Результаты работы алгоритма сортировки “sample sort”.

* 1. Неупорядоченный набор чисел:

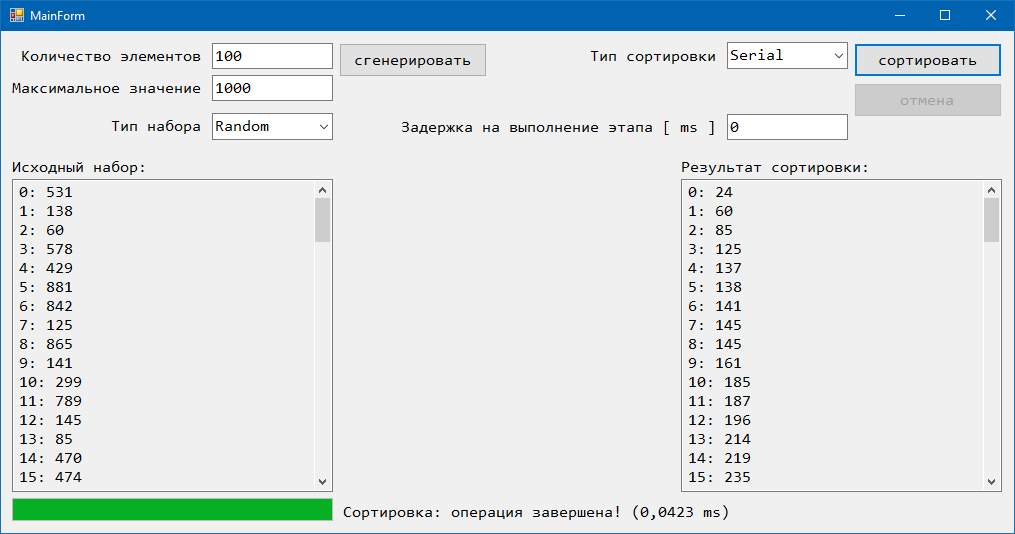


Рисунок №1.1 – последовательная обработка набора чисел (число элементов = 100)

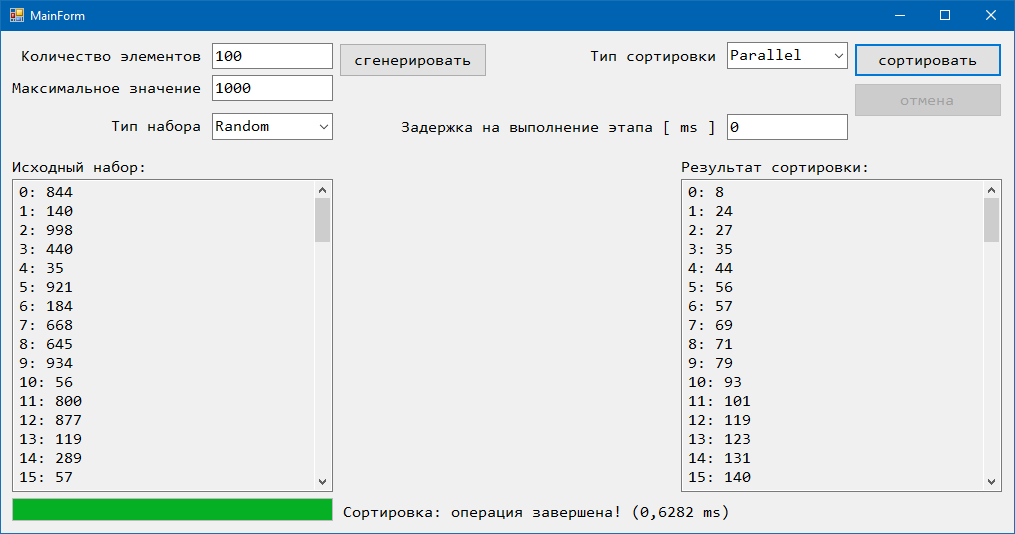


Рисунок №1.2 – параллельная обработка набора чисел (число элементов = 100)

