

Анализ различных сортирововок

Данные о скорости тех или иных сортирововок основаны на среднем времени за 5 прогонов сортировки, до измерения времени так же совершалось два "холостых" прогона сортировки.

Замеры для массивов размерности до 4100 элементов

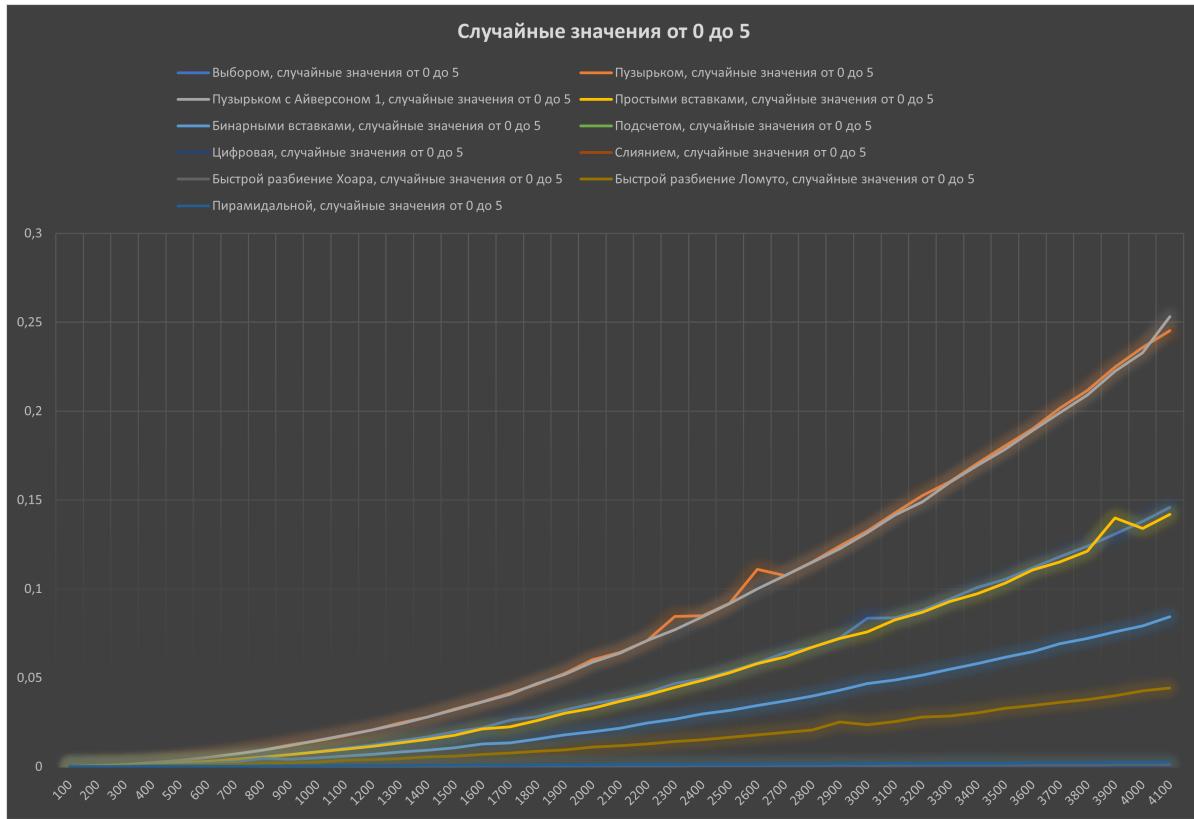


Рис. 1: Сравнение сортировок для массива со случайными значениями от 0 до 5

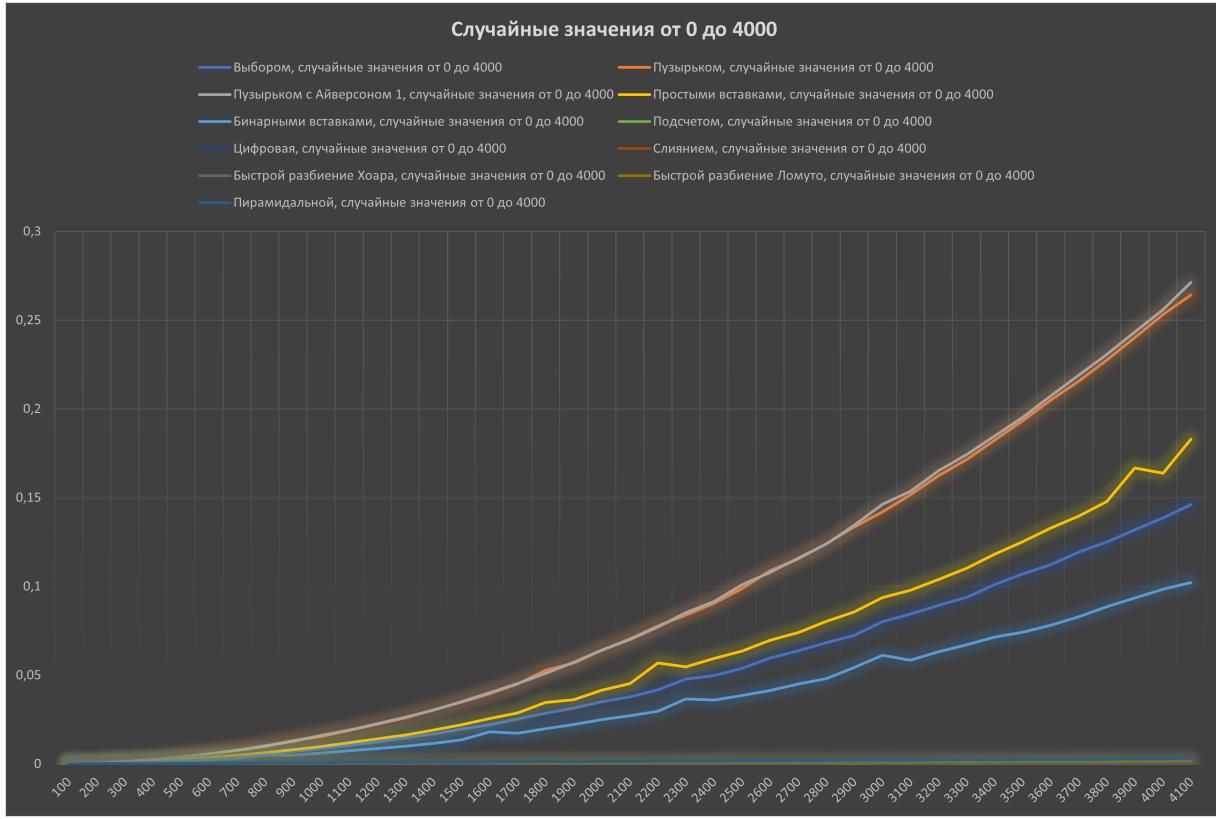


Рис. 2: Сравнения сортировок для массива со случайными значениями от 0 до 4000

Из первых двух графиков видно, что сортировки, для которых требуется выполнение определенных условий для исходных данных, показывают свое наилучшее время, для сортировки подсчета достаточно небольшой диапазон значений и размер массива, аналогично для цифровой. В остальном на первых двух графиках сортировки как раз ведут себя предсказуемо.

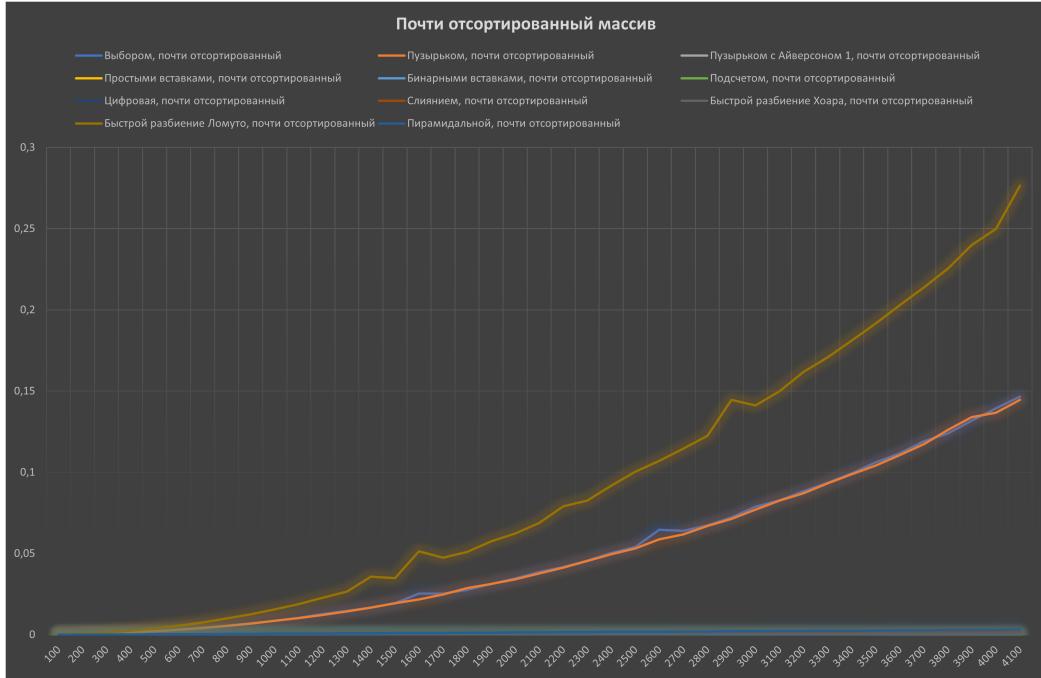


Рис. 3: Сравнение сортировок для почти отсортированного массива

На почти отсортированном наборе данных быстрая сортировка с разбиением Ломуту показывает наихудшее время, так как происходит слишком большое количество итераций по отсортированной части массива и обменов элемента с самим собой, пока алгоритм не натыкается на место смещения. Пузырек с условием Айвэрсона справляется с этой задачей достаточно быстро, так как не требует заново проходиться по отсортированной части, а начинает цикл с момента последнего обмена.

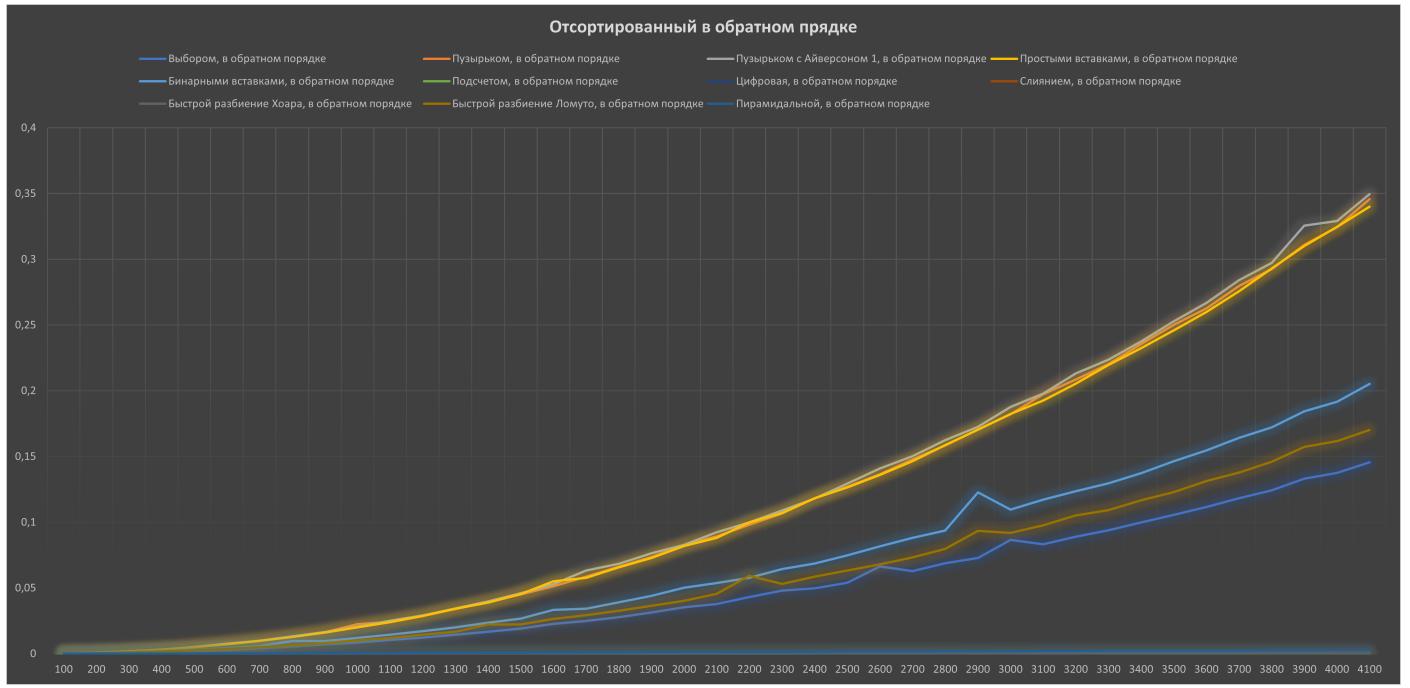


Рис. 4: Сравнение сортировок для массива, отсортированного в обратном порядке

Для массива, отсортированного в обратном порядке сортировки пузырьком(обе) и вставками показывают наихудшее время, так как на каждой итерации необходимо делать обмен. Соответственно модернизированная версия вставок ускоряет процесс поиска места для элемента. Сортировка выбором работает как и должна, за свое худшее время. Графики этих сортировок так отличаются из-за количества обменов, которые необходимо совершить, то есть отличаются в константу.

