

УДК 004.633.2-047.37

108. ИССЛЕДОВАНИЕ БЫСТРОГО АЛГОРИТМА СОРТИРОВКИ БОЛЬШИХ ДАННЫХ

Король А.В., Житко Р.В., студенты гр.573904

*Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь*

Кириенко Н.А. – канд. тех. наук, доцент

Аннотация. В работе представлено экспериментальное исследование 13 стратегий выбора опорного элемента в алгоритме быстрой сортировки Хоара. Протестировано быстроедействие методов на массивах различного размера (20 000–100 000 элементов) и четырёх степеней упорядоченности. Построены отдельные графики для случайных данных по кластерам эффективности и для всех типов данных с индивидуальным масштабом. Установлено, что медиана трёх и адаптивный метод обеспечивают оптимальный баланс скорости и устойчивости.

Ключевые слова. Быстрая сортировка, алгоритм Хоара, опорный элемент, точка разбиения, стратегии выбора, анализ производительности, медиана.

Алгоритм сортировки Хоара является одним из наиболее широко применяемых алгоритмов упорядочивания данных в современных программных системах. Среднее время работы алгоритма составляет $O(n \cdot \log_2 n)$. Несмотря на теоретическую эффективность, практическое быстроедействие алгоритма принципиально зависит от одного ключевого решения – выбора опорного элемента, определяющего точку разбиения подмассива на каждом шаге рекурсии. При неудачном выборе опорного элемента алгоритм снижает скорость своей работы до $O(n^2)$. Цель работы: экспериментально определить и обосновать наиболее эффективную стратегию выбора опорного элемента в алгоритме быстрой сортировки применительно к массивам различных типов и размеров. Для достижения данной цели была создана программа на языке C++, в которой рассматривались 13 вариантов выбора опорного элемента, различные массивы по степени упорядоченности (почти/обратно отсортированные, отсортированные и случайные данные), различные по количеству элементов массивы (от 20 000 элементов до 100 000).

Была изучена эффективность выбора следующих точек разбиения быстрой сортировки: первого элемента; среднего элемента; последнего элемента; случайного элемента. Также в качестве точек разбиения были рассмотрены различные медианы. «Медиана трёх» представляет собой число, которое не является ни самым маленьким, ни самым большим среди первого, последнего и среднего элементов [1]. Медиана первой половины – медиана первого, четвертного от начала и среднего элементов, медиана второй половины – медиана среднего, четвертного от конца и последнего элементов. Медиана пяти элементов – средний по значению элемент среди пяти равномерно распределённых в массиве элементов. Медиана медиан – медиана из групп медиан по пять элементов (массив делится на группы по пять элементов, находится медиана в каждой группе, затем находится медиана среди данных медиан). Ещё одна медиана, реализованная в программе, – медиана трёх случайных элементов массива. В качестве точек разбиения были рассмотрены и элементы, стоящие на позиции «одна треть массива» и «две трети массива». Ещё одна стратегия – адаптивный метод. Адаптивный метод выбора точки разбиения проверяет порядок трёх элементов массива, чтобы понять, не отсортирован ли он уже, и выбирает такой опорный элемент, который гарантирует быструю работу даже на почти упорядоченных данных. Если три контрольных элемента (левый, средний, правый) упорядочены по возрастанию, возвращается средний; если левый элемент по своему значению находится между средним и правым, возвращается левый; если правый элемент между левым и средним, возвращается правый; в остальных случаях возвращается элемент из середины массива [2].

Исследование строилось в два этапа. На первом – все 13 стратегий тестировались исключительно на случайных данных, что позволило оценить чистые накладные расходы механизма (дополнительная работа, которую алгоритм выполняет не ради основной задачи, а ради вспомогательной) выбора точки разбиения. Размеры массивов: от 20 000 до 100 000 элементов с шагом 10 000. Для каждого размера и каждой из 13 стратегий замерялось время выполнения в миллисекундах.

Проведя исследование на случайных данных, можно разбить стратегии на две группы – эффективные и неэффективные. Наиболее неэффективная точка разбиения – медиана медиан: при 100 000 элементах она тратит 49,9 мс, что в 4,5 раза медленнее остальных. Метод

работает рекурсивно, из-за чего и увеличивается время работы алгоритма. Медиана трёх случайных элементов и медиана пяти работают медленнее из-за дополнительных вычислений при каждом выборе. Случайный выбор проигрывает по другой причине: программа постоянно обращается к генератору случайных чисел, а это «дорогая» операция.