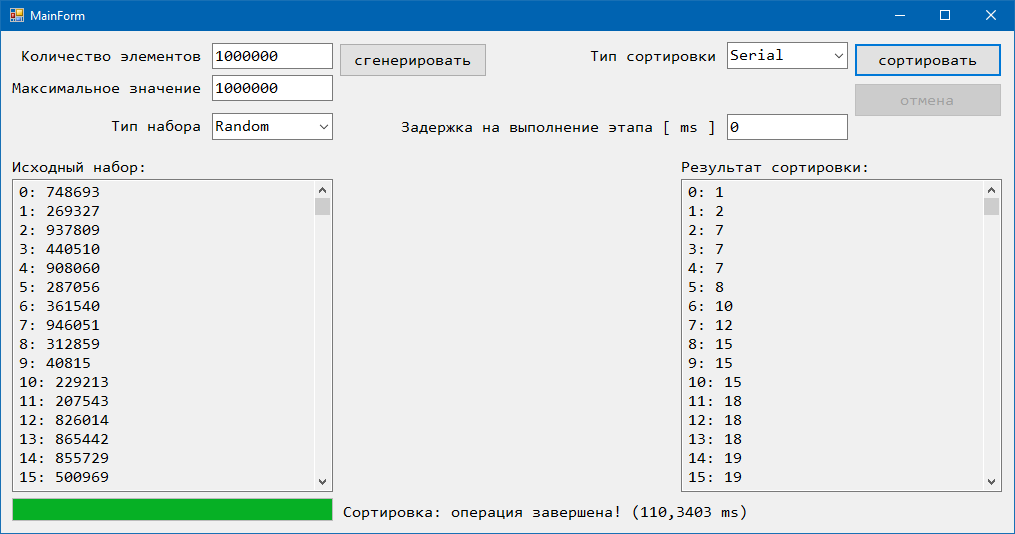


Рисунок №1.3 – последовательная обработка набора чисел (число элементов = 1 000 000)

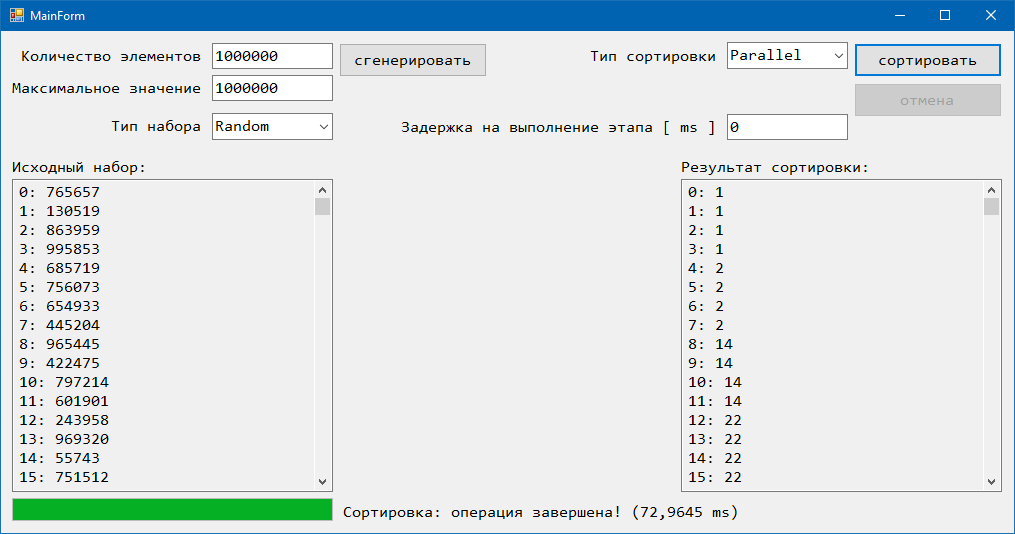


Рисунок №1.4 – параллельная обработка набора чисел (число элементов = 1 000 000)

При увеличении числа сортируемых элементов наблюдается выигрыш при использовании параллельной сортировки около 25% по времени выполнения. Среднее время выполнения при параллельной сортировки 80 мс, при последовательной 115 мс.

