

Laboratorium 4 - Sprawozdanie

Wojciech Makuch

8 kwietnia 2015

1 Zadanie

Program framework benchmarkujący dla zaimplementowanego algorytmu sortowania szybkiego opartego na strukturze typu lista.

2 Realizacja

Gotową implementację listy skopiowano z repozytorium Sheaim/209226. Dodano metodę sortowania szybkiego *quicksort(int left, int right)*; oraz uzupełniono kod o niezbędne metody, jak np *idz(int i)*; pozwalająca odnieść się do konkretnego elementu listy. Dla algorytmu sortowania za pivot wybrano element środkowy sortowanej tablicy.

3 Działanie

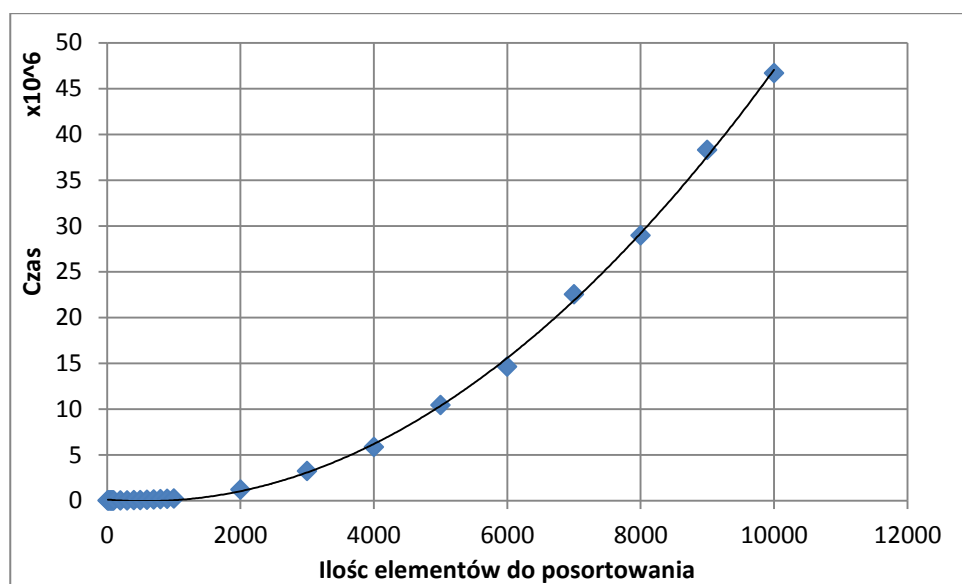
Główna funkcja programu zawiera jedną pętlę testującą algorytm. Użyta funkcja *licz(dllist *obiekt, int N)*; wypełnia tablicę liczbami pseudolosowymi z zakresu 0-9 następnie mierzy i zwraca czas ich sortowania. Uzyskane wyniki program zapisuje do pliku o nazwie *pomiar_czasu_4.txt*

4 Wyniki

Z uzyskanych wyników wyrysowano przebieg funkcji pokazanej na Rysunku 1. Wynika z niego, że złożoność obliczeniowa takiego algorytmu wynosi $O(n^2)$. Z teoretycznego punktu widzenia taka złożoność jest poprawna przy najgorszych założeniach (np. za duża lub za mała wartość pivotu). Średnia złożoność obliczeniowa tego algorytmu wynosi $O(n \log n)$.

5 Komentarz

Funkcja pomiaru czasu dla systemu Windows pobrana ze strony dr. J. Mierwy. Program skompilowano w środowisku Code::Blocks. Do stworzenia wy-



Rysunek 1: Wykres złożoności obliczeniowej

kresu posłużono się pakietem MS Excel, sprawozdanie napisano używając systemu $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$.