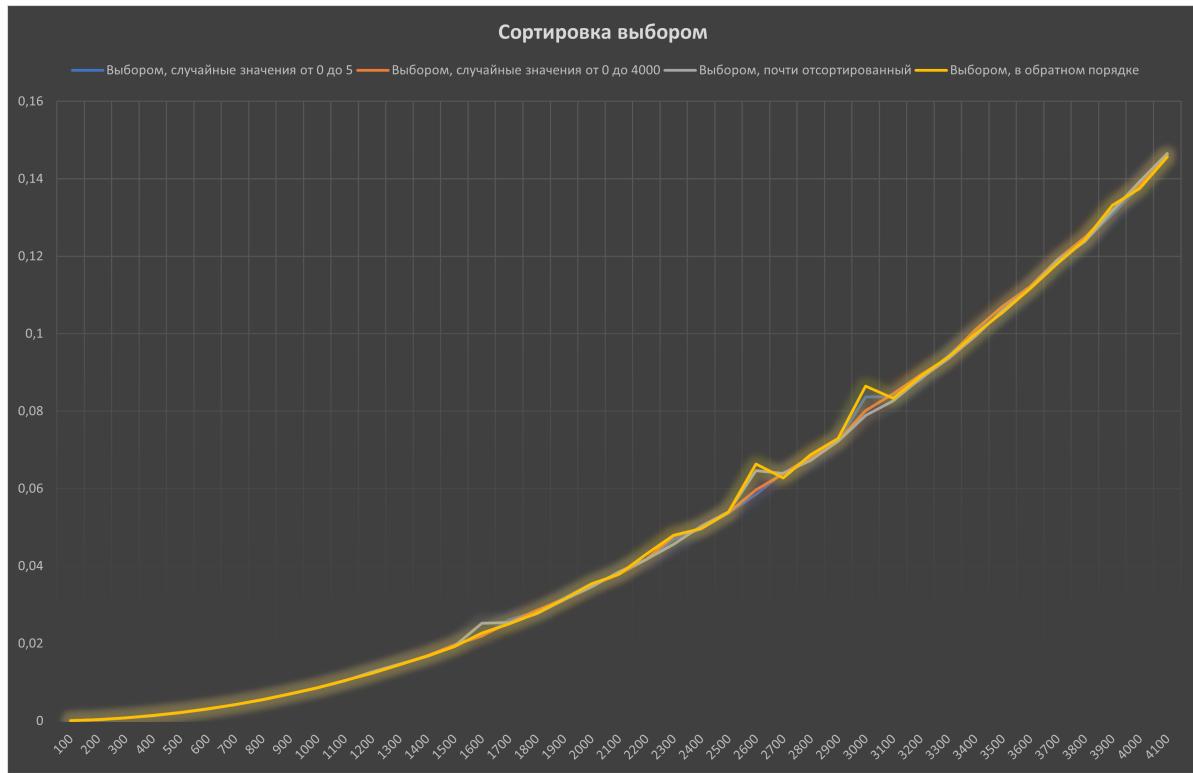


Рис. 5: Сравнение скорости сортировки выбором для различных видов массивов

Сортировка выбором показывает идентичные результаты по времени, так как ее наихудшее время совпадает с наилучшим, независимо от набора данных.

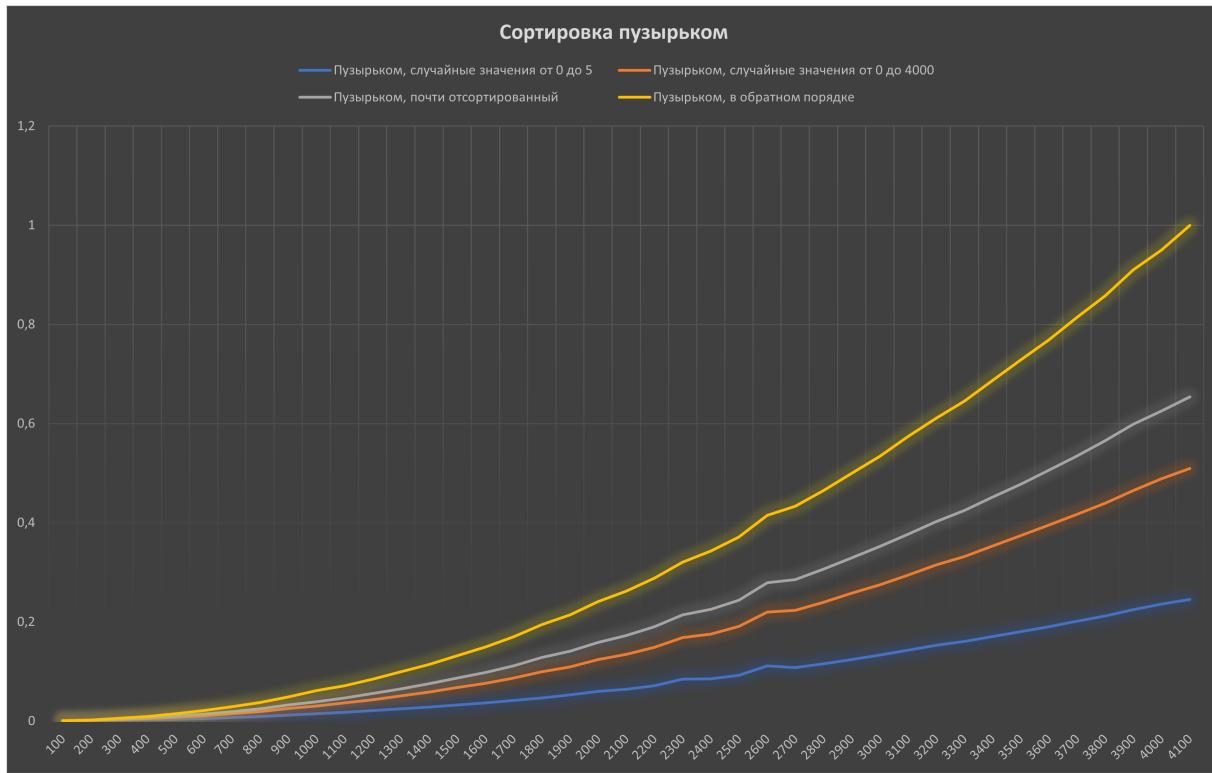


Рис. 6: Сравнение скорости сортировки пузырьком для различных видов массивов

Сортировка пузырьком пузырьком для массива в обратном порядке так же занимает так много времени из-за большого количества обменов

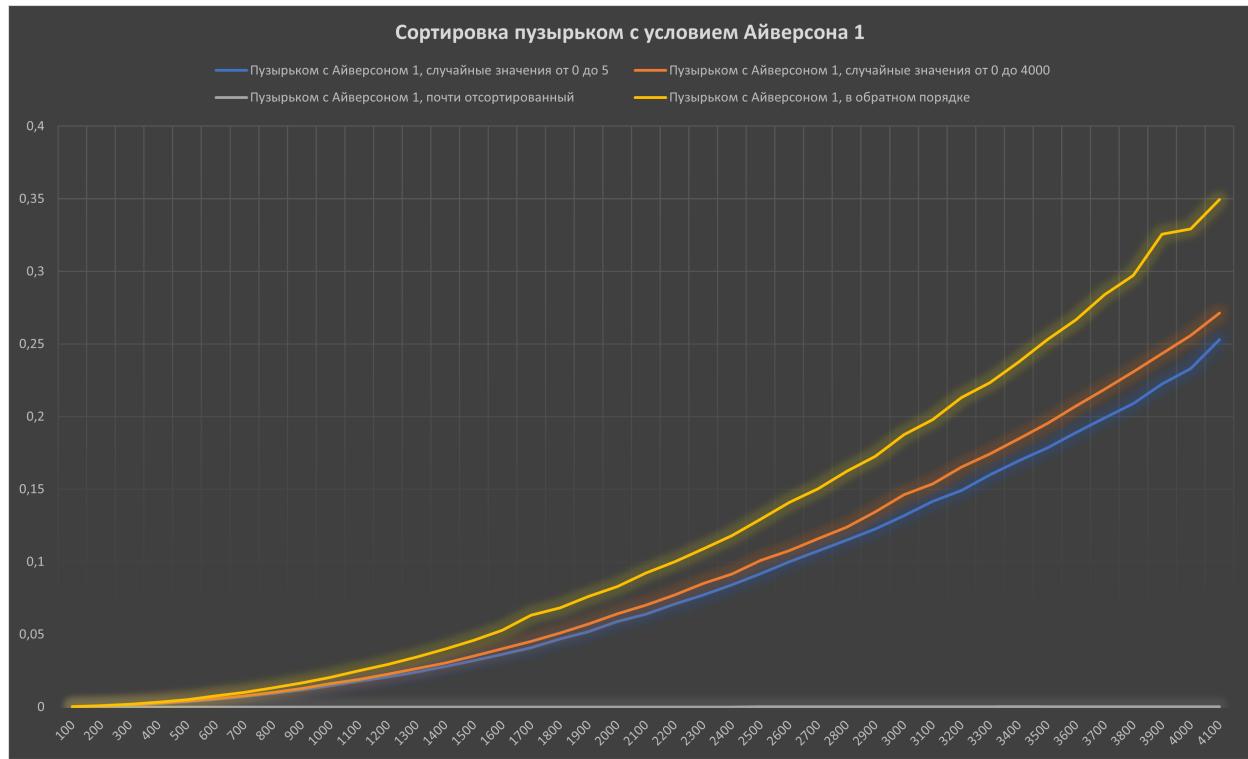


Рис. 7: Сравнение скорости сортировки пузырьком с условием Айверсона 1 для различных видов массивов