* 1. Упорядоченный набор чисел (по возрастанию):

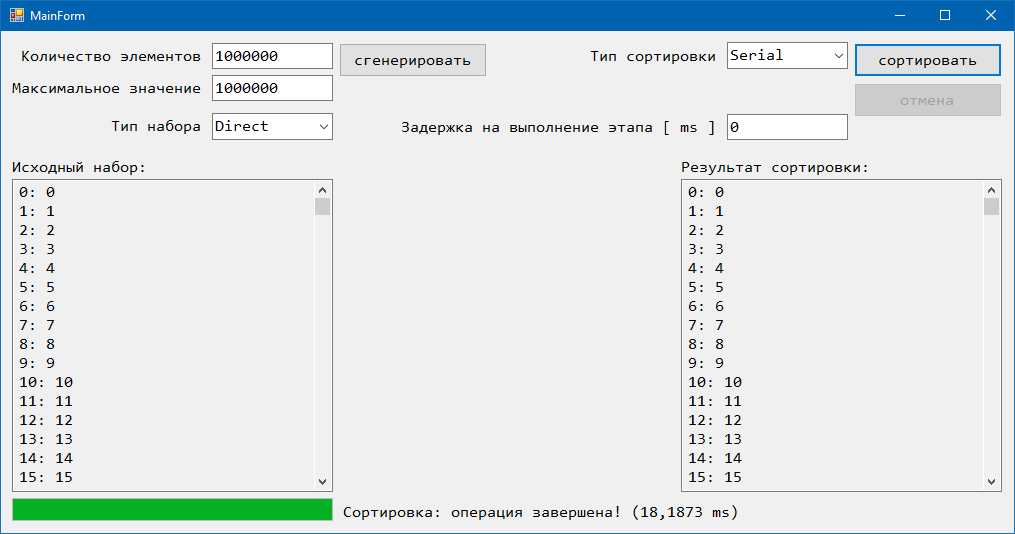


Рисунок №2.1 – последовательная обработка набора чисел (число элементов = 1 000 000)

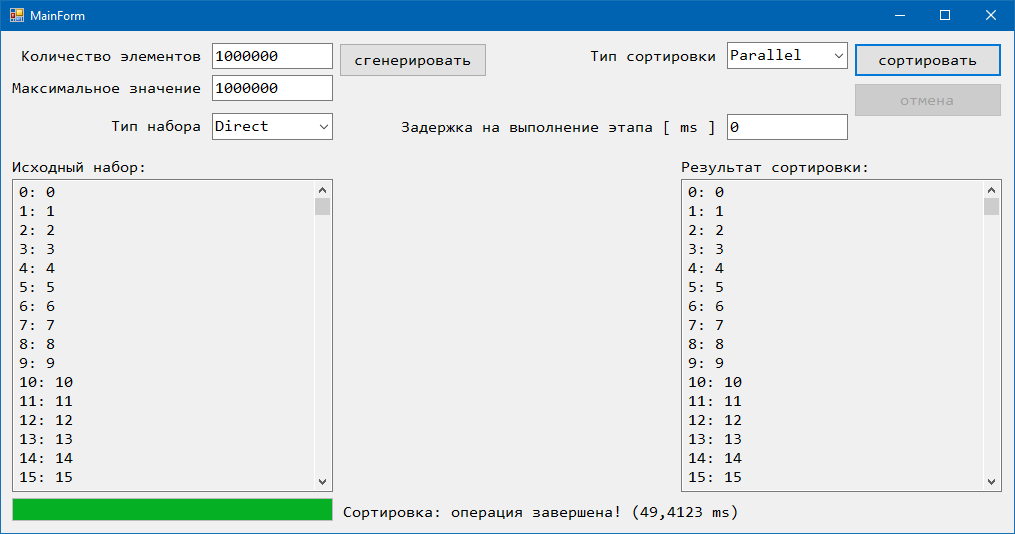


Рисунок №2.2 – параллельная обработка набора чисел (число элементов = 1 000 000)

Последовательная сортировка набора упорядоченных чисел статистически выполняется быстрей. Среднее время выполнения при параллельной сортировки 45 мс, при последовательной 20 мс.

PS: в алгоритм можно добавить проверку на упорядоченность набора.

