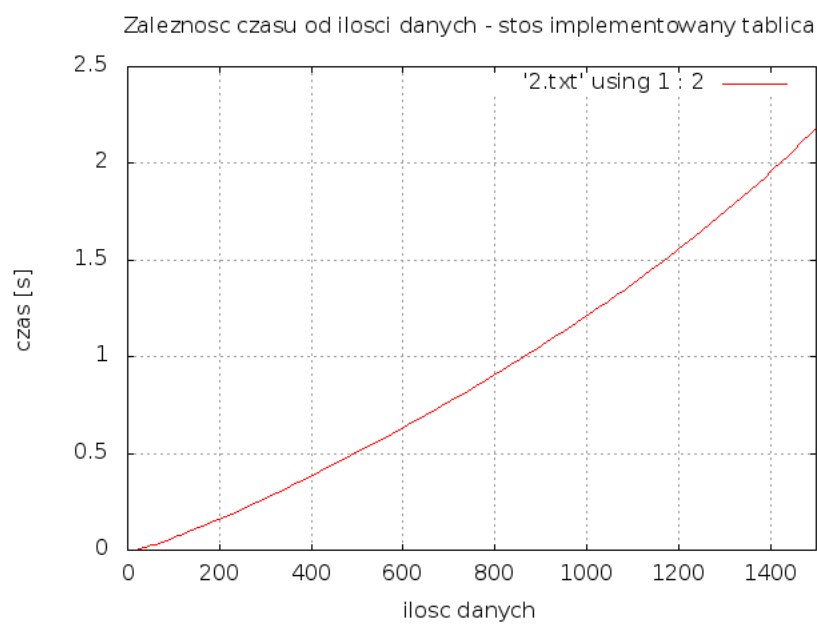


Różnice w czasie realizacji algorytmu wypełniania stosu oraz listy w zależności od implementacji

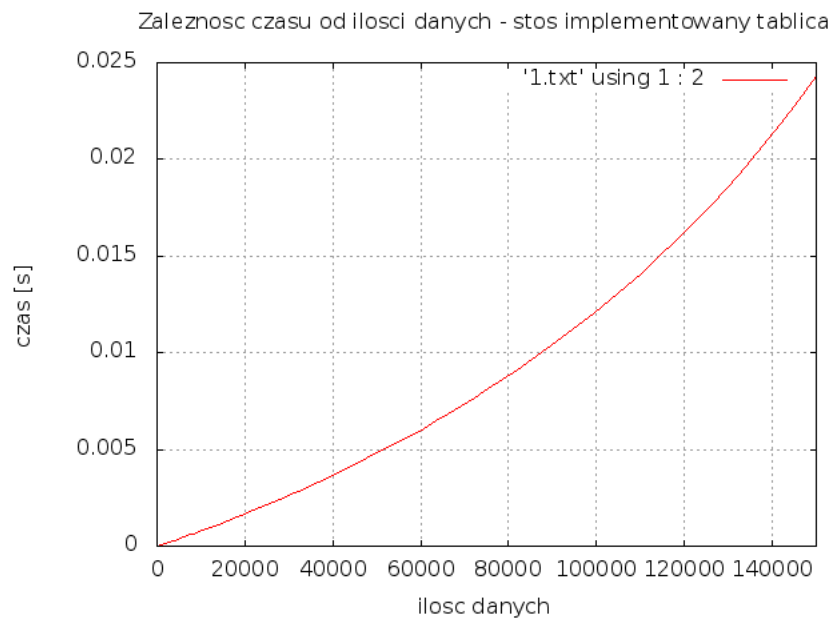
Arkadiusz Cyktor 200367

19 marca 2014

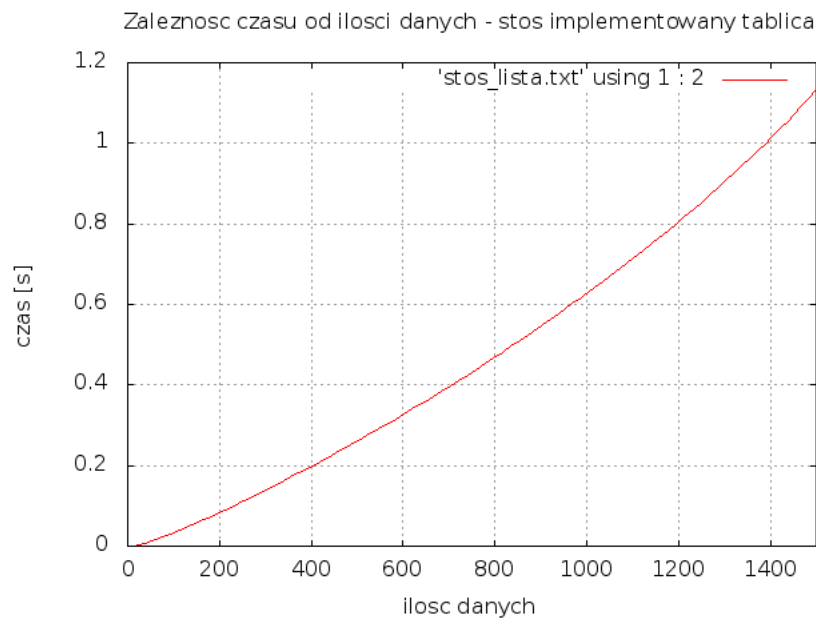
1. Poniższy wykres przedstawia zależność czasu potrzebnego na wykonanie algorytmu od ilości danych, dla stosu zaimplementowanego przy użyciu tablicy. W tym przypadku tablica zwiększana jest o jedno pole, za każdym razem gdy dodawana jest nowa zmienna. Na wykresie widac, że złożoność obliczeniowa tego algorytmu rośnie wykładniczo.