Аналогично с предыдущим пунктом

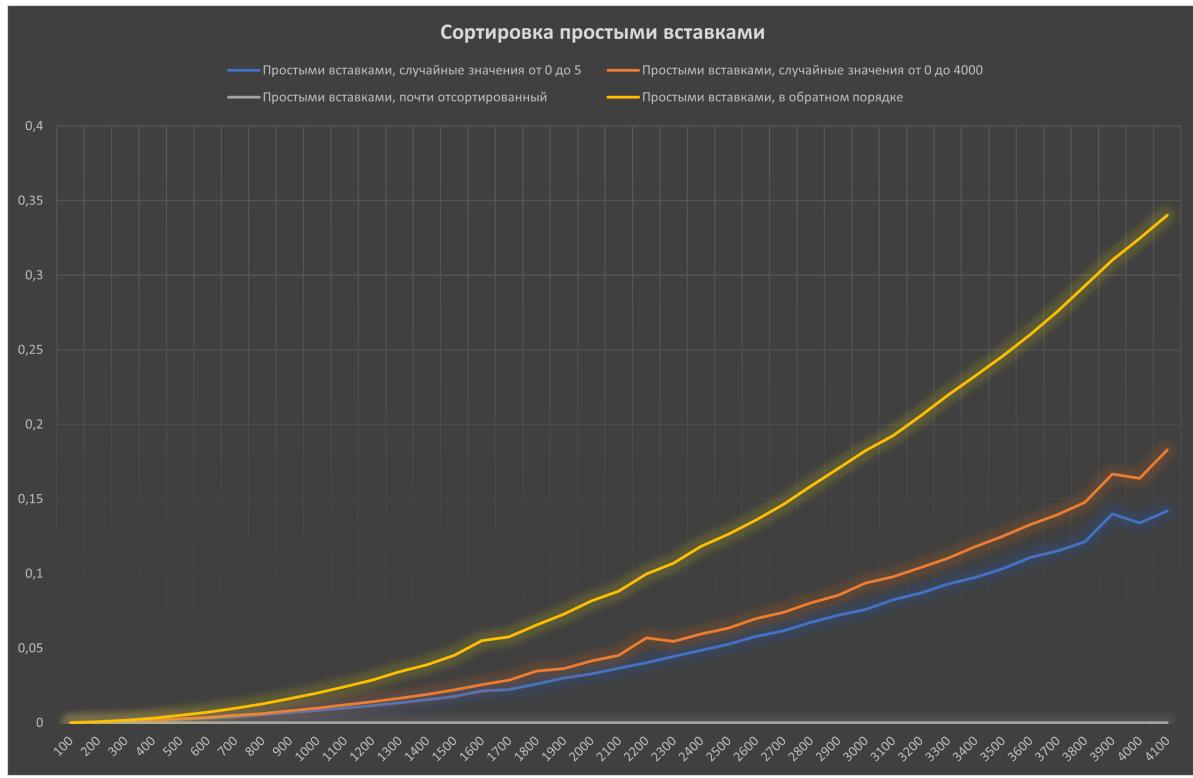


Рис. 8: Сравнение скорости сортировки простыми вставками для различных видов массивов

Сортировка простыми вставками сортируем массив, в котором все элементы почти отсортированы быстро, так как это почти наилучший случай.

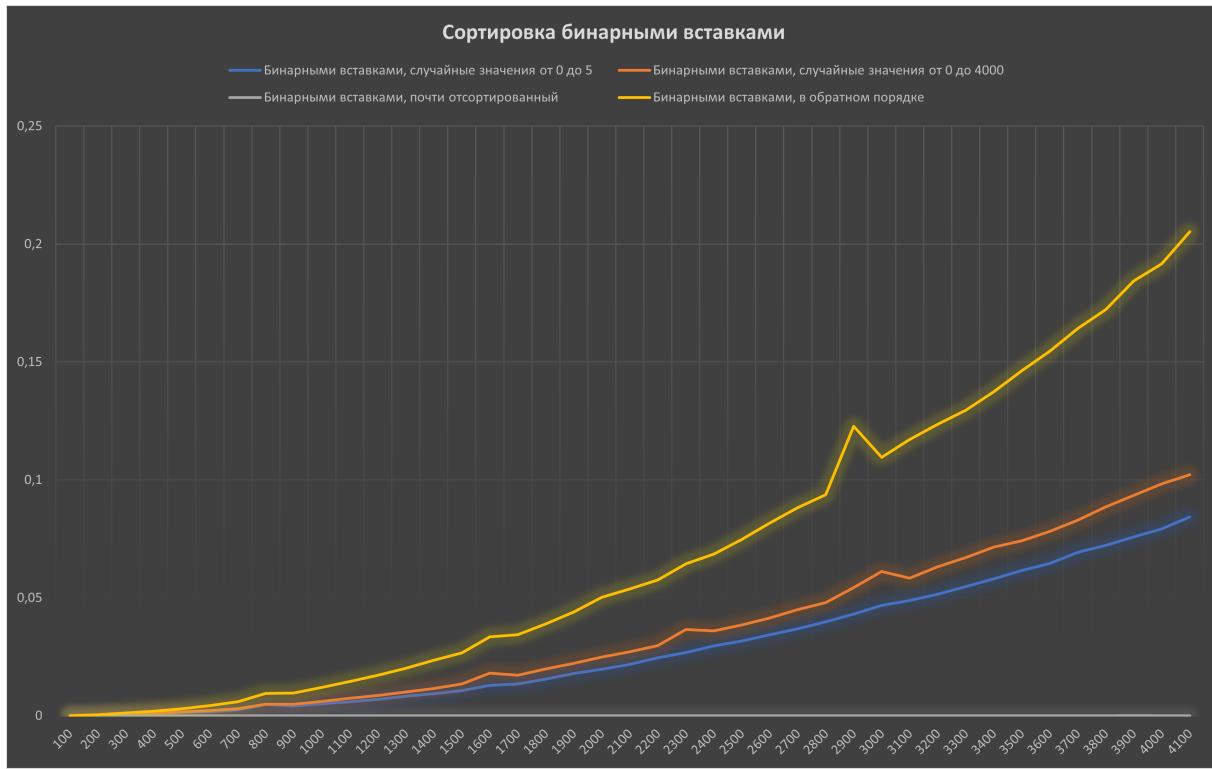


Рис. 9: Сравнение скорости сортировки бинарными вставками для различных видов массивов

Аналогично с предыдущим пунктом

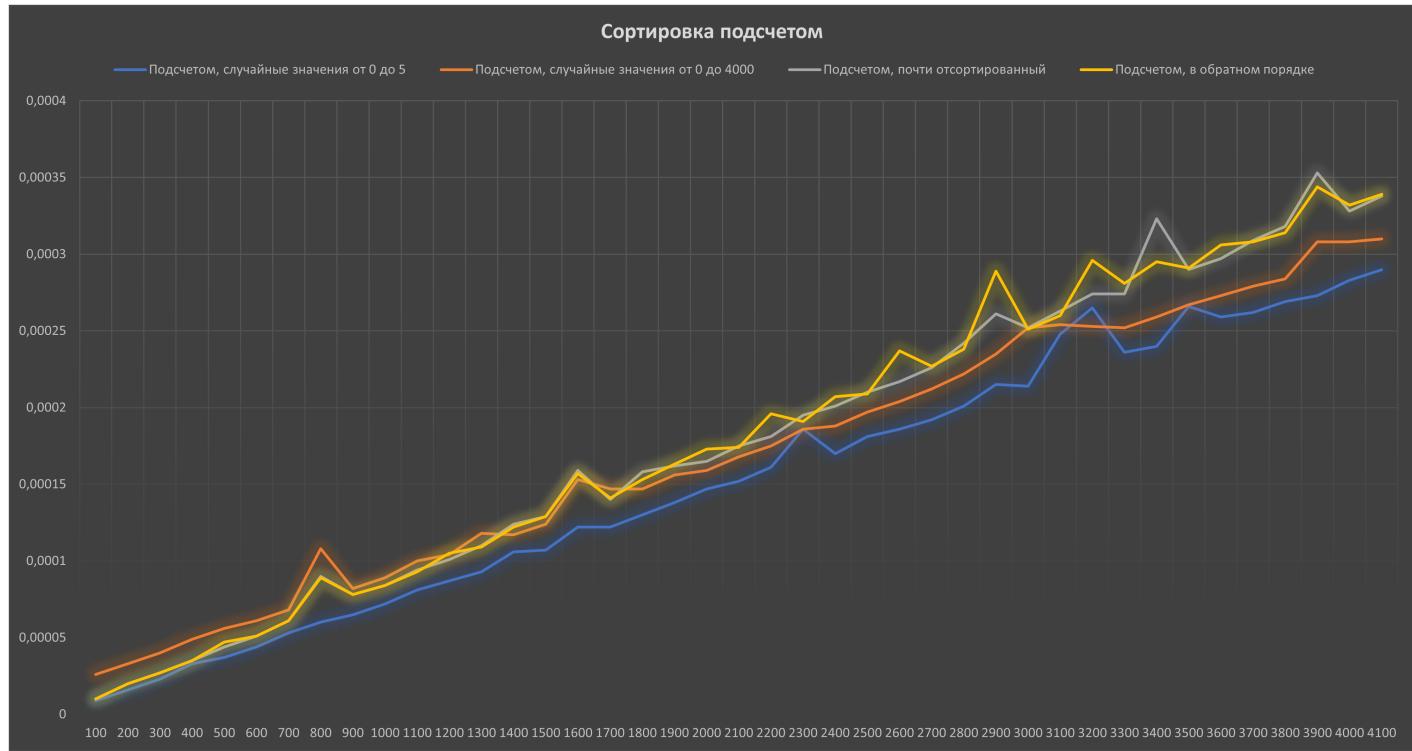


Рис. 10: Сравнение скорости сортировки подсчетом для различных видов массивов

Сортировка подсчетом работает за линейное время, быстро, так как имеем небольшой диапазон значений.

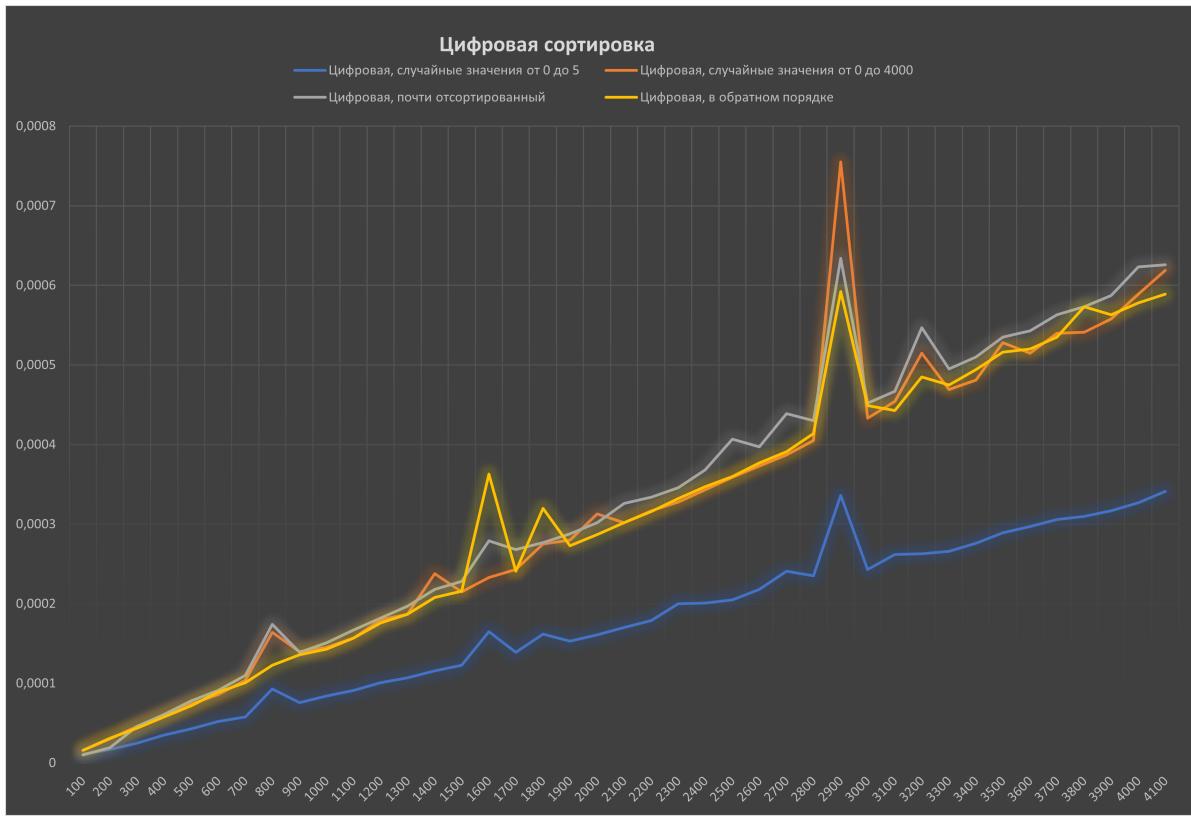


Рис. 11: Сравнение скорости цифровой сортировки для различных видов массивов

Цифровая сортировка работает на значениях от 0 до 5 быстрее, тк в числе меньше разрядов.

