки в нашому випадку більш ефективний, ніж алгоритм злиття, і в ньому ітеративна версія трохи швидша за рекурсивну.
* Алгоритм злиття рекурсивно працює швидше, ніж ітеративно, але обидві реалізації поступаються вбудованим методам.