

Sortowanie	Dane losowe	Tablica posortowana	Tablica odwrotnie posortowana
QuickSort	3.0439(s)	-	-
HeapSort	5.4194(s)	0.5260(s)	0.5278(s)
MergeSort	3.7786(s)	3.2198(s)	3.2980(s)
BubbleSort	214.5654(s)	211.5164(s)	214.4846(s)

Dla wszystkich algorytmów została podana próbka 500000 jedynie dla bubblesorta została zmniejszona do 50000 z powodu zbyt długiego czasu pracy i braku odpowiedzi PyCharm. Nie udało się jednak znaleźć rozwiązania dla dwóch pozostałych tablic w przypadku QuickSorta, każda próba kończyła się błędem „Process finished with exit code -1073741571 (0xC00000FD)”

Algorytmy z złożonością czasową $O(n \log n)$ takie jak merge sort czy heap sort radzą sobie znacznie lepiej od tych z $O(n^2)$ (bubblesort). Mergesort posortował szybciej tablice losową niż heapsort jednak w przypadku tablicy posortowanej i odwrotnie posortowanej lepiej zadziałał Heapsort