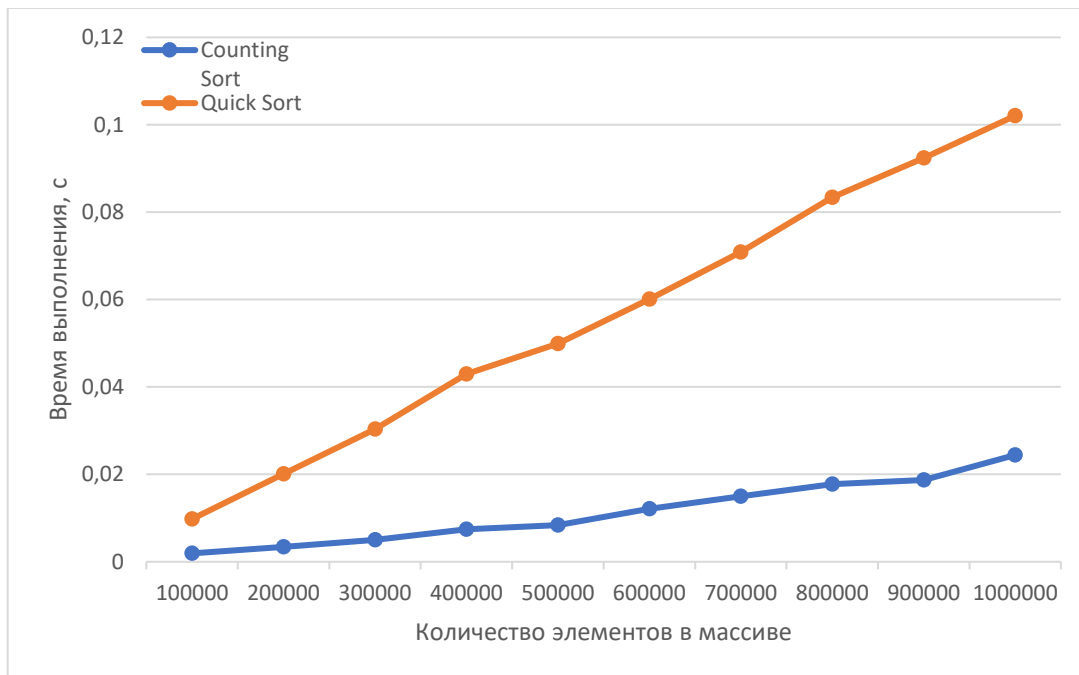
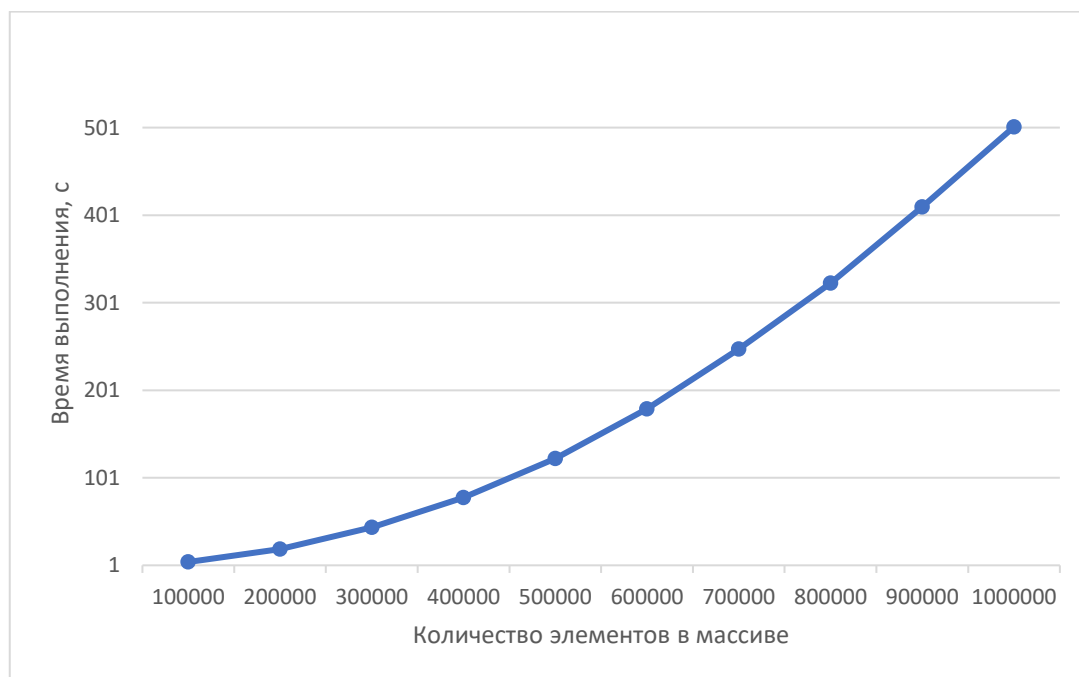