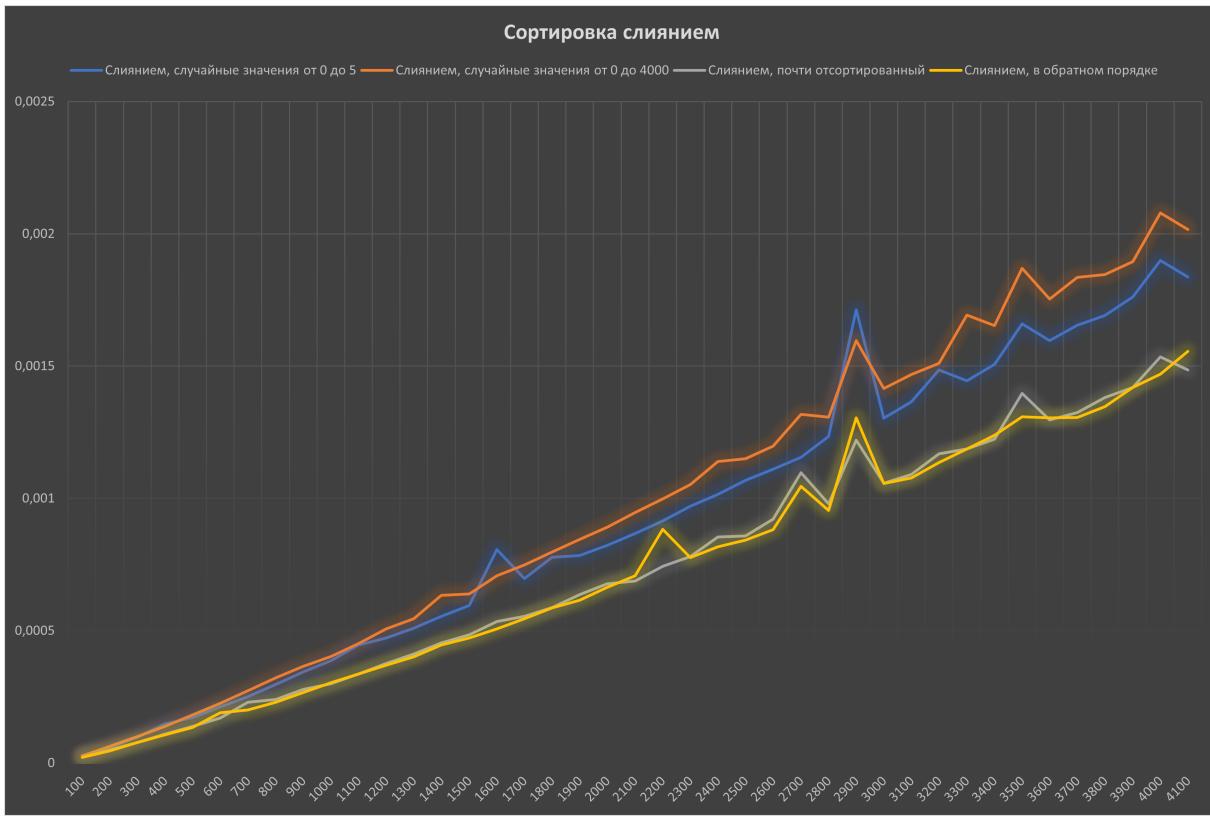


Рис. 12: Сравнение скорости сортировки слиянием для различных видов массивов

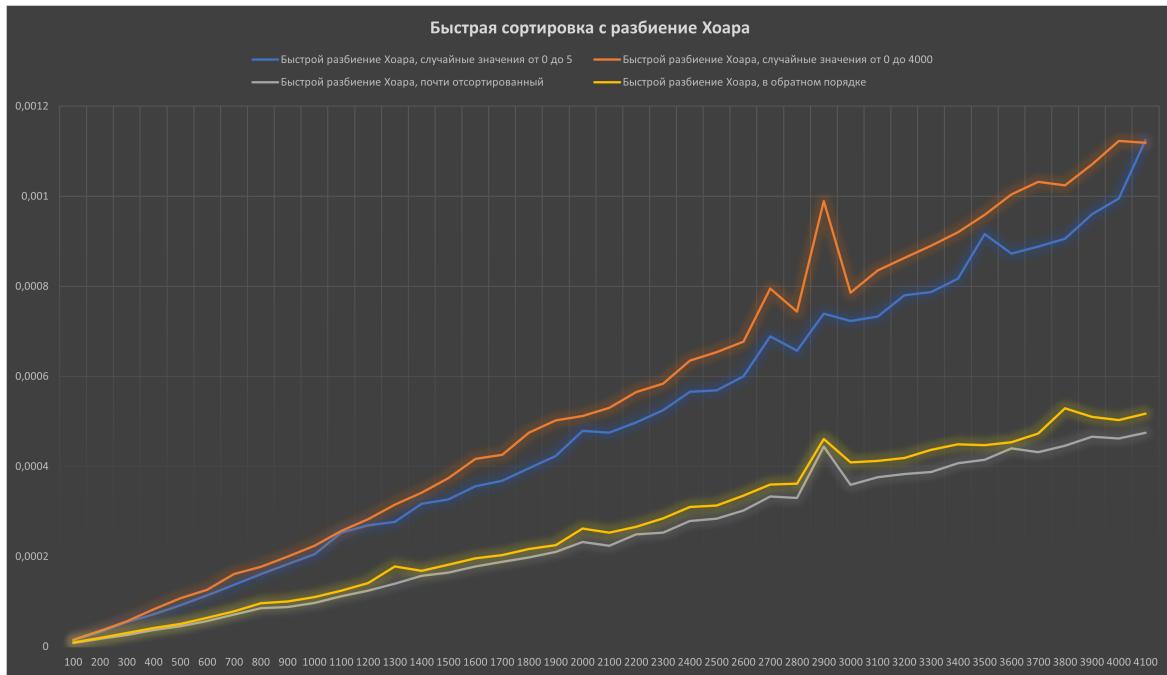


Рис. 13: Сравнение скорости быстрой сортировки соазбиением Хоара для различных видов массивов

Быстрая сортировка сортирует массив в обратном порядке и почти отсортированный быстрее, чем остальные, тк почти все элементы в правой части строго меньше или опорного(в обратном меньше, в отсортированном больше).

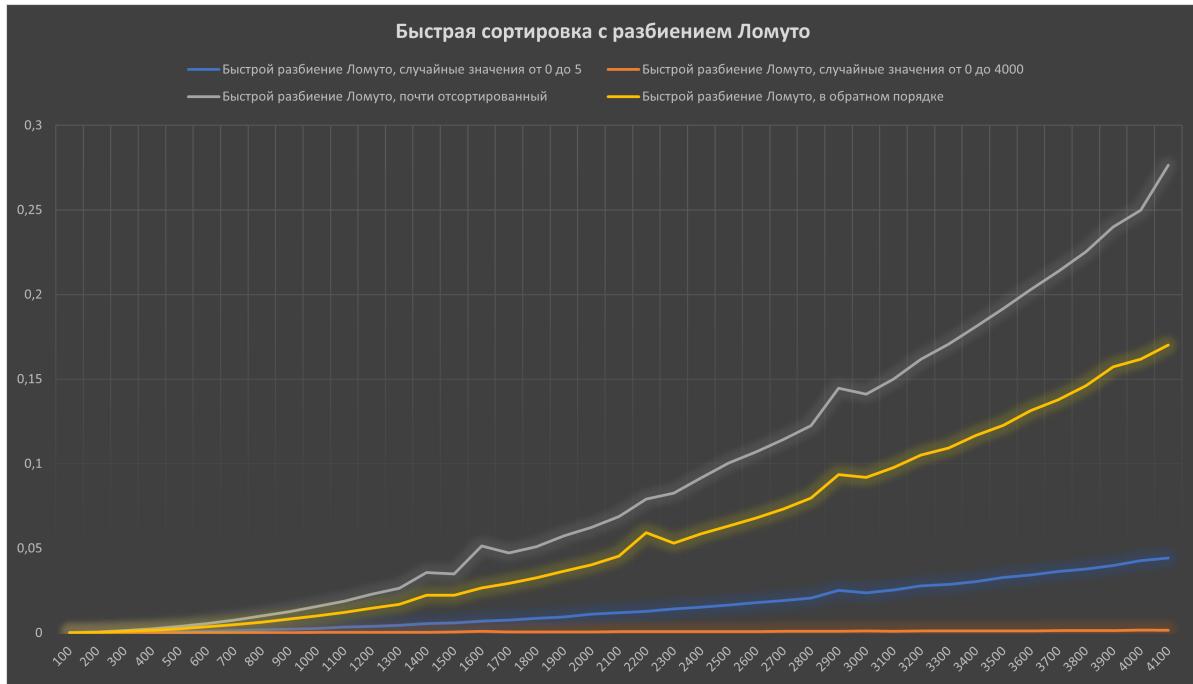


Рис. 14: Сравнение скорости быстрой сортировки с разбиением Ломуто для различных видов массивов

На почти отсортированном наборе данных быстрая сортировка с разбиением Ломуто показывает наихудшее время, так как происходит слишком большое количество итераций по отсортированной части массива и обменов элемента с самим собой, пока алгоритм не натыкается на место смещения.

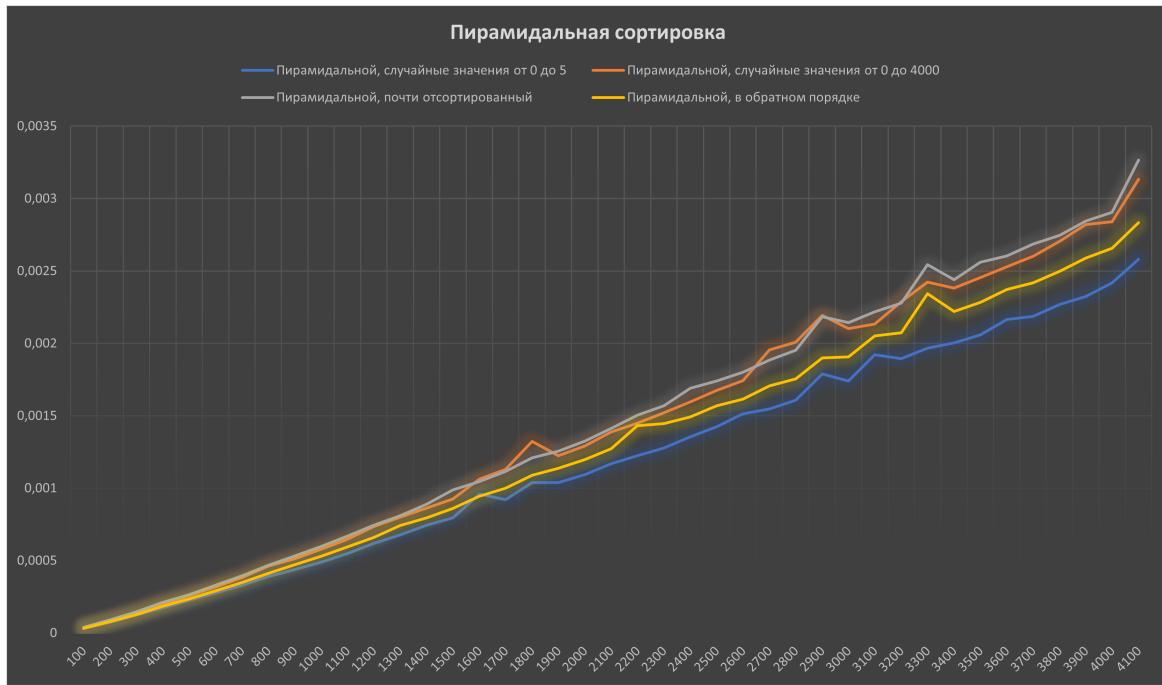


Рис. 15: Сравнение скорости пирамидалной сортировки для различных видов массивов

Пирамидалная сортировка работает одинаково на всех массивах, тк алгоритм не выигрывает времени на различных данных.

