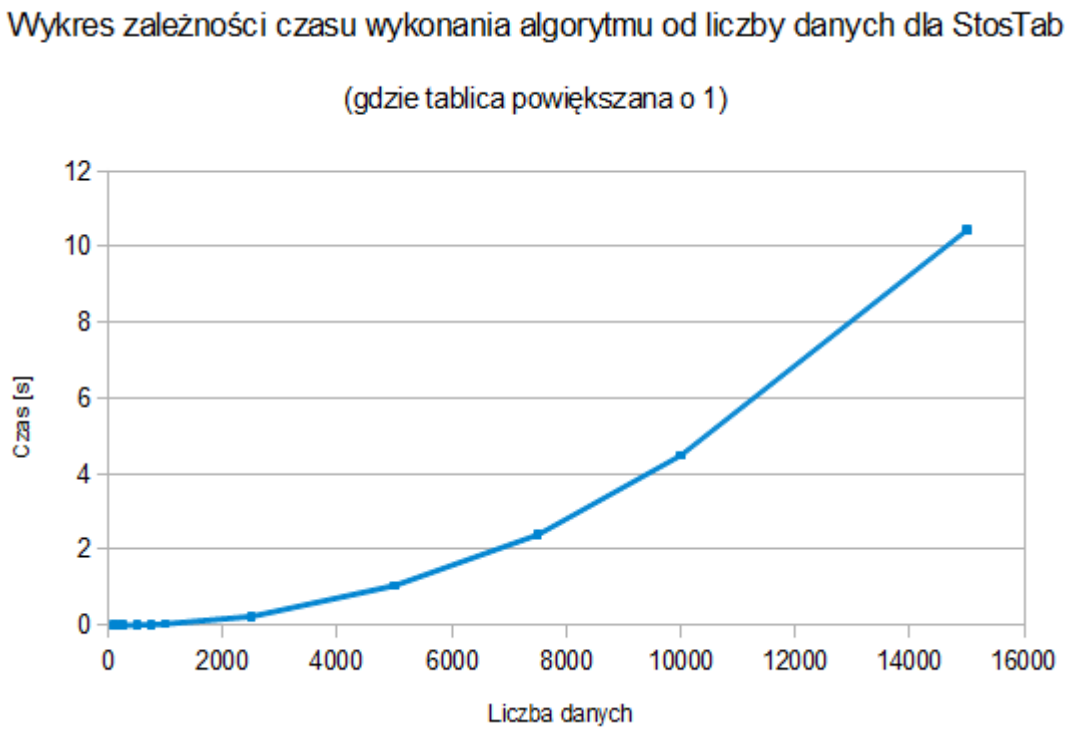
Czas realizacji algorytmów obsługujących stos lub kolejkę.

Monika Litwin 200586

21 marca 2014

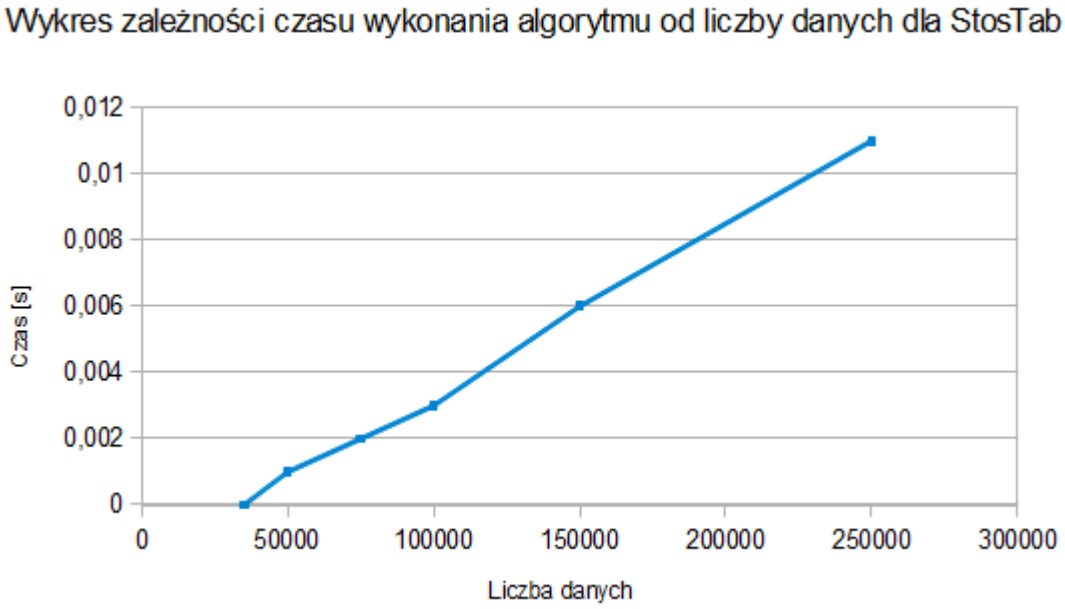
StosTab

Stos zaimplementowany przy użyciu tablicy. Rozmiar stosu zostaje zwiększany o jeden w trakcie dodawaniu elementów na stos. Jak widać na wykresie, czas realizacji algorytmu dla tej klasy rośnie wykładniczo wraz ze wzrostem liczby elementów.