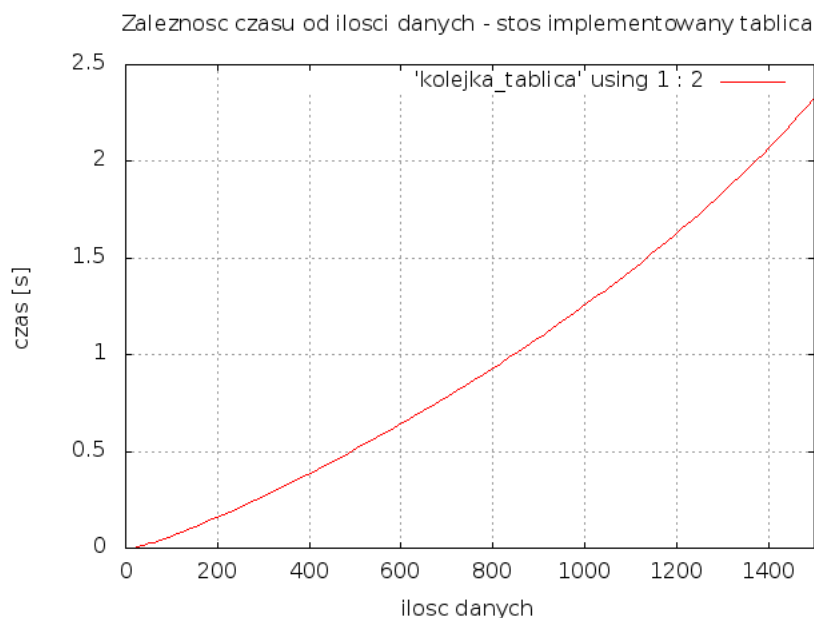
2. Ponizszy wykres przedstawia te sama zaleznosc dla tej samej implementacji stosu, jednak tym razem rozmiar tablicy zostaje zwiekszony dwukrotnie w momencie osiagniecia przez nia zapelnienia. Z wykresu mozna wywnioskowac, ze zlozonosc obliczeniowa takiego algorytmu rowniez zwieksza sie wykladniczo.



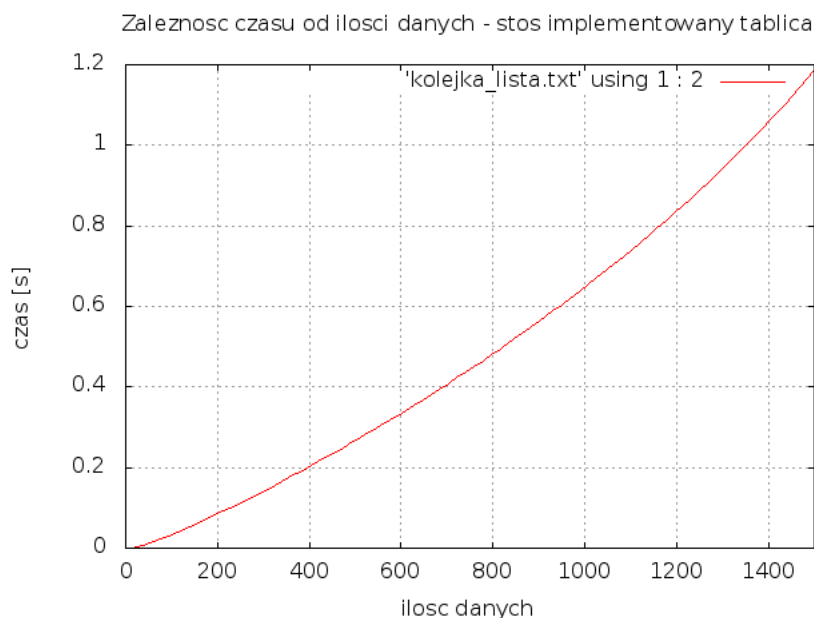
3. Ponizszy wykres przedstawia zaleznosc czasu potrzebnego na wykonanie algorytmu od ilosci danych, dla stosu zaimplementowanego przy uzyciu listy. Ponownie widzimy, ze zlozonosc obliczeniowa rosnie wykladniczo.



4. Ponizszy wykres przedstawia zaleznosc czasu potrzebnego na wykonanie algorytmu od ilosci danych, dla kolejki zaimplementowanej przy uzyciu tablicy. Zlozonosc rosnie wykladniczo.



5. Ponizszy wykres przedstawia zaleznosc czasu potrzebnego na wykonanie algorytmu od ilosci danych, dla kolejki zaimplementowanej przy uzyciu listy. Zlozonosc rosnie wykladniczo.



Wnioski:

- Porównanie dwóch pierwszych wykresów pokazuje, że dwukrotne zwiększanie tablicy jest lepszym rozwiązaniem, przy pracy na dużej ilości danych, niż rozszerzanie jej o pojedyncze pola. Mimo, że złożoność obu rozwiązań różnie wykładniczo, to jednak przyrost ten jest znacznie mniejszy w przypadku drugiej metody, wymaga jednak ona znacznie więcej pamięci.

- Możemy zauważyć, że implementacja stosu przy pomocy listy osiąga lepsze wyniki do tych, które uzyskaliśmy podczas testowania implementacji tablicowej z powiększaniem o jeden, pomimo faktu, że lista również jest powiększana o jeden element w danej chwili, jeśli zachodzi taka konieczność.

- Implementacja kolejki przy pomocy tablicy wydaje się być najmniej wydajna ze wszystkich testowanych.

- Podsumowując: lista wydaje się być wydajniejsza obliczeniowo niż stosowana w tym samym celu tablica.