Temat N 2. Badanie efektywności szybkich algorytmów sortowania

Treść zadania.

1. Zapoznać się dokładnie z szybkimi algorytmami sortowania: qsort (sortowanie szybkie), sortowanie przez scalanie, sortowanie przez kopcowanie, sortowania Shella, sortowanie Batchera (bitoniczne) z ocenami złożoności łącznie.
2. Zbadać relacje pomiędzy ww. algorytmami a algorytmami opracowanymi w zadaniu nr 1. Innymi słowy, wykonać klasyfikację względem trzech podstawowych technik sortowania z zadania 1.
3. Określić najlepsze i najgorszy warianty rozkładu danych dla szybkich algorytmów z punktu 1. Zbadać eksperymentalnie faktyczną złożoność czasową optymistyczna, średnią, pesymistyczną oraz porównać z odpowiednia złożonością teoretyczną.
4. Przemyśleć zasadność ulepszenia algorytmu przez połączenia dwóch podstawowych technik budowy algorytmów sortowania.
5. Zaproponować Autorski algorytm szybkiego sortowania, który jest lepszym pod jakimś względem, np. a) przy najgorszym rozkładzie danych potrzebuje najmniej porównań; b) liczba przestawień danych jest najmniejszą.
6. Napisać sprawozdanie z przeprowadzonych badań.

Termin realizacji (łącznie z sprawozdaniem): 14.12.2016.

Literatura:

1. Knuth D.E. Sztuka programowania. T.3. Sortowanie i wyszukiwanie. WNT, Warszawa 2002.
2. Sedgewick R. Algorytmy w C++. Wydawnictwo RM, Warszawa 1999.
3. Wróblewski P. Algorytmy, struktury danych i techniki programowania, Helion, Gliwice, 2003.
4. Kotowski P. Algorytmy + Struktury danych = Abstrakcyjne Typy Danych, Wydawnictwo BTC, Warszawa, 2006.
5. Internet.