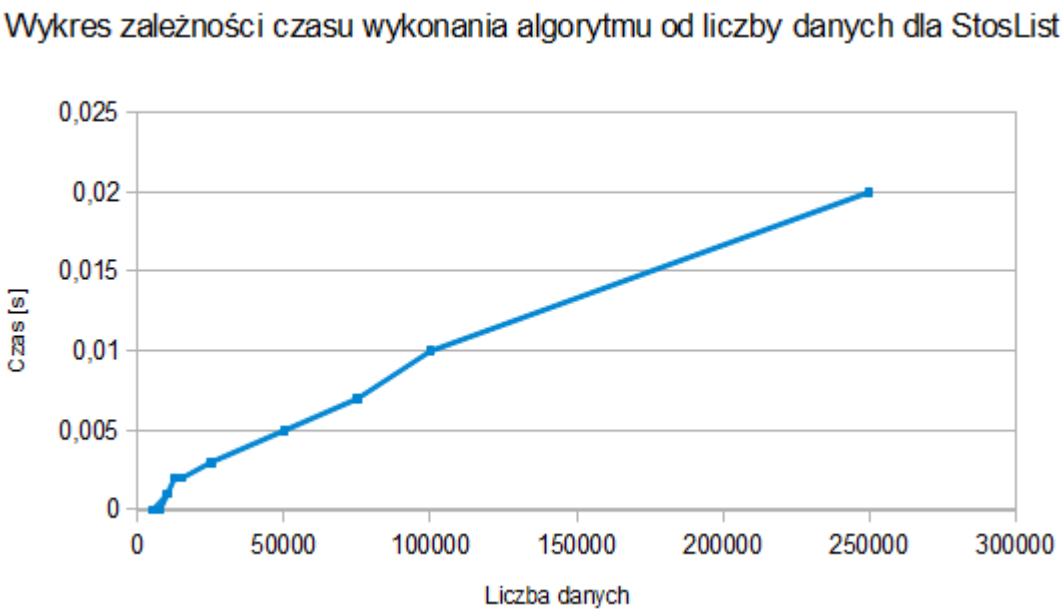
StosTab

Stos zaimplementowany przy użyciu tablicy. Rozmiar stosu zostaje zwiększany dwukrotnie, gdy już kończy się miejsce na nim, w trakcie dodawaniu elementów na stos. Przyrost czasu realiza-
cji algorytmu wraz ze wzrostem liczby elementów dla tej klasy jest zbliżony do liniowego.



