

II INFORMATYKA

Algorytmy i struktury danych

Lista nr 3

Zaproponuj nierekurencyjną implementację procedury H-PRZYWRACANIE dla tablicowej reprezentacji kopca

Rozbuduj program z Listy nr 2, uzupełniając go o funkcję sortującą n liczb przez kopcowanie.

Podobnie jak poprzednio, należy przewidzieć możliwość sortowania długich (np. 300 000 000 elementów) ciągów (o wartościach losowych i specyficznych - np. podawanych z klawiatury). Dla zaimplementowanego algorytmu należy obliczać t - liczbę wykonanych porównań pomiędzy elementami sortowanego ciągu (typu $a_i < a_j$) oraz wartość ilorazu $c = \frac{t}{n \log n}$.

Program powinien dać możliwość racjonalnego sprawdzenia poprawności działania algorytmów poprzez wydrukowanie fragmentu tablicy (np. 50 początkowych elementów) przed i po sortowaniu.

Przetestuj działanie programu dla ciągów różnych długości. Sprawdź doświadczalnie (sortując wiele ciągów losowych różnych długości), jaka jest średnia wartość czynnika c .

Uzupełnij program o pomiar rzeczywistego czasu działania każdego algorytmu (w przeliczeniu na sekundy z dokładnością do drugiego miejsca po przecinku). Który z tych algorytmów (sortowanie przez wstawianie, własną metodą oraz przez kopcowanie) jest najszybszy? Odpowiedź uzasadnij porównując czasy działania algorytmów oraz liczbę wykonanych porównań pomiędzy elementami sortowanych ciągów. Czy istnieje jakiś związek pomiędzy wartościami czynnika c oraz różnicą w czasie działania algorytmu?

Na podstawie wiedzy o czasie działania algorytmu (1 lub 2 lub 3) dla ciągu długości 10000 o wartościach losowych, spróbuj przewidzieć czas działania tego algorytmu dla ciągów długości 100000 o wartościach losowych oraz dla ciągów o innych długościach.