Замеры для массивов размерности до 300 элементов

Для массивов размерности 300 комментарии аналогичные.

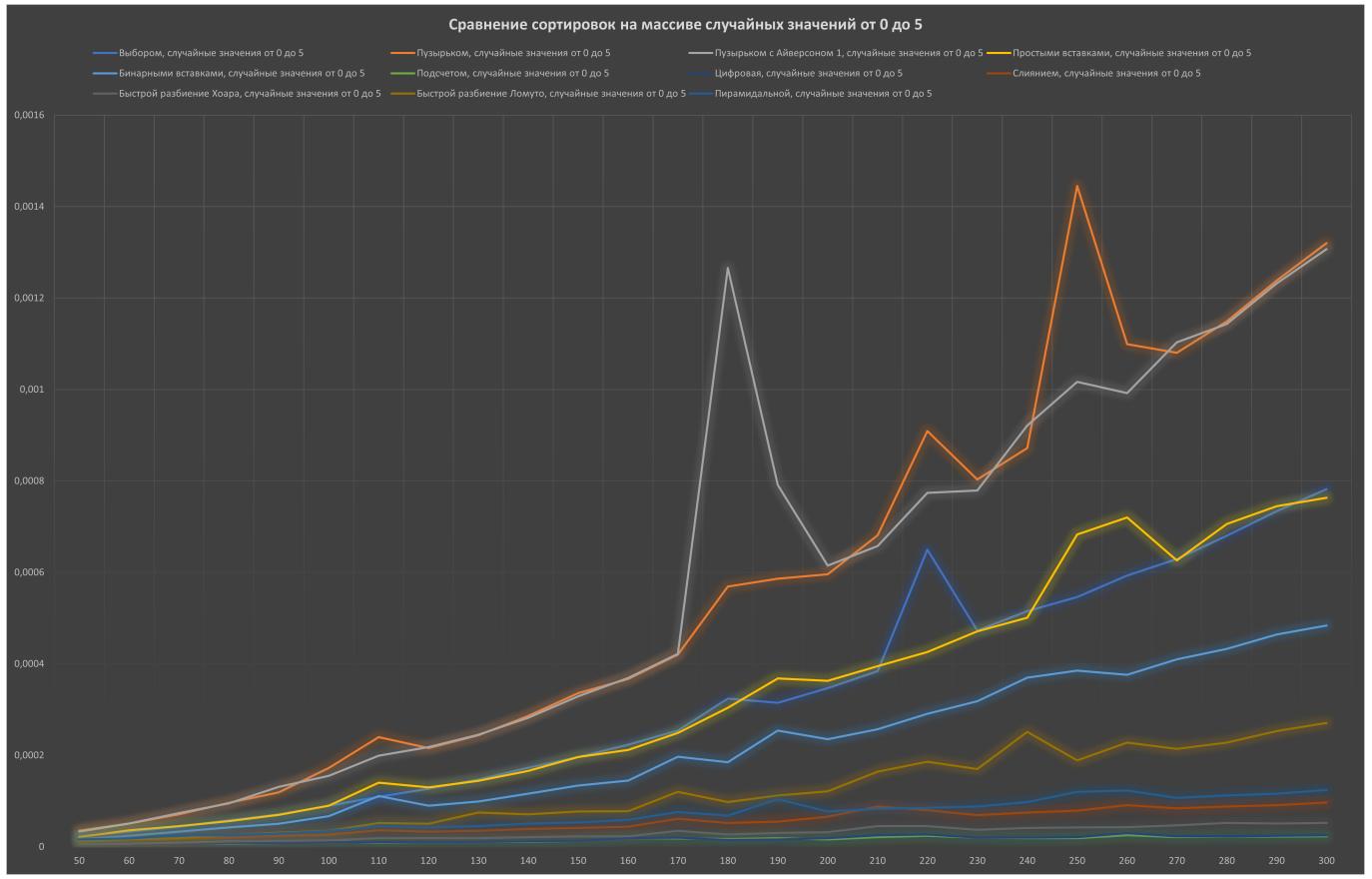


Рис. 16: Сравнение сортировок для массива со случайными значениями от 0 до 5

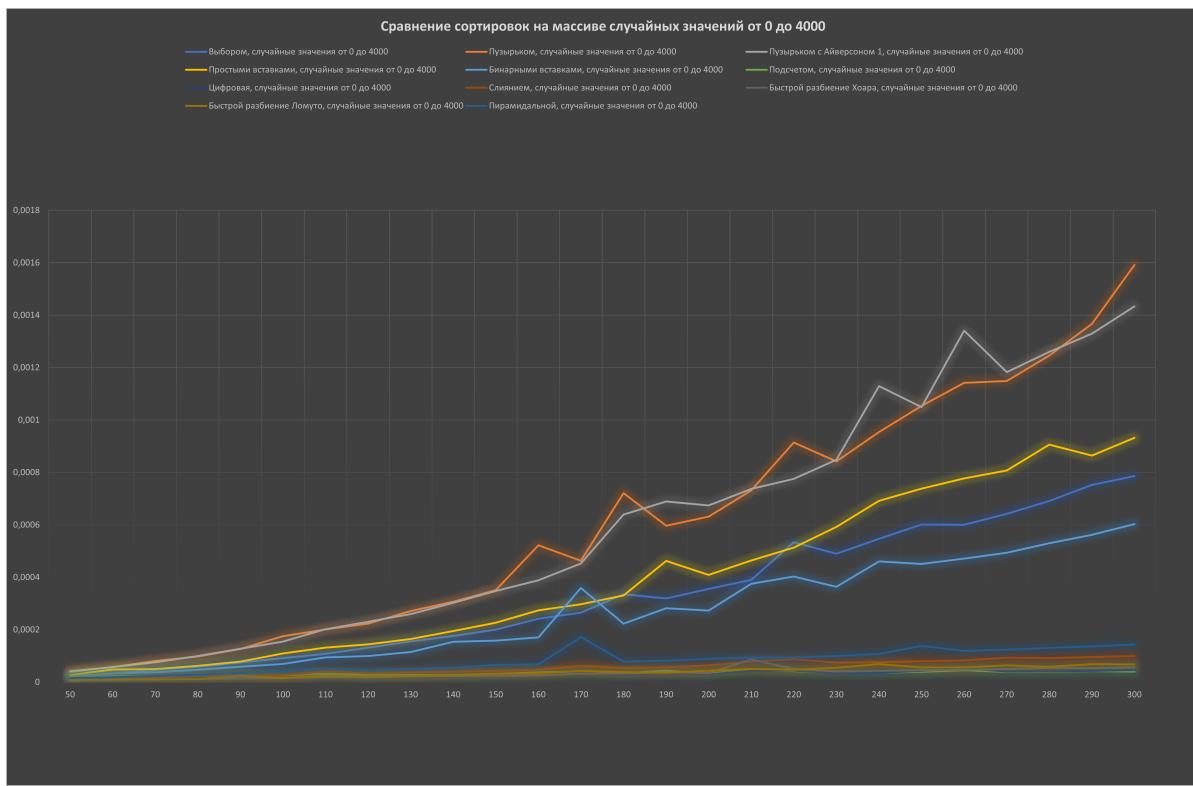


Рис. 17: Сравнения сортировок для массива со случайными значениями от 0 до 4000

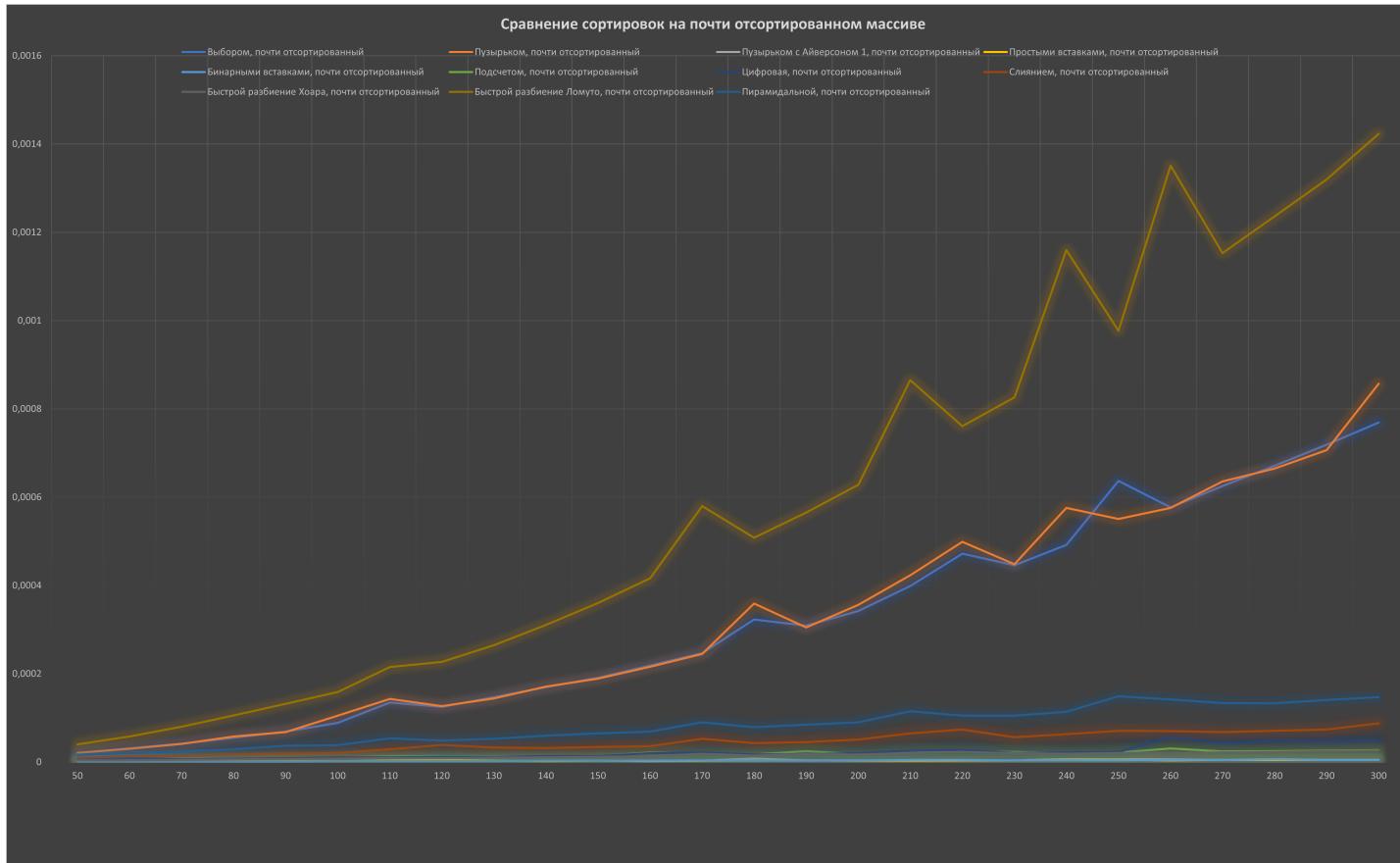


Рис. 18: Сравнение сортировок для почти отсортированного массива

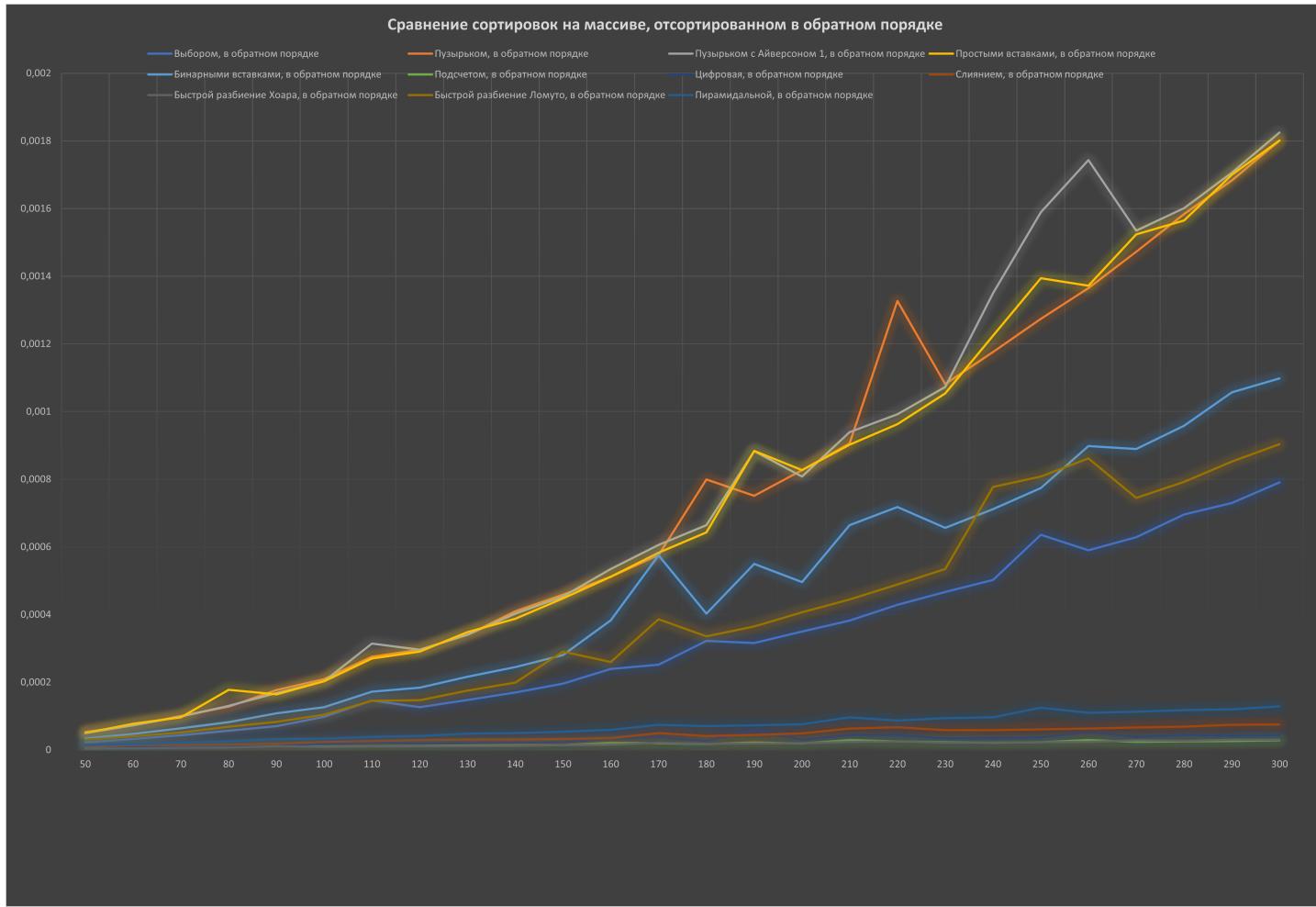