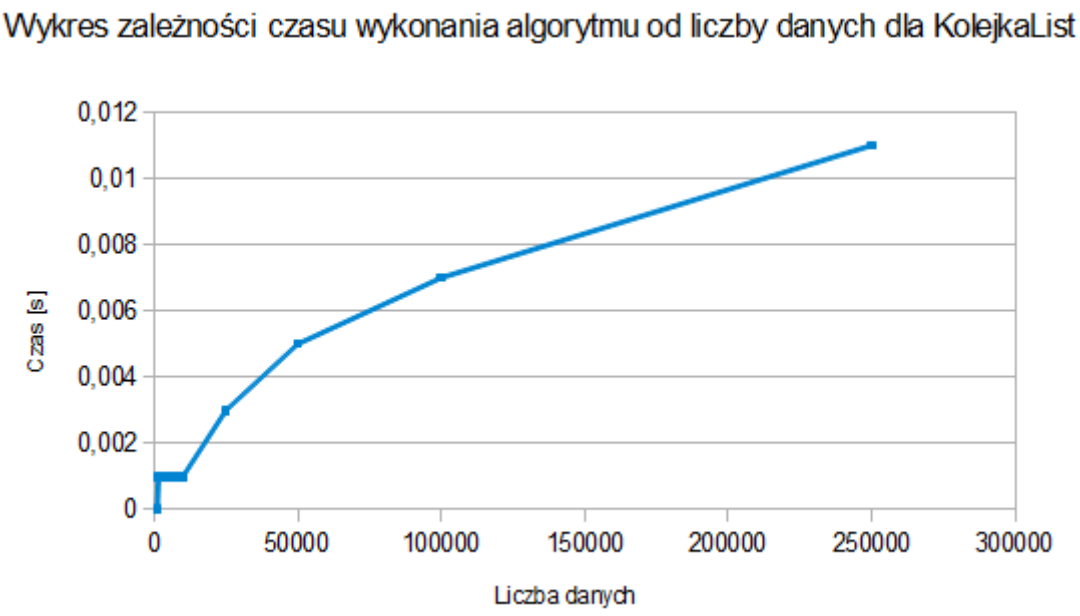
StosList

Stos zaimplementowany za pomocą listy. Listę pomaga zrealizować wewnętrzna struktura - Element. Czas wykonania algorytmu w zależności od liczby elementów, wzrasta w sposób zbliżony do liniowego.



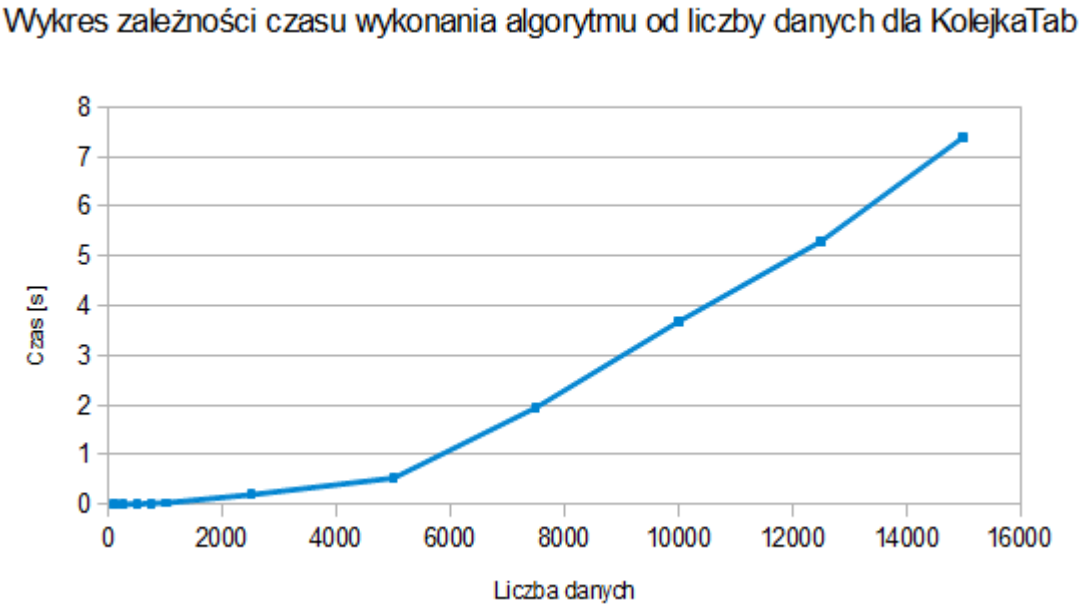
KolejkaList

Kolejka zaimplementowana za pomocą listy. Listę pomaga zrealizować wewnętrzna struktura - Element. Czas wykonania algorytmu w zależności od liczby elementów, wzrasta logarytmicznie.



KolejkaTab

Kolejka zaimplementowana przy użyciu tablicy. Rozmiar kolejki zostaje zwiększany o jeden w trakcie dodawania do niej elementów. Jak widać na wykresie, czas realizacji algorytmu dla tej klasy rośnie wykładniczo wraz ze wzrostem liczby elementów.



WNIOSKI:

-Najdłuższym czasem wykonania algorytmu, charakteryzuje się ten realizowany dla klasy StosTab ze zwiększaniem rozmiaru o jeden. Zaraz po nim jest ten wykorzytywany w klasie KolejkaTab, w którym również używane jest zwiększanie o jeden. Ten sposób z pewnością pozwala oszczędzić pamięć, ale też znacznie zwiększa złożoność obliczeniową.

-Dużo wydajniejszy czasowo algorytm wykorzystuje klasa StosTab, która podwaja rozmiar, za każdym razem gdy miejsce się kończy. Dla przykładu 0,1s przy 50 000 danych na stosie (w poprzednim sposobie było ponad 10s przy 15 000 danych). Powoduje jednak dużo większe zużycie pamięci.

-Jeśli chodzi o wykorzystanie implementacji listowej, wydajniejszym czasowo algorytmem jest również ten, zajmujący się kolejką. Są one jednak zbliżone wartościowo, czasy wychodzą tych samych rzędów.