На рисунке 1 показаны остальные девять эффективных для случайных данных стратегий. Графики данных стратегий практически «сливаются в пучок» поэтому принципиальных отличий для выбора опорного элемента на случайных данных нет. Однако на массиве в 80000 элементов медиана половины сильно замедляет свою работу, а на массиве в 90000 элементов замедляет свою работу опорный первый элемент.

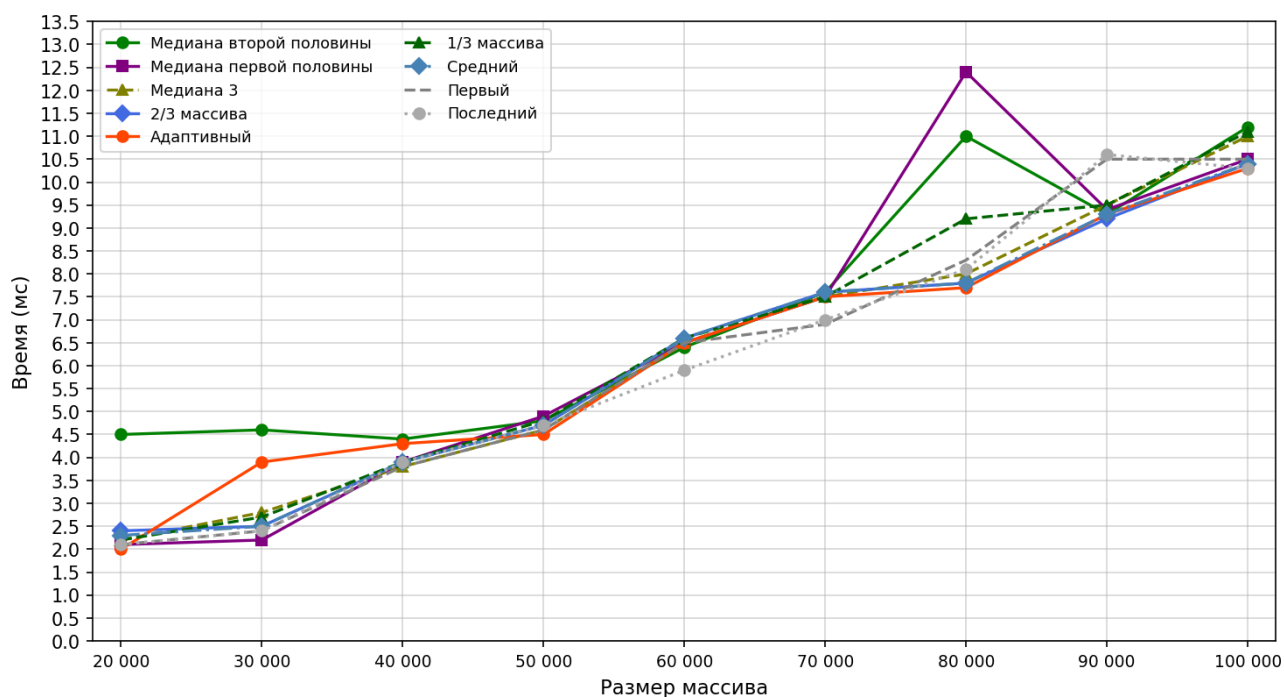


Рисунок 1 – Эффективные стратегии на случайных данных

На втором этапе исследования поведение каждой группы стратегий анализировалось на всех четырёх типах данных (полностью/почти/обратно отсортированных и случайных) для массива. На рисунке 2 представлен график стратегий, которые замедляют скорость работы на всех типах данных. На отсортированных данных происходит замедление работы стратегии первого элемента (до 5676 мс) и последнего элемента (до 4724 мс). Такое же замедление происходит для данных стратегий и на обратных и почти отсортированных данных.