Рис. 19: Сравнение сортировок для массива, отсортированного в обратном порядке

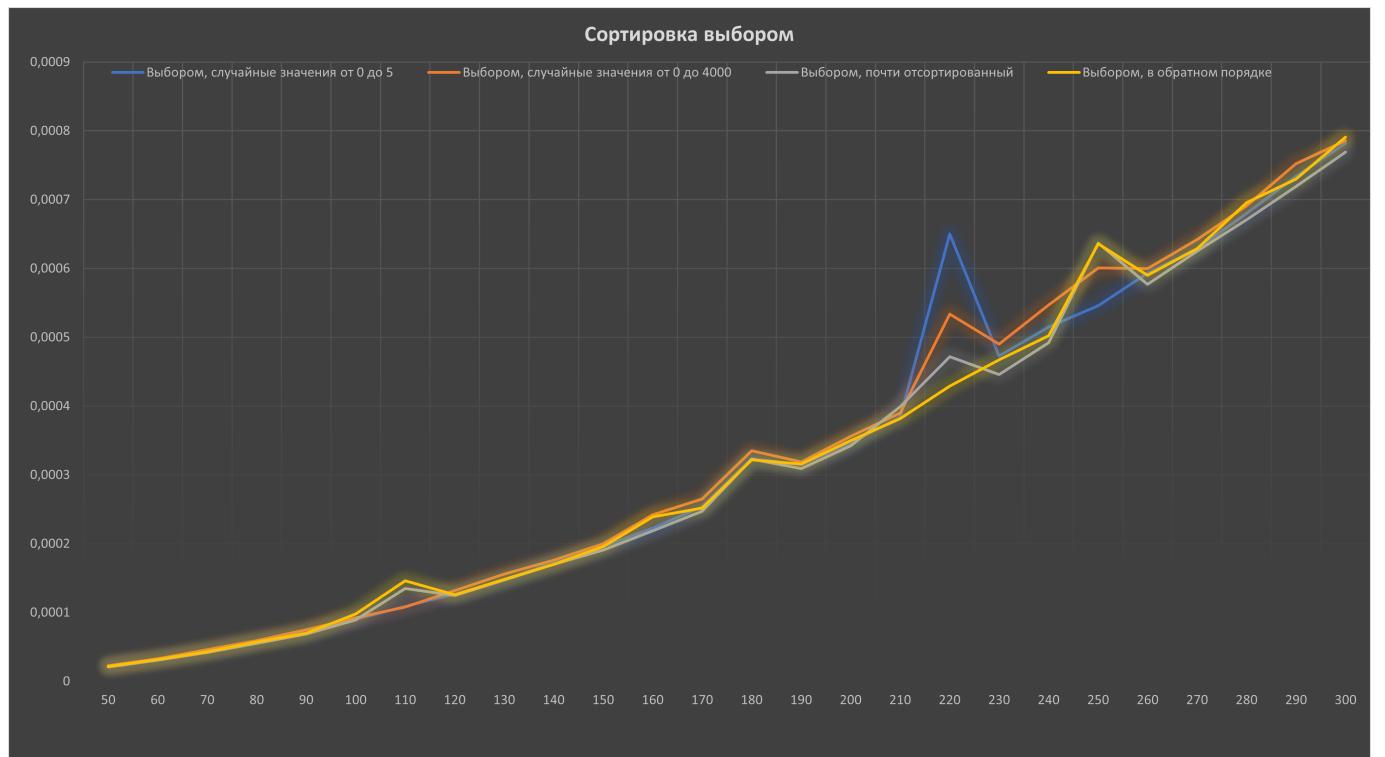


Рис. 20: Сравнение скорости сортировки выбором для различных видов массивов

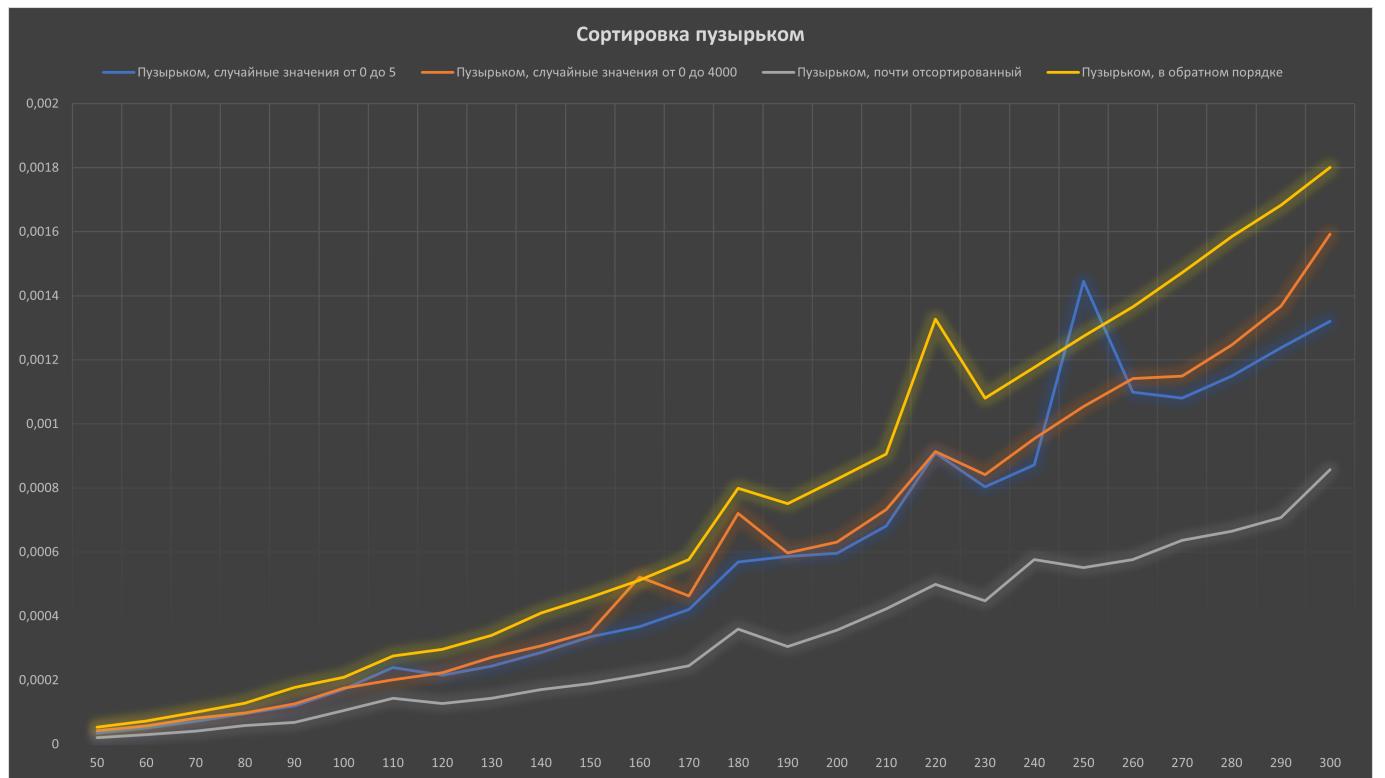


Рис. 21: Сравнение скорости сортировки пузырьком для различных видов массивов

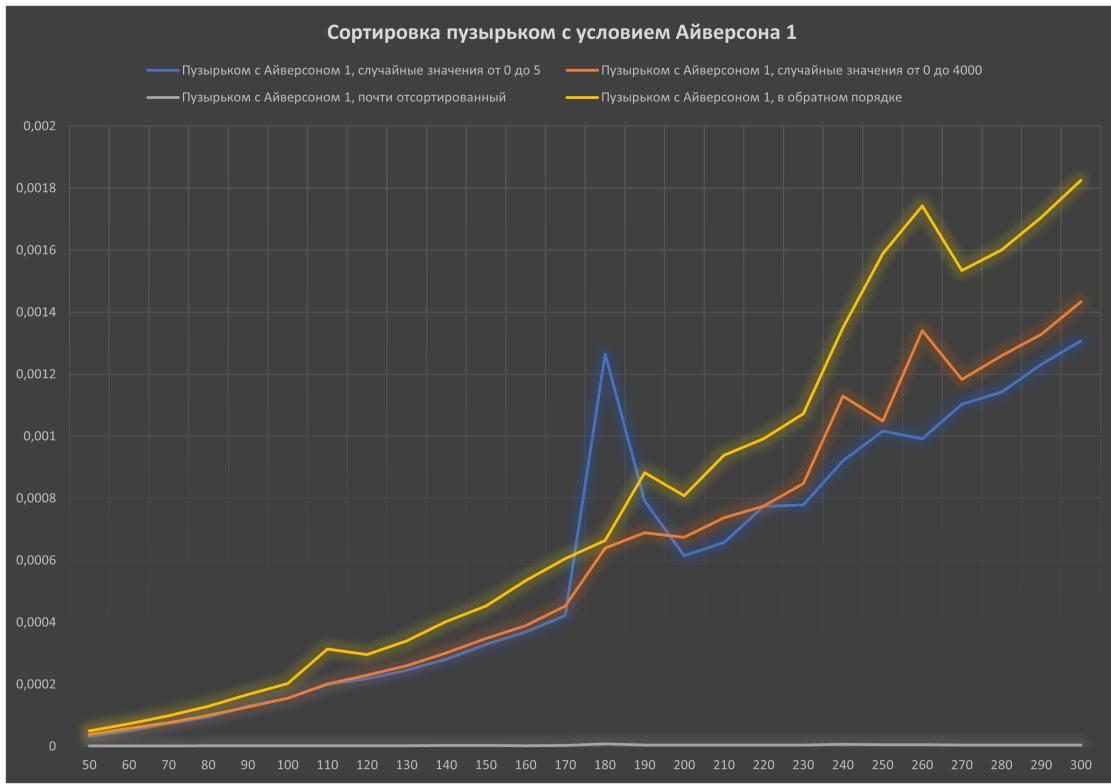


Рис. 22: Сравнение скорости сортировки пузырьком с условием Айверсона 1 для различных видов массивов

