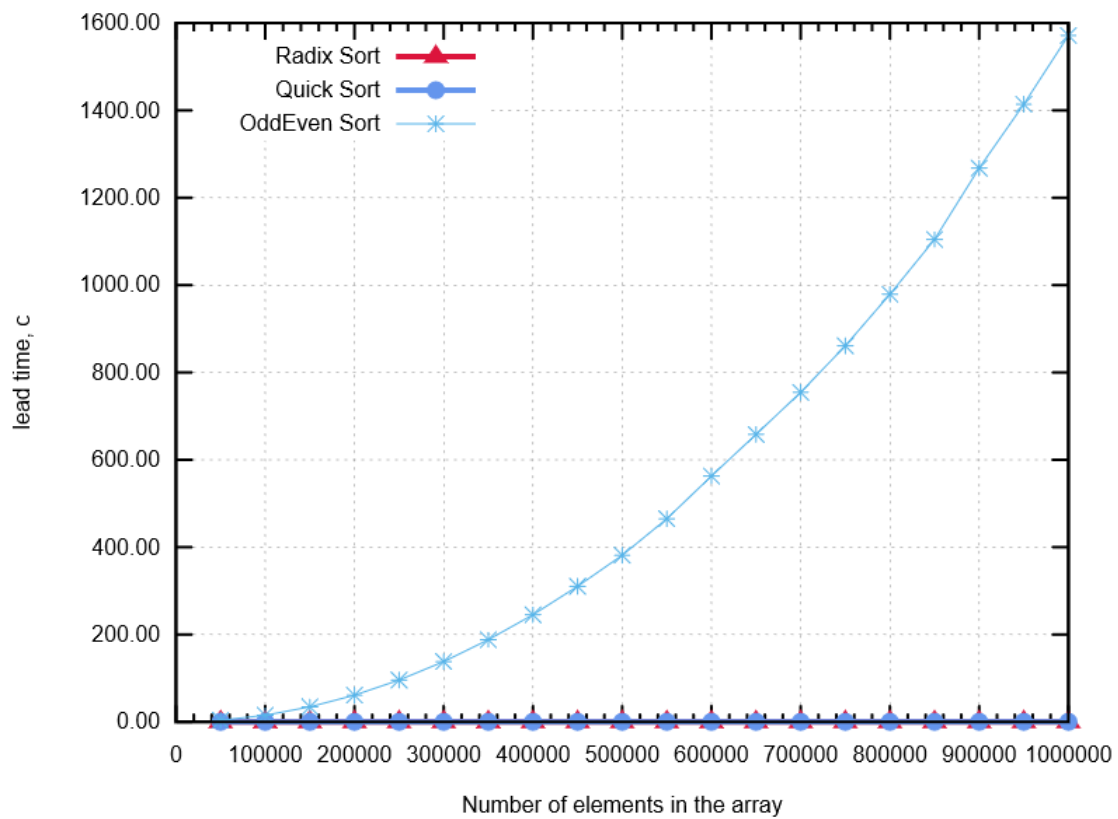
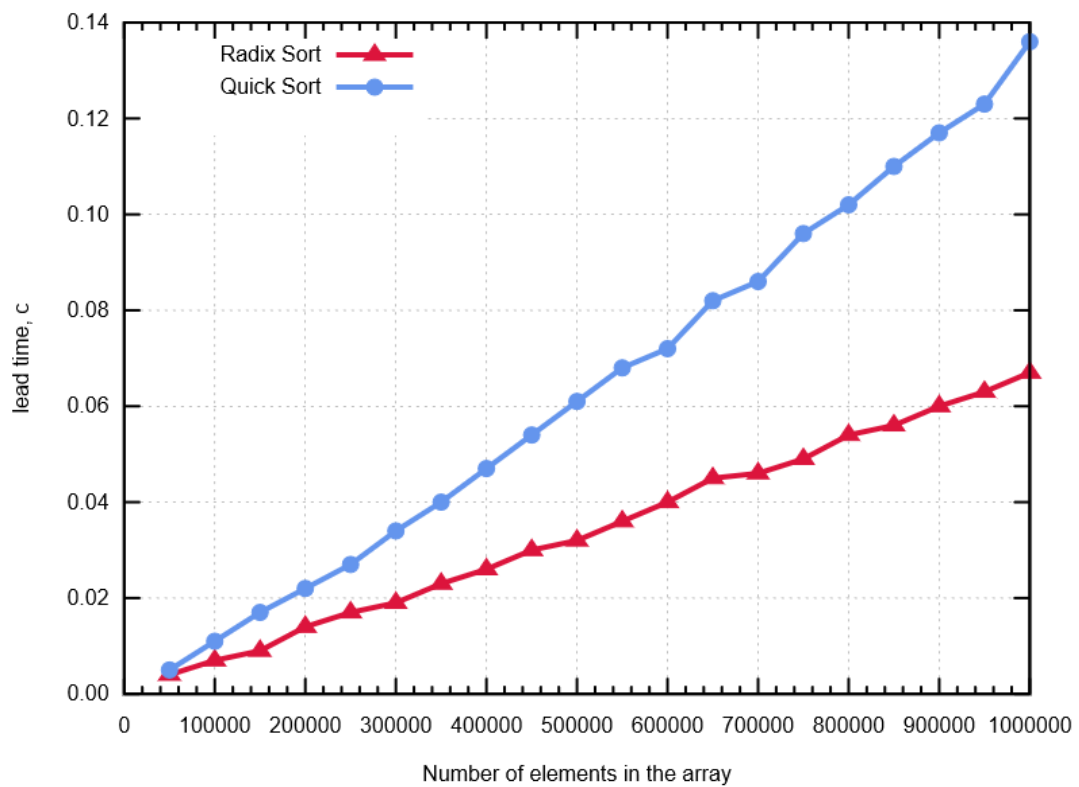


