

Sprawozdanie z laboratorium nr 4
„Quick sort”

1. Cel zajęć

- implementacja algorytmu quick sort
- zbadanie zależności $t(n)$, gdzie t - czas, n - ilość danych

2. Realizacja

Korzystając ze stworzonej wcześniej struktury danych „Array” (dynamicznie alokowane tablice), zaimplementowany został algorytm quick sort:

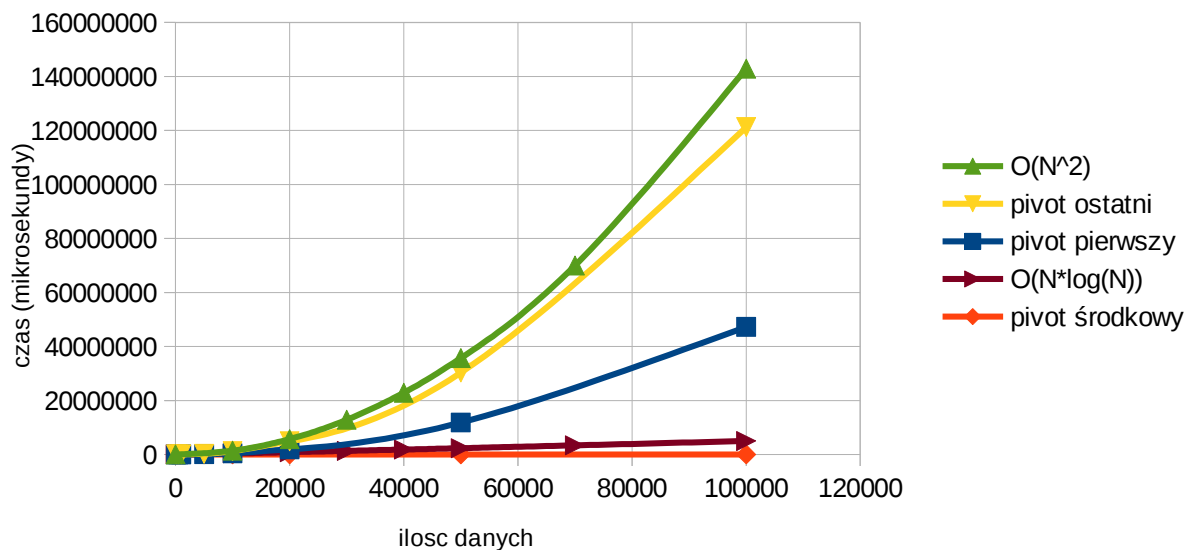
quick_sort(Array & array, int first, int last, char what_pivot)

Pierwszy argument to referencja do tablicy, argumenty first i last to indeksy pierwszego oraz ostatniego elementu tablicy, ostatni argument to znak, który decyduje jak zostanie dobrany pivot - 'f' pierwszy, 'l' ostatni, 'm' środkowy.