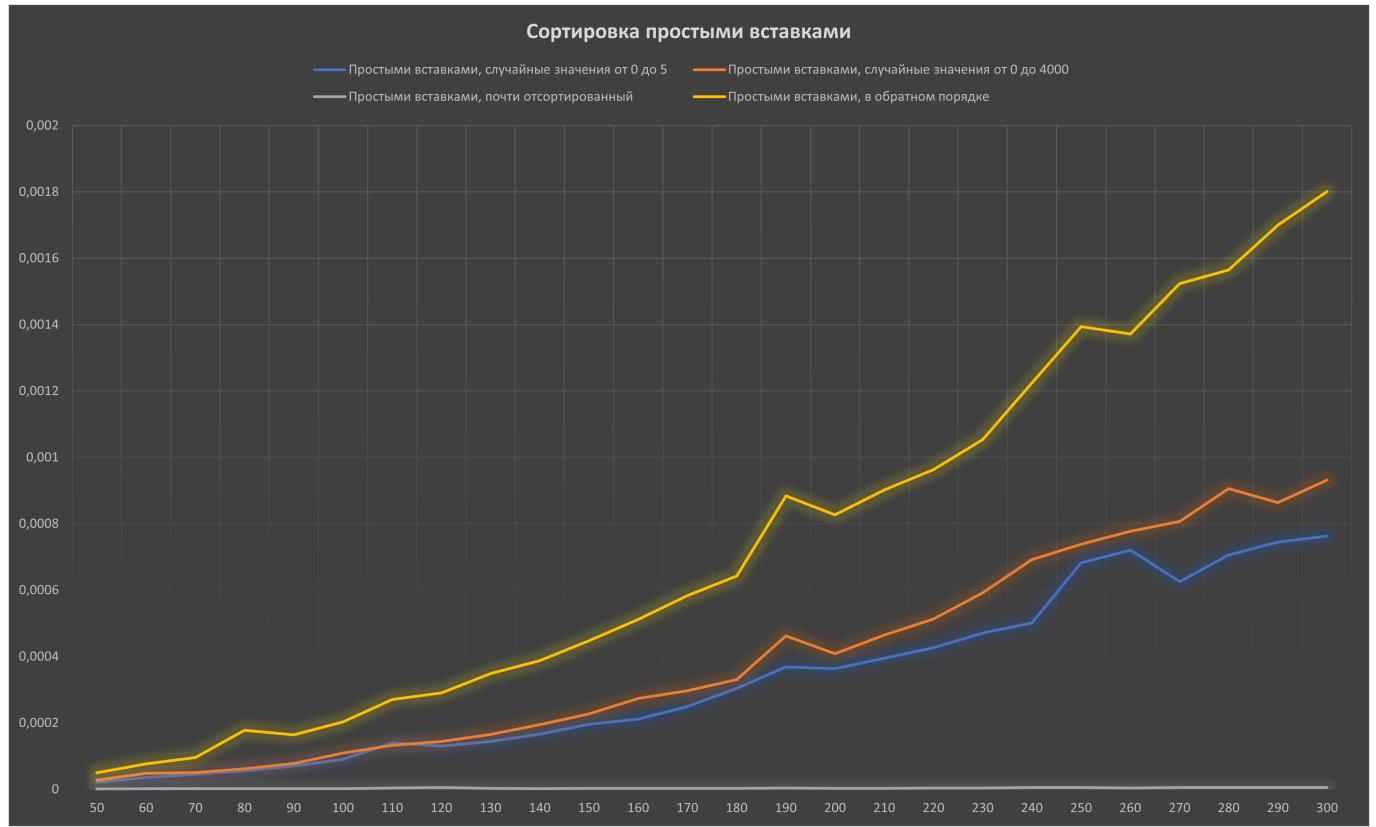


Рис. 23: Сравнение скорости сортировки простыми вставками для различных видов массивов

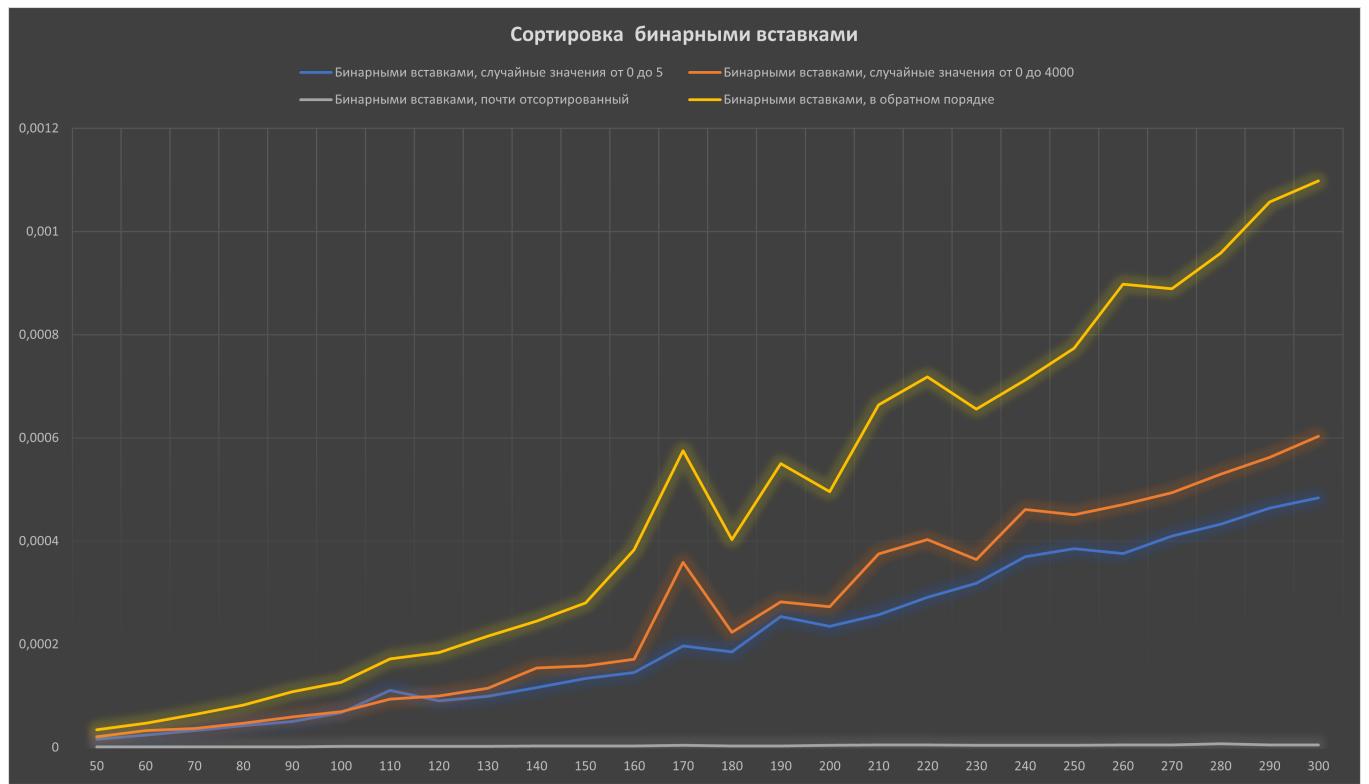


Рис. 24: Сравнение скорости сортировки бинарными вставками для различных видов массивов