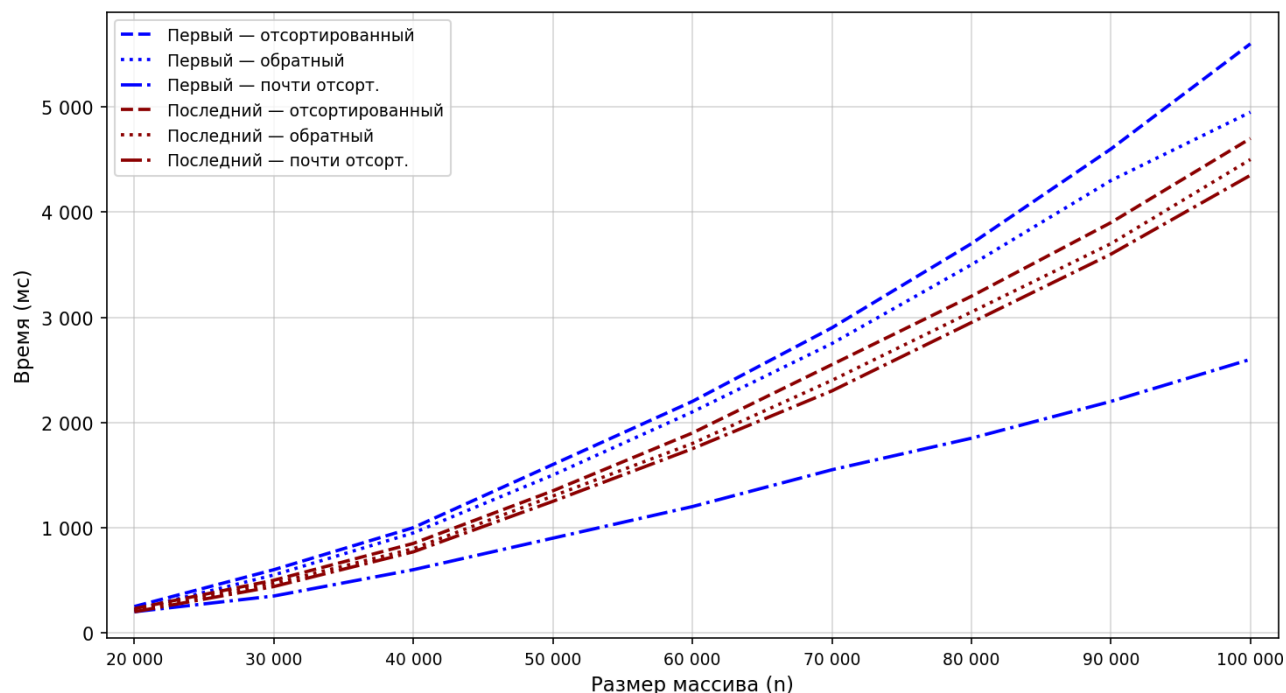


Рисунок 2 – Скорость стратегий первого и последнего элемента на различных типах данных

На рисунке 3 представлены оптимальные стратегии по результатам второго исследования, это средний элемент, медиана первого, последнего и среднего элемента, элемент, стоящий на позиции «одна треть массива» и адаптивный способ выбора точки разбиения. На случайных данных все четыре стратегии укладываются в 10–11 мс. На отсортированных – ускоряются до 1,9–2,4 мс.