| #  | Количество элементов в массиве | Counting Sort | Insertion Sort | Quick Sort |
|----|--------------------------------|---------------|----------------|------------|
| 1  | 50000                          | 0,001208      | 1,208306       | 0,004552   |
| 2  | 100000                         | 0,001926      | 4,795559       | 0,009788   |
| 3  | 150000                         | 0,002980      | 10,993341      | 0,014967   |
| 4  | 200000                         | 0,003409      | 19,564763      | 0,020102   |
| 5  | 250000                         | 0,004279      | 30,874486      | 0,025517   |
| 6  | 300000                         | 0,005019      | 44,452471      | 0,030323   |
| 7  | 350000                         | 0,006442      | 60,531256      | 0,035380   |
| 8  | 400000                         | 0,007403      | 78,513135      | 0,042960   |
| 9  | 450000                         | 0,008203      | 102,157211     | 0,046608   |
| 10 | 500000                         | 0,008376      | 123,086487     | 0,049908   |
| 11 | 550000                         | 0,009975      | 151,434659     | 0,056278   |
| 12 | 600000                         | 0,012125      | 179,591548     | 0,060083   |
| 13 | 650000                         | 0,014130      | 213,284402     | 0,066491   |
| 14 | 700000                         | 0,014983      | 248,026504     | 0,070872   |
| 15 | 750000                         | 0,015805      | 286,704944     | 0,077110   |
| 16 | 800000                         | 0,017731      | 323,268141     | 0,083367   |
| 17 | 850000                         | 0,018234      | 325,584995     | 0,090536   |
| 18 | 900000                         | 0,018705      | 410,404175     | 0,092368   |
| 19 | 950000                         | 0,019511      | 454,262714     | 0,096456   |
| 20 | 1000000                        | 0,024423      | 501,797948     | 0,102052   |



«Зависимость времени выполнения алгоритмов Quick Sort и Merge Sort от количества элементов в массиве»



«Зависимость времени выполнения алгоритма **Insertion sort** от размера массива»

## Контрольные вопросы

- 1, Вычислительная сложность алгоритма - это количество вычислительных ресурсов, необходимых для выполнения алгоритма, как правило, измеряется в количестве операций.
- 2,  $f(n) = O(g(n))$  означает, что  $f(n)$  растет не быстрее, чем  $g(n)$ ,  $f(n) = \Theta(g(n))$  означает, что  $f(n)$  растет пропорционально  $g(n)$ ,  $f(n) = \Omega(g(n))$  означает, что  $f(n)$  растет не медленнее, чем  $g(n)$ .
- 3, Устойчивый (stable) алгоритм сортировки сохраняет относительный порядок элементов с одинаковыми ключами в отсортированном массиве.
- 4, Алгоритм сортировки "на месте" (in-place) не требует дополнительной памяти для хранения временных данных при сортировке.
- 5, Вычислительная сложность в худшем случае:  
Counting Sort -  $O(n+k)$   
Quicksort -  $O(n^2)$ , обычно  $O(n \log n)$ ,  
Insertion Sort -  $O(n^2)$

Наихудший случай для быстрой сортировки происходит, когда выбранный элемент для разделения массива является наименьшим или наибольшим элементом в массиве, Это может привести к ситуации, когда массив делится на две подмассива различных размеров, что увеличивает количество сравнений и операций сдвига, что снижает эффективность алгоритма.

- 6, На графике можно наблюдать, что алгоритмы Counting и Quick sort работают значительно быстрее алгоритма Insertion sort на всех размерах входных данных. Кривая Insertion sort имеет более плавный характер, что говорит о лучшей вычислительной сложности алгоритмов по сравнению с алгоритмами Counting и Quick sort. Экспериментальные результаты согласуются с оценкой вычислительной сложности алгоритмов. Алгоритм Counting Sort имеет временную сложность  $O(n+k)$ , а для Quicksort -  $O(n \log n)$ , Insertion Sort имеет временную сложность  $O(n^2)$ , что отражается на более крутой кривой на графике.
- 7, Алгоритмы сортировки с вычислительной сложностью  $O(n \log n)$  для худшего случая: сортировка слиянием, пирамидальная сортировка.
- 8, Сортировка Radix sort -  $O(nk)$ .