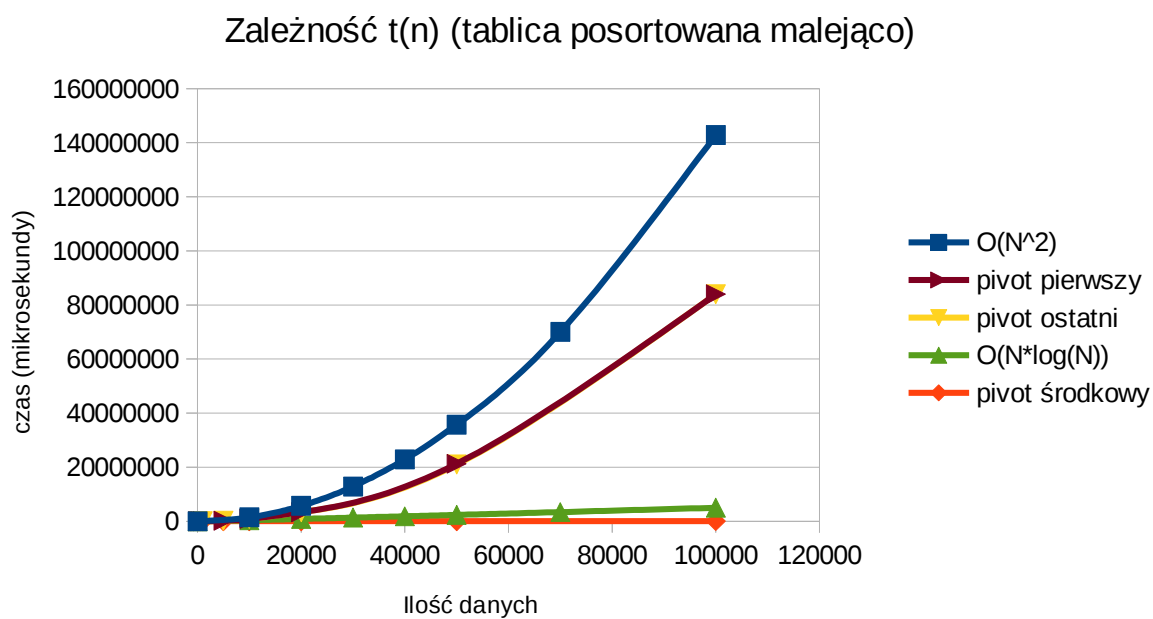
Pomiary zostały zrealizowane dla 3 rodzajów danych, tablicy posortowanej rosnąco, tablicy posortowanej malejąco oraz tablicy z losowo wybranymi elementami. Dla każdego rodzaju danych zostały wykonane 3 różne pomiary dla 3 różnych pivotów.

3. Wyniki

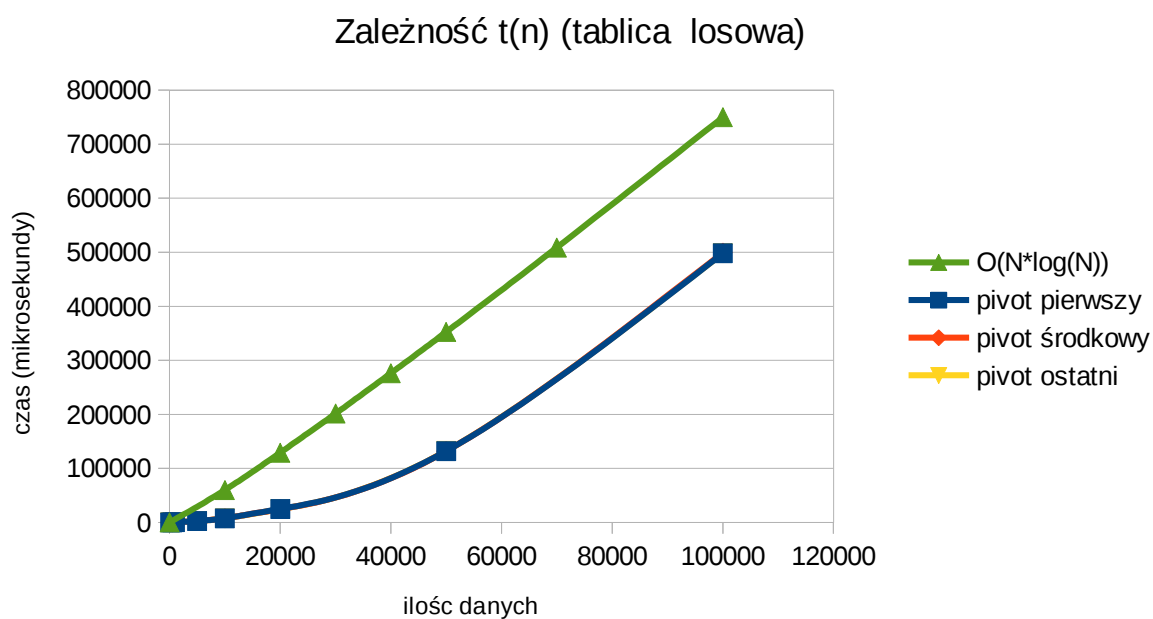
Zależność $t(n)$ (tablica posortowana rosnąco)