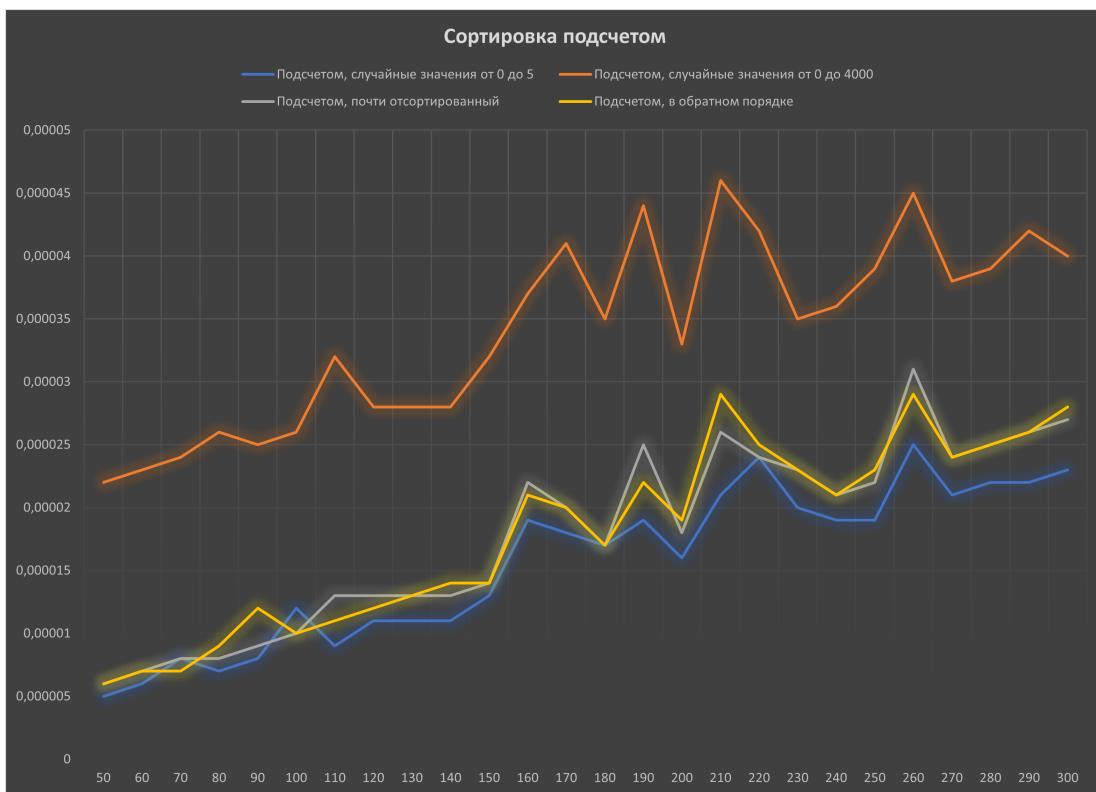


Рис. 25: Сравнение скорости сортировки подсчетом для различных видов массивов

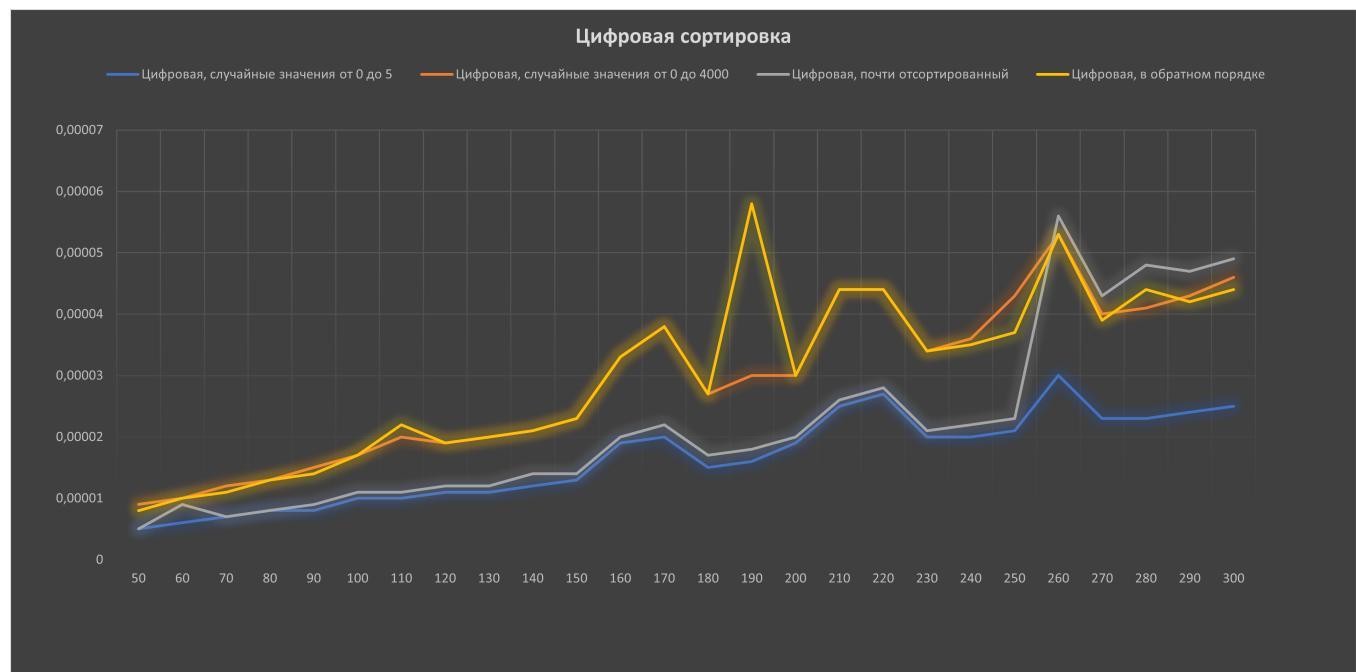


Рис. 26: Сравнение скорости цифровой сортировки для различных видов массивов

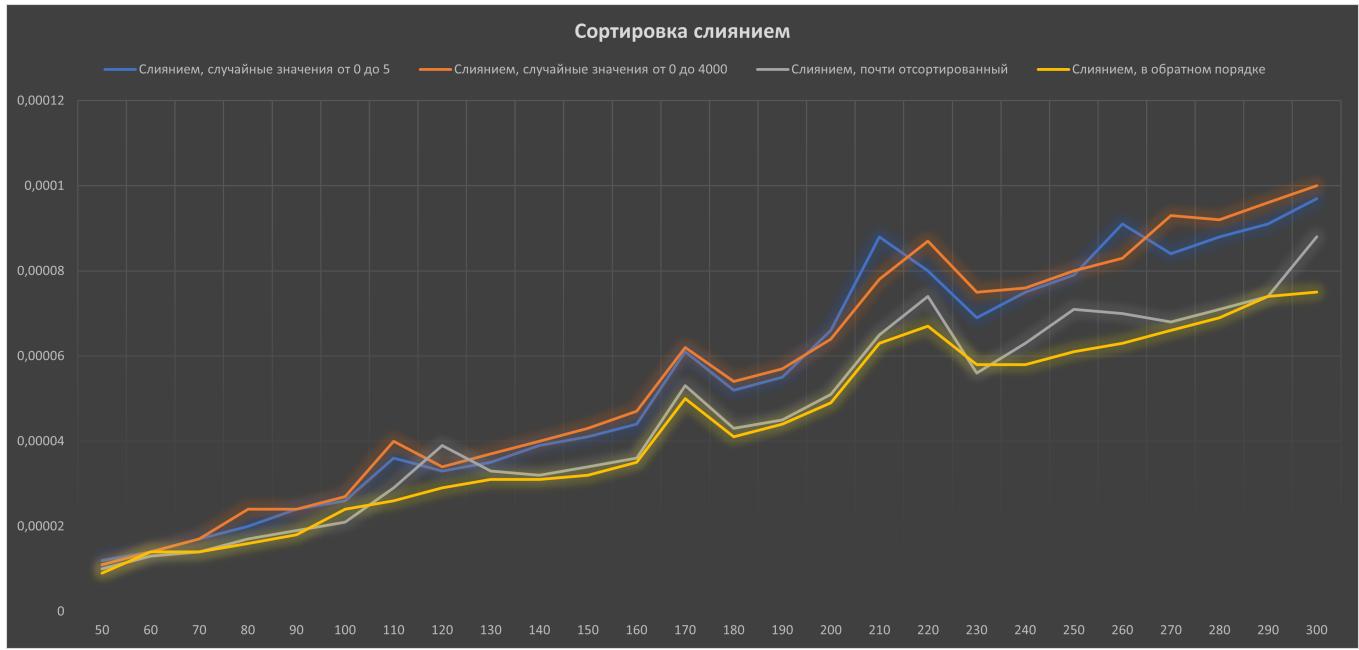


Рис. 27: Сравнение скорости сортировки слиянием для различных видов массивов

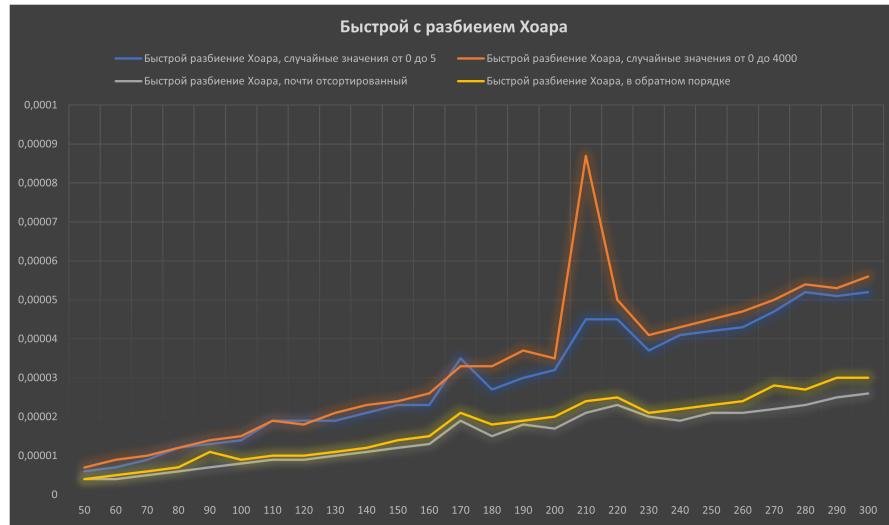


Рис. 28: Сравнение скорости быстрой сортировки соазбиением Хоара для различных видов массивов

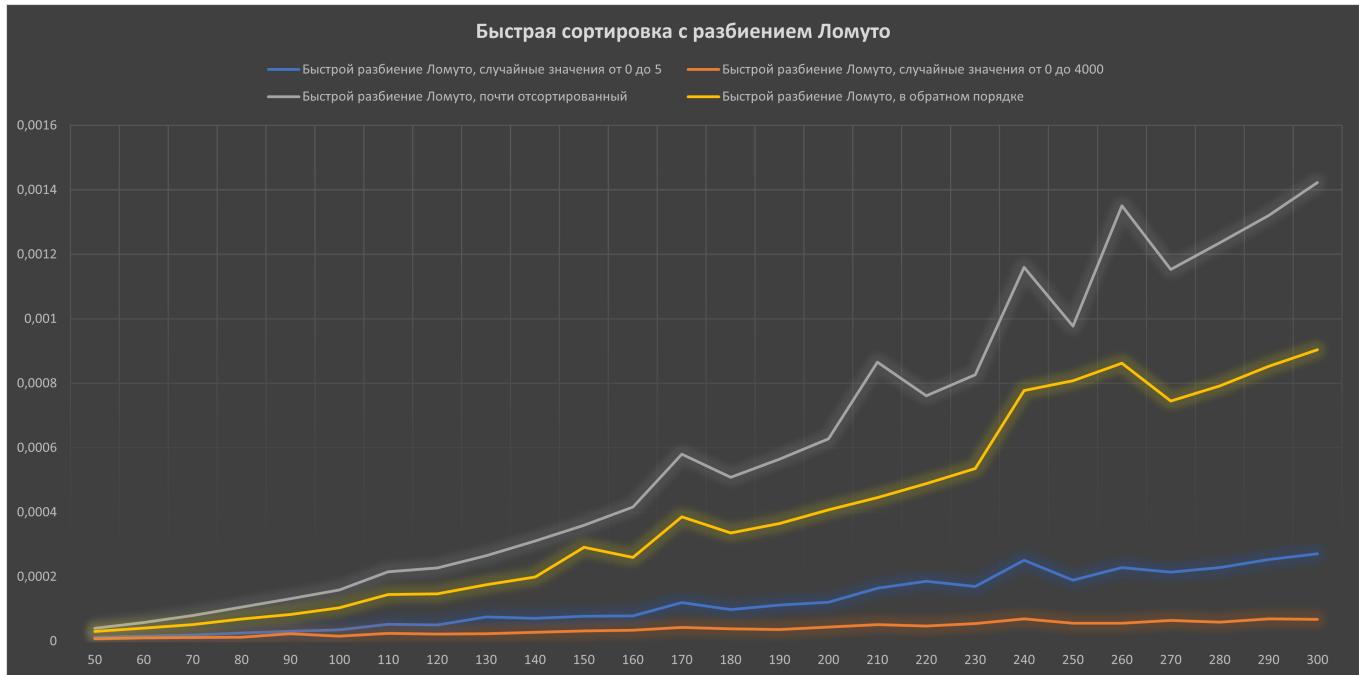


Рис. 29: Сравнение скорости быстрой сортировки с разбиением Ломуто для различных видов массивов

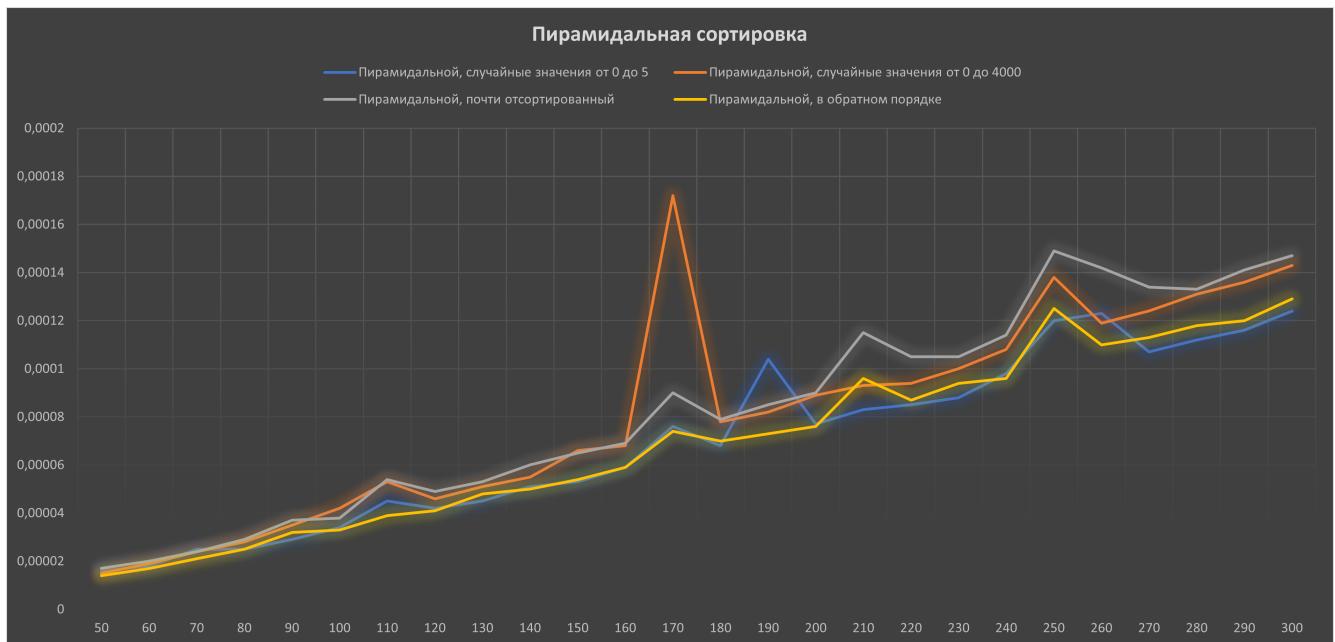


Рис. 30: Сравнение скорости пирамидальной сортировки для различных видов массивов