

# Laboratorium 6 - Sprawozdanie

Wojciech Makuch

21 kwietnia 2015

## 1 Zadanie

Poprawa programu framework benchmarkujący dla algorytmu sortowania szybkiego na strukturze kolejka utworzonym na liście bazując na kodzie użytkownika Sheaim/209226. Zadanie polegało na optymalizacji algorytmu i porównania złożoności obliczeniowej.

## 2 Realizacja

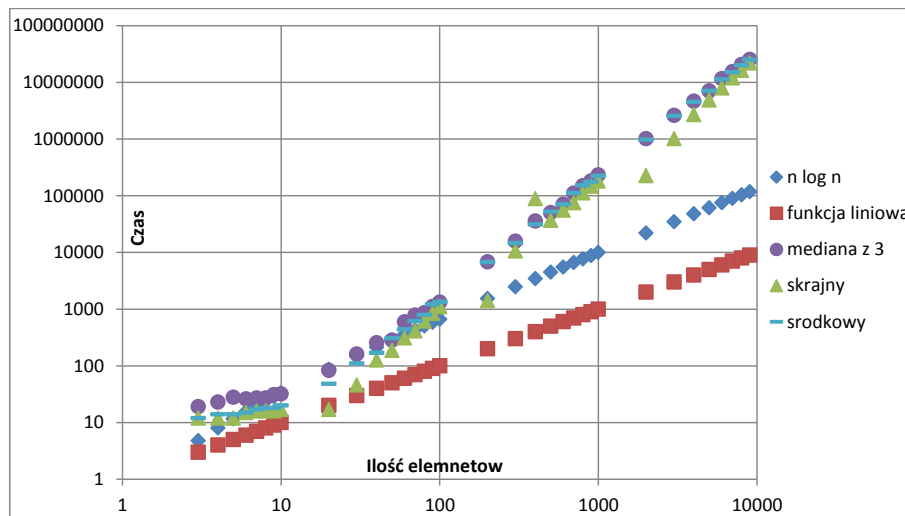
Na samym początku zwrócono uwagę na niepoprawne działanie listy. Usunięto ten problem modyfikując metody *delete\_cell(int)*, *add(cell\*,int)*, *Zamien(int,int)*. Lista działa poprawnie z indeksowaniem od 1 elementu (dla sortowania nie stanowi to problemu, zerowy element wypełniono zerem :-). Program wzbogacono o metody sortowania szybkiego, które za pivot wybierają: pierwszy element, medianę z losowego ciągu trzech liczb oraz środkowy element. Ponadto podzielono program na moduły z rozszerzeniami *.hh* oraz *.cpp*. Do programu nie utworzono dokumentacji, ponieważ kod źródłowy należy do innego użytkownika.

## 3 Działanie

Program nie udostępnia menu użytkownika. Główna funkcja programu wywołuje jedynie funkcję benchmarkującą, która zawiera pętlę zliczającą czas trwania sortowania szybkiego, wyświetlając dane na strumień wyjściowy i zapisując do pliku o nazwie *pomiar\_czasu\_6.txt*. Program zlicza czas dla trzech metod pokoleji: pivot - skrajny element, pivot - mediana oraz pivot - środkowy element.

## 4 Wyniki

Na rys 1. pokazano wykresy złożoności obliczeniowej dla wymienionych metod sortowania szybkiego. Ponadto zamieszczono dla porównania wykres funkcji liniowej oraz funkcji *nlogn*.



Rysunek 1: Wykres złożoności obliczeniowej dla dużej ilości elementów.

## 5 Wnioski

1. Wszystkie metody sortowania dla dużej ilości elementów degenerują się do  $O(n^2)$ .
2. Do 100 elementów przebiegi sortowań pokrywają się z przebiegiem  $n \log n$ .
3. Dla piwota wyliczonego jako mediana z trzech liczb losowego ciągu oraz dla piwota środkowego wykresy pokrywają się już dla ilości większej od 50.
4. Wykres dla złożoności - piwot - skrajny element styka się z przebiegiem liniowym dla bardzo małych wartości. Ponadto dla dużych ilości elementów degeneruje się najwolniej.

## 6 Podsumowane

Średnia złożoność obliczeniowa algorytmu sortowania szybkiego wynosi  $O(n \log n)$ . W przypadku pesymistycznym oraz dla dużych ilości elementów do posortowania złożoność degeneruje się do  $O(n^2)$ .

## 7 Komentarz

Do utworzenia dokumentacji wykorzystano system Doxygen. Funkcja pomiaru czasu dla systemu Windows pobrana ze strony dr. J. Mierzwy. Program skompilowano w środowisku Code::Blocks. Do stworzenia wykresu posłużono się pakietem MS Excel, sprawozdanie napisano używając systemu  $\text{\LaTeX}$ .