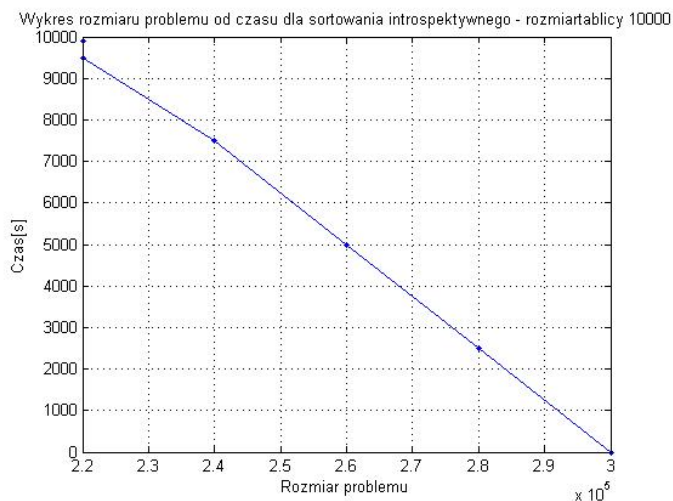


Sprawozdanie z laboratorium nr5 -PAMSI

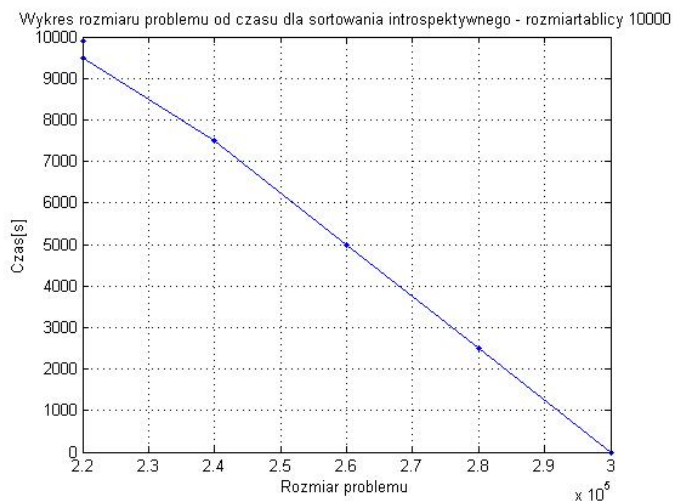
Karolina Morawska

30 03 2014

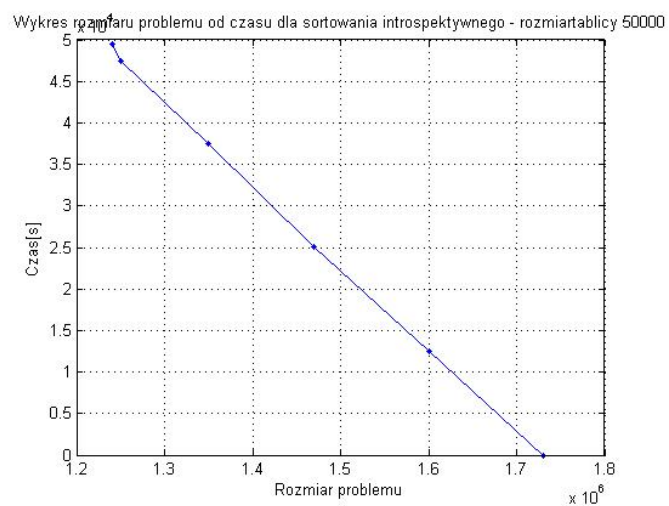
Zadanie do wykonania
Ocena złożoności obliczeniowej poszczególnych algorytmów sortowania .



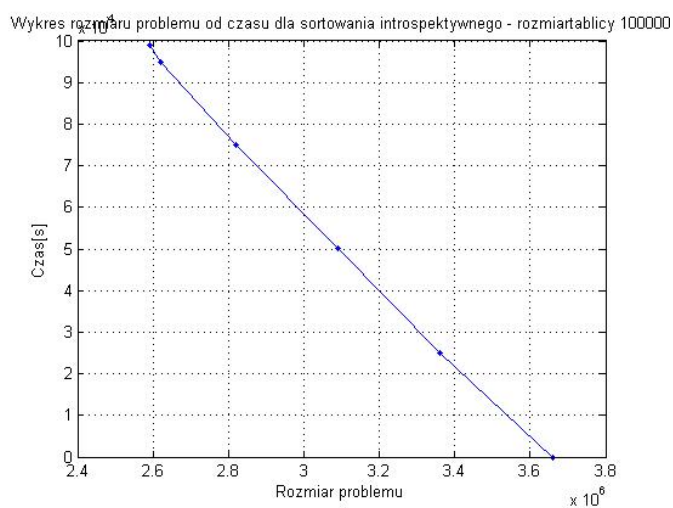
Rysunek 1: Wykres zależności rozmiaru problemu od czasu działania algorytmu dla sortowania introspektywnego.



