

Sprawozdanie z laboratorium nr 3

STRUKTURY DANYCH

16.03.2014