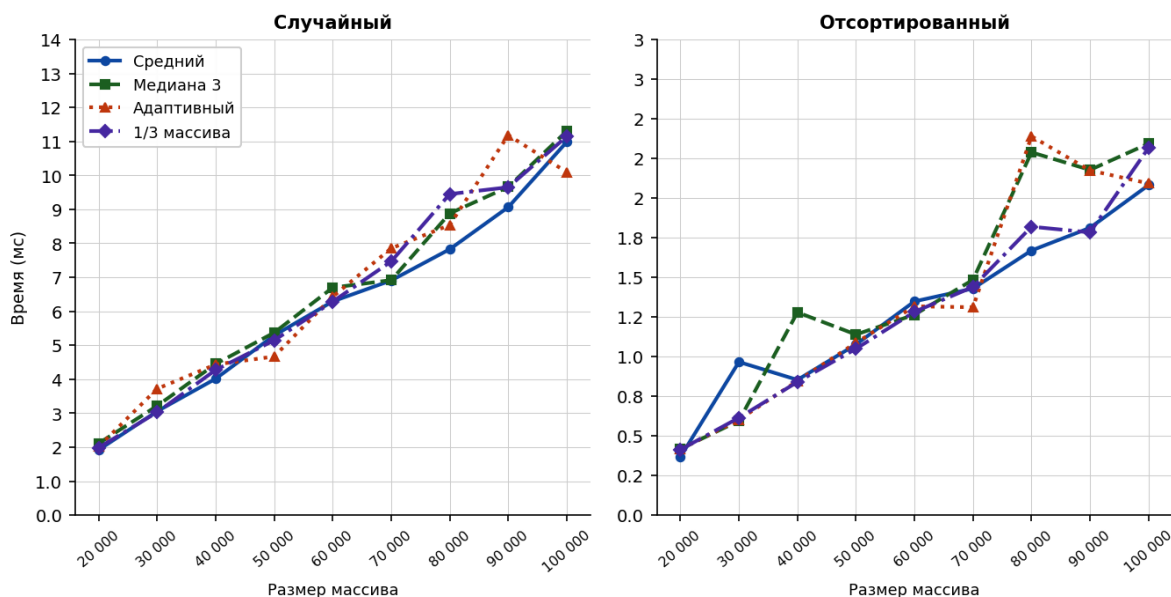


Рисунок 3 – Оптимальные стратегии для случайных и отсортированных типов данных

На рисунке 4 показаны средние стратегии для выбора по результатам второго исследования: медиана трех случайных элементов, медиана пяти элементов и случайный элемент в качестве точки разбиения. Данные стратегии устойчивы к любым входным данным и не испытывают критического замедления. На случайных данных они медленнее оптимальных, на упорядоченных данных наблюдается значительное ускорение: случайный элемент – до 5-6 мс, медиана 5 элементов – менее 4 мс, медиана трех случайных элементов – 9 мс. Это объясняется тем, что на структурированных данных случайный элемент статистически чаще оказывается ближе к медиане подмассива, чем на полностью хаотичных данных.

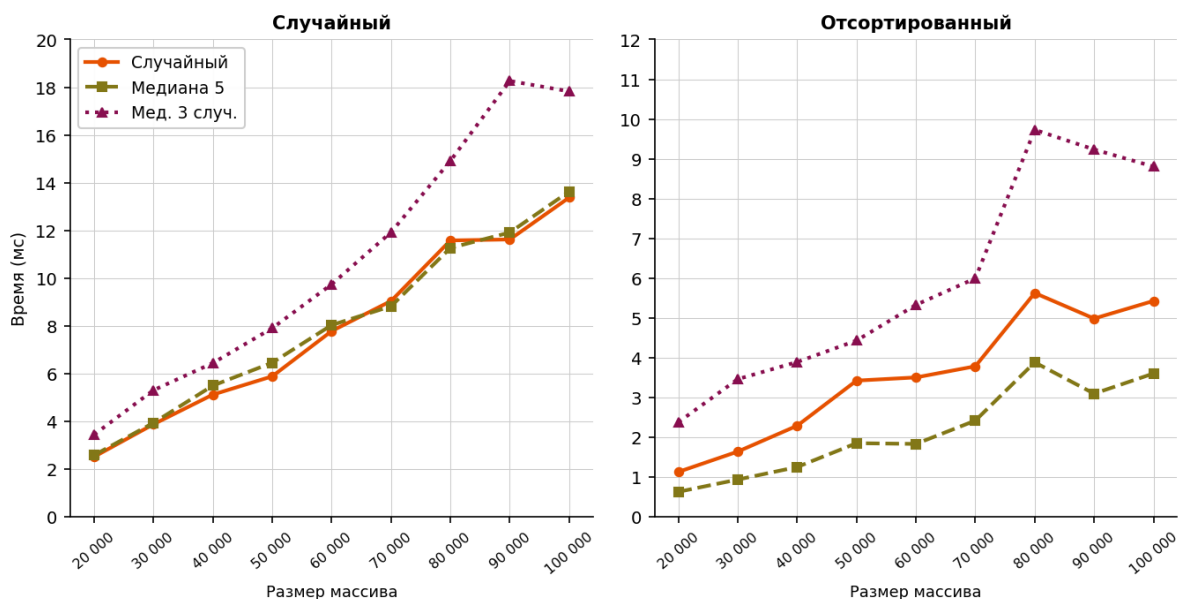


Рисунок 4 – Стратегии среднего уровня для случайных и отсортированных типов данных

Рисунок 5 сравнивает медиану медиан и средний элемент, чтобы показать, насколько теоретические гарантии могут расходиться с реальной производительностью, и полностью доказать неэффективность метода медианы медиан. Разрыв оказывается огромным. На случайных данных медиана медиан тратит 49,9 мс против 11,0 мс у среднего – почти в 5 раз медленнее. На отсортированных результаты становятся ещё хуже: средний элемент ускоряется до 2,1 мс, а медиана медиан замедляется до 188,8 мс – разница в 90 раз.