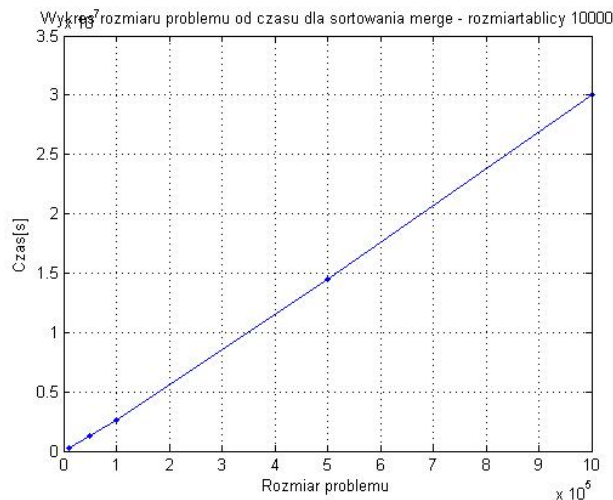
Rysunek 2: Wykres zależności rozmiaru problemu od czasu działania algorytmu dla sortowania introspektywnego.



Rysunek 3: Wykres zależności rozmiaru problemu od czasu działania algorytmu dla sortowania introspektywnego.



Rysunek 4: Wykres zależności rozmiaru problemu od czasu działania algorytmu dla sortowania introspektywnego.



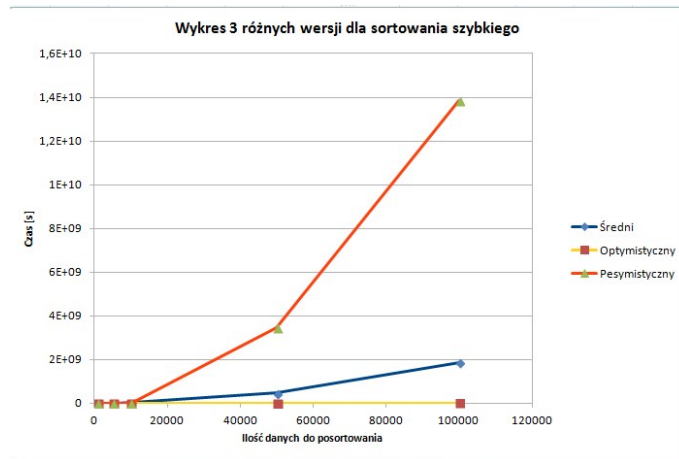
Rysunek 5: Wykres zależności rozmiaru problemu od czasu działania algorytmu dla sortowania merge czyli przez scalanie

Wybrano sortowanie szybkie którego złożoność obliczeniowa wynosi $O(n \log n)$. Jest to sortowanie wydajne oraz często używane ze względu na prostotę implementacji i szybkość. Następnym sortowaniem jest mergesort-rekurencyjny algorytm sortowania danych który działa w taki oto sposób:

- dzieli zestaw danych na dwie równe części
- stosuje sortowanie przez scalanie dla każdej z nich oddzielnie, chyba że pozostał już tylko jeden element;
- łączy posortowane podciągi w jeden.

Jego złożoność obliczeniowa jest taka sama jak w przypadku quicksort.

Ostatnie sortowanie to introspektywne. W przypadku ogólnym, a więc również w najgorszym, algorytm Sortowania Introspektywnego posiada złożoność obliczeniową taką samą jak 2 poprzednie. W najgorszym przypadku algorytm wykonuje najpierw rekurencyjne wywołanie, takie jak w Sortowaniu Szybkim, a następnie dla pozostałego podzbioru wywołuje procedurę Heap Sort. Jest on algorytmem sortującym w miejscu.



Rysunek 6: Wykres zależności rozmiaru problemu od czasu działania algorytmu dla sortowania szybkiego - 3 różne przypadki

Wnioski:

Czas sortowania dla zbiorów nieuporządkowanych jest dużo większy od czasu sortowania dla zbiorów uporządkowanych.

Optymistyczny przypadek sortowania zachodzi tylko i wyłącznie wtedy gdy kluczem podziału jest mediana z sortowanego zbioru. Podział daje nam równe zbiory.

W przypadku przeciętnym, gdzie rozkład prawdopodobieństwa wyboru elementu jest taki sam złożoność jest zaledwie wyższa o 39

Przypadek pesymistyczny zachodzi wtedy, gdy każdy podział jest skrajnie nierównomierny. tj. generuje jedno podzadanie puste, a drugie o 1 mniejsze niż całe zadanie.

Wszystkie otrzymane czasy sortowania są proporcjonalne do iloczynu $n \log^2 n$, możemy zauważyć zatem, iż klasa złożoności obliczeniowej algorytmu sortowania szybkiego jest równa $O(n \log n)$.