

# Sprawozdanie Quicksort

Mateusz Król 226400

April 26, 2017

## 1 Wstęp oraz krótki opis programu

Celem obecnych zajęć było zrealizowanie sortowania szybkiego korzystając, z kodu napisanego wcześniej przez inną osobę. Zajęcia miały za zadanie pokazać nam, jak ważna jest higiena kodu. W programie nie wprowadzono wielkich zmian, nic nie usunięto, a jedynie dodano możliwość sortowania tablicy przez scalanie.

Algorytm merge sort, podobnie jak realizowany wcześniej quick sort opiera swoje działanie na zasadzie "Dziel i zwyciężaj". Program otrzymuje na wejściu tablicę do posortowania (w I przypadku jest to tablica z losowymi liczbami, a w II przypadku jest to tablica wcześniej już posortowana). Program dzieli otrzymaną tablicę  $n$ -elementową na połowy rekurencyjnie, aż do otrzymania  $n$  zbiorów jednoelementowych. Następnie porównując liczby ze sobą "scala" tablicę na powrót.

## 2 Uśrednione czasy dla 10 pomiarów, 2 przypadki wybory środka podziału

ilosc elementow/początek podziału	początek [ms]	środek[ms]
100	0,0097	0,0063
1000	0,1724	0,087
10000	2,2829	1,0929
100000	28,1063	13,945

Table 1: Tablica nieposortowana

ilosc elementow/początek podziału	początek [ms]	środek[ms]
100	0,00367	0,0224
1000	2,938	2,024
10000	336,71	207,0306
100000	33231,94	20584,26

Table 2: Tablica posortowana rosnąco

## 3 Wnioski

Jak możemy zauważyć z tabeli powyżej i z wykresów poniżej (skala log-log) w obydwu przypadkach algorytm minimalnie szybciej sortuje tablice, które wcześniej były już posortowane. Program w przypadku gdy wybierzemy podział na początku tablicy prezentuje złożoność obliczeniową  $O(n^2)$ , jednak idea tego algorytmu polega na tym, aby za środek przedziału obierać środek tablicy, dlatego, że ilość gałęzi w drzewie robi się niepotrzebnie zbyt duża. Gdy za początek podziału wybierzemy środek tablicy, to algorytm prezentuje złożoność obliczeniową na poziomie  $O(n \cdot \log(n))$ . Algorytm sortowania przez scalanie w przeciwieństwie do quicksortu potrzebuje więcej pamięci, aby kopiować tablicę.

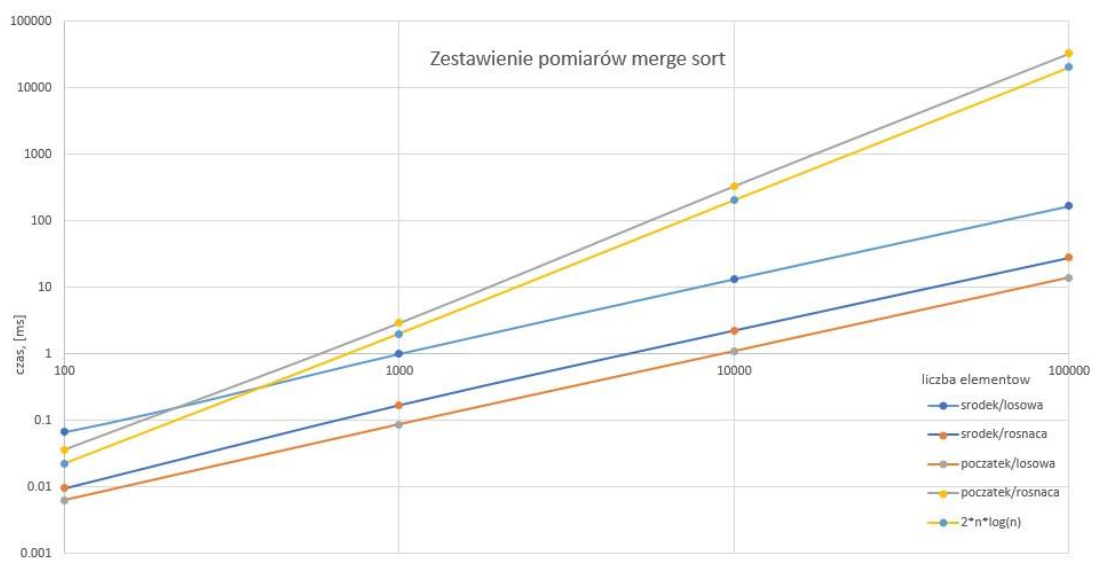


Figure 1: Działanie algorytmu merge sort