* 1. Упорядоченный набор чисел (по убыванию):

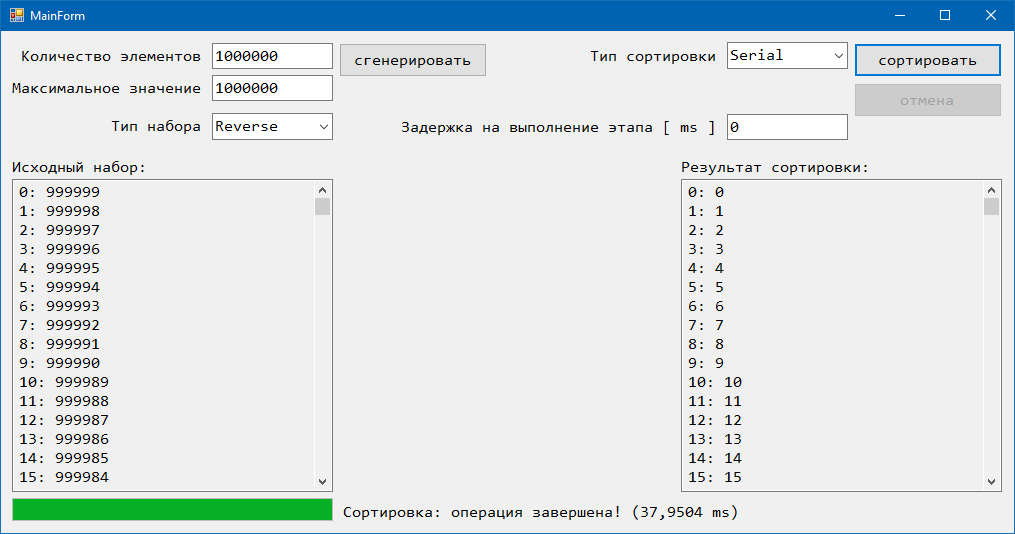


Рисунок №3.1 – последовательная обработка набора чисел (число элементов = 1 000 000)

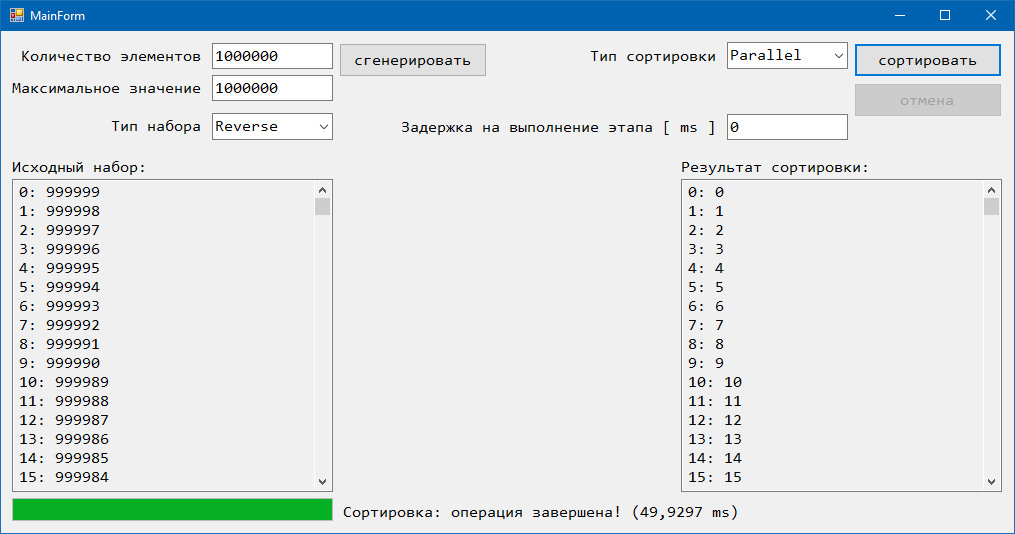


Рисунок №3.2 – параллельная обработка набора чисел (число элементов = 1 000 000)

Результаты обработки упорядоченного набора чисел по убыванию в среднем получились примерно одинаковыми. Среднее время выполнения при параллельной сортировки 45 мс, при последовательной 45 мс.