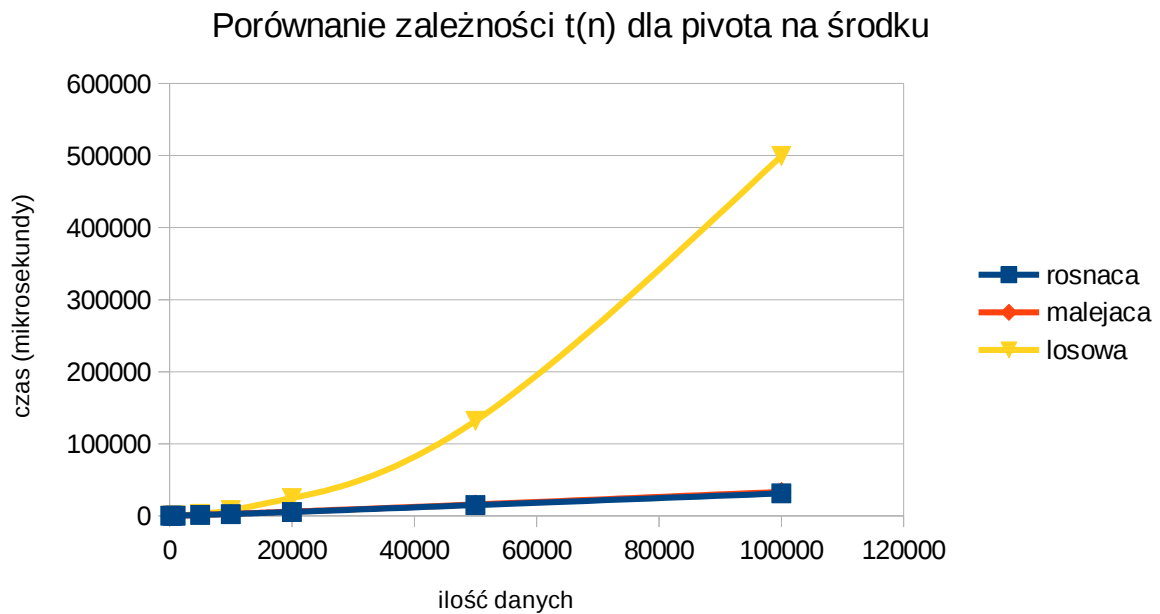
Wykres 1.



Wykres 2.



Wykres 3.



Wykres 4.

4. Wnioski

Na podstawie powyższych wykresów można zauważyć, że jedynym przypadkiem gdzie dobór pivotu nie ma znaczenia to przypadek tablicy z losowo dobranymi elementami (wykres 3). W tym przypadku jego złożoność obliczeniowa jest zbliżona do $O(N \cdot \log(N))$ co jest pozytywnym wynikiem dla algorytmu quick sort.

W przypadkach tablic posortowanych rosnąco i malejąco (wykres 1 i 2), dobór pivotu ma duże znaczenie. W przypadku doboru pivotu na początku lub na końcu, uzyskuje się najgorszą złożoność obliczeniową dla tego algorytmu czyli $O(N^2)$, dzieje się tak dla obu tablic. W przypadku tablicy posortowanej rosnąco, sortowanie wykona się szybciej dla pivotu wybranego jako pierwszy element niż jako ostatni. (wykres 1). W obu przypadkach dobór pivotu jako środkowy element da najkrótszy czas sortowania ze złożonością obliczeniową $O(N \cdot \log(N))$ (wykres 1 i 2).

Algorytm wykonuje się najszybciej dla tablic posortowanych (wykres 4), na podstawie czego można wnioskować, że dla struktur danych częściowo posortowanych warto używać tego algorytmu z wyborem pivotu na środkowych elementach (np. mediany kilku losowo wybranych elementów).