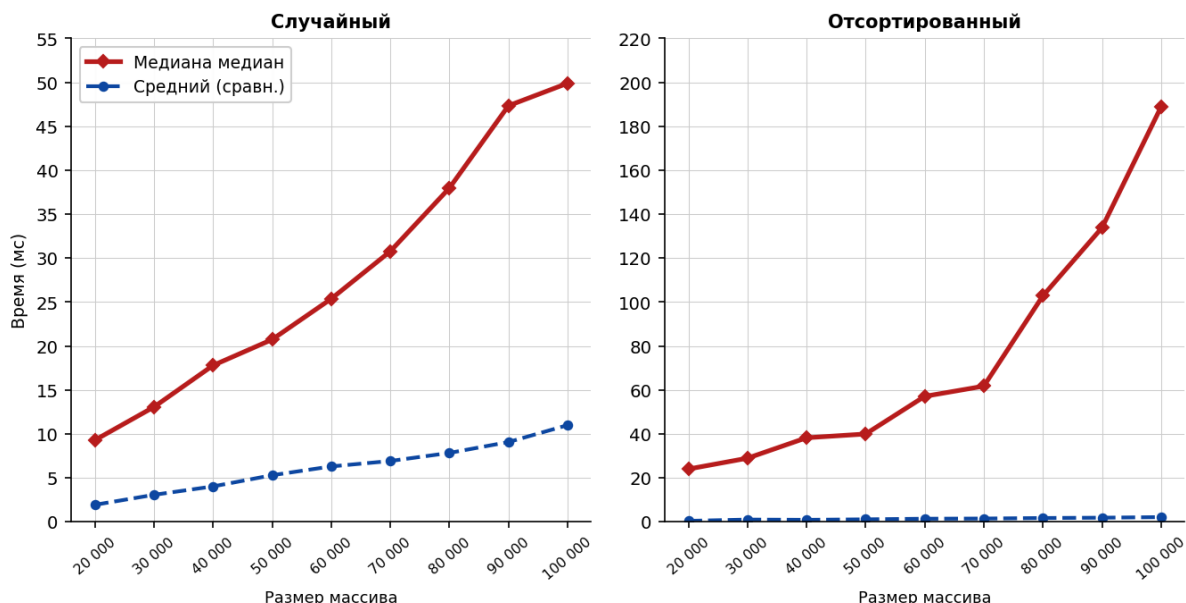


Рисунок 5 – Сравнение эффективности медианы медиан и среднего элемента

По итогам исследования можно дать конкретные рекомендации. Лучший выбор для большинства задач – медиана трёх (первого, среднего и последнего элементов) и адаптивный метод. На случайных данных они работают примерно за 11 мс при 100 000 элементов, а на уже отсортированных даже ускоряются до 2 мс. Главное их преимущество: они никогда не замедляются при других типах массивов, при этом почти не тратят время на выбор опорного элемента. Это делает их надёжным вариантом на практике, когда заранее не известно, какие данные придут на вход.

Допустимо использование среднего элемента, элемента на позиции «одна треть массива» и «две трети массива». Они не тратят время на вычисление опорного элемента вообще – просто берут

фиксированную позицию – и на упорядоченных данных работают так же хорошо, как медиана трёх и адаптивный метод. Риска деградации скорости нет, но они чуть менее универсальны: хорошо подходят там, где структура данных известна заранее.

Не рекомендуется использовать первый и последний элементы в качестве опорных. На отсортированных и почти отсортированных данных данные стратегии замедляются в 450-540 раз. Также не рекомендуется использовать медиану медиан. На практике стратегия уступает всем остальным стратегиям как на случайных, так и на упорядоченных данных из-за высоких накладных расходов на рекурсивное вычисление опорного элемента.

Список использованных источников:

1. Быстрая сортировка [Электронный ресурс]. – Режим доступа: <https://en.wikipedia.org/wiki/Quicksort> – Дата доступа: 05.03.2026.
2. Сортировка Хоара и другие способы сортировки массивов [Электронный ресурс]. – Режим доступа: https://otus.ru/journal/sortirovka-hoara-i-dругие-sposoby-sortirovki-massivov/#%D0%92%D1%8B%D0%B1%D0%BE%D1%80_%D0%BE%D0%BF%D0%BE%D1%80%D0%BD%D0%BE%D0%B3%D0%BE_%D1%8D%D0%BB%D0%B5%D0%BC%D0%B5%D0%BD%D1%82%D0%B0 – Дата доступа: 05.03.2026.

UDC 004.633.2-047.37

STUDY OF A QUICK BIG DATA SORTING ALGORITHM

Korol A.V., Zhitko R.V.

Belarusian State University of Informatics and Radioelectronics, Minsk, Republic of Belarus

Kirienko N.A. – candidate of Technical Sciences, associate professor

Annotation. This paper presents an experimental study of 13 pivot selection strategies in the Hoare quicksort algorithm. The methods' performance was tested on arrays of varying sizes (20,000–100,000 elements) and four degrees of ordering. Separate plots were constructed for random data by efficiency clusters and for all data types with individual scales. It was found that the median-of-three and adaptive methods provide the optimal balance of speed and robustness.

Keywords. Quicksort, Hoare sort, pivot, partition point, selection strategies, performance analysis, median.