

Laboratorium nr 4

Testowanie algorytmu sortowania przez scalanie i quicksort

Program ma na celu przetestowanie sortowania tablic metodą mergesort, a także quicksort. Tablice których rozmiar przekazywany był z napisanej na poprzednich zajęciach kolejki wypełniane były losowymi liczbami.

1. Wstęp teoretyczny

Quicksort jest to rekurencyjne sortowanie polegające na podziale tablicy na pomniejsze. Na początku obierana jest oś – element z tablicy, względem którego będą porównywane jej poszczególne elementy. Jeżeli jakiś element jest mniejszy od niej – idzie na lewo, jeżeli większy – na prawo. Następnie dzielona jest na dwie pomniejsze i w dokładnie ten sam sposób postępuje się z kolejnymi podzielonymi już tablicami. Rekurencja kończy się, gdy tablica zostanie podzielona na pojedyncze elementy. Średnia złożoność obliczeniowa wynosi $O(n \log n)$, natomiast w najbardziej pesymistycznym wariancie będzie ona wynosić $O(n^2)$.

Mergesort jest to również rekurencyjna metoda sortowania podobna do quicksorta. Różnica między nimi jest taka, że elementy sortuje się po rozbiciu ich na podstawowe elementy, w momencie scalania jej w jedną. Teoretyczna złożoność obliczeniowa wynosi $O(n \log n)$.

2. Tabele

Quicksort:

l. elementów	Pivot w środku	Pivot na początku
10	0,0000012	0,0000012
100	0,0000167	0,0000158
1000	0,0003233	0,0002687
10000	0,0012619	0,0012877
100000	0,0150255	0,0154789
1000000	0,17729	0,183869
10000000	2,00974	2,05225
100000000	21,3444	22,3292
1000000000	234,568	248,592

Mergesort:

l. elementow	Pivot w srodku	Pivot na poczatku	Pivot losowo
10	0,0000032	0,0000048	0,000005
100	0,0000566	0,0000436	0,0000432
1000	0,0005351	0,0002956	0,0001748
10000	0,0021459	0,0021485	0,0021271
100000	0,0253992	0,026133	0,0265194
1000000	0,291698	0,300619	0,29284
10000000	3,32493	3,22013	3,15741
100000000	34,9111	36,7302	35,3955
1000000000	x	x	x

3.Wnioski

Podane algorytmy posortowały dane w czasie $O(n \log n)$. Żaden z podanych algorytmów quicksort nie zbliżył się do pesymistycznego wariantu. Dzieje się tak, gdyż sortowane dane były całkowicie losowe, małe prawdopodobieństwo było wystąpienia takich liczb, które by były ustawione (w przypadku pivota ustawionego na skrajnym miejscu, a sortowana tablica byłaby już wcześniej posortowana, lub ustawiona w odwrotnej kolejności).

Mergesort pomimo teoretycznie takiej samej złożoności średnio wykonywał się dłużej. Zauważono natomiast, że pomimo różnego rodzaju ustanawiania pivota, czas nie zmieniał się znacząco. Jedyne odchył pojawił się dla 1000 elementów.

3. Wykresy