#	Количество элементов в массиве	Radix Sort, с	Quick Sort, с	Odd Even Sort, с
1	50000	0.004000	0.005000	3.656000
2	100000	0.007000	0.011000	15.134000
3	150000	0.009000	0.017000	34.332000
4	200000	0.014000	0.022000	61.059000
5	250000	0.017000	0.027000	95.698000
6	300000	0.019000	0.034000	137.991000
7	350000	0.023000	0.040000	188.066000
8	400000	0.026000	0.047000	245.245000
9	450000	0.030000	0.054000	310.339000
10	500000	0.032000	0.061000	381.443000
11	550000	0.036000	0.068000	464.898000
12	600000	0.040000	0.072000	562.790000
13	650000	0.045000	0.082000	658.199000
14	700000	0.046000	0.086000	754.439000
15	750000	0.049000	0.096000	860.928000
16	800000	0.054000	0.102000	979.009000
17	850000	0.056000	0.110000	1104.439000
18	900000	0.060000	0.117000	1267.811000
19	950000	0.063000	0.123000	1413.926000
20	1000000	0.067000	0.136000	1571.434000



«Зависимость времени выполнения алгоритмов **Quicksort**, **Radix Sort**, **Odd Even Sort** от размера массива»



«Зависимость времени выполнения алгоритмов **Quicksort**, **Radix 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:
Сортировка чет-нечет - $O(n^2)$.
Сортировка Radix sort - $O(nk)$.
Алгоритм Radix sort не имеет худшего случая, так как его временная сложность $O(nk)$, где n - количество элементов, а k - количество разрядов в числах.
Быстрая сортировка - $O(n^2)$, обычно $O(n \log n)$.
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 Radix и Quicksort работают значительно быстрее алгоритма Odd-Even sort на всех размерах входных данных. Кривые для Radix и Quicksort имеют более плавный характер, что говорит о лучшей вычислительной сложности алгоритмов по сравнению с алгоритмом Odd-Even sort. Экспериментальные результаты согласуются с оценкой вычислительной сложности алгоритмов. Для алгоритма Radix вычислительная сложность составляет $O(nk)$, а для Quicksort - $O(n \log n)$. Алгоритм Odd-Even sort имеет вычислительную 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 - $O(nk)$.