

II INFORMATYKA

Algorytmy i struktury danych

Lista nr 4

Rozbuduj program z Listy nr 3, uzupełniając go o funkcję sortującą n liczb metodą scalania.

Podobnie jak poprzednio, należy przewidzieć możliwość sortowania długich (np. 100 000 000 elementów) ciągów (o wartościach losowych i specyficznych - np. podawanych z klawiatury). Dla algorytmu 5 należy także obliczać t - liczbę wykonanych porównań pomiędzy elementami sortowanego ciągu (typu $a_i < a_j$; te porównania są wykonywane w procedurze SCAL) oraz wartość ilorazu $c = \frac{t}{n \log n}$. Przetestować działanie programu dla ciągów różnych długości. Należy prowadzić pomiar rzeczywistego czasu działania wszystkich algorytmów. Sprawdzić doświadczalnie (sortując wiele ciągów losowych różnych długości), jaka jest średnia wartość czynnika c .

Który z czterech algorytmów (sortowanie przez wstawianie, własną metodą, kopcowanie oraz sortowanie przez scalanie) ma najlepszą złożoność czasową? Odpowiedź uzasadnij porównując czasy działania trzech algorytmów (i/lub liczbę wykonanych porównań pomiędzy elementami sortowanych ciągów).

Czy znając czas działania algorytmu 5 dla ciągu długości 100 000 o wartościach losowych będziesz umiał przewidzieć (szybko obliczyć) czas działania tego algorytmu dla ciągu długości 10 000 000 o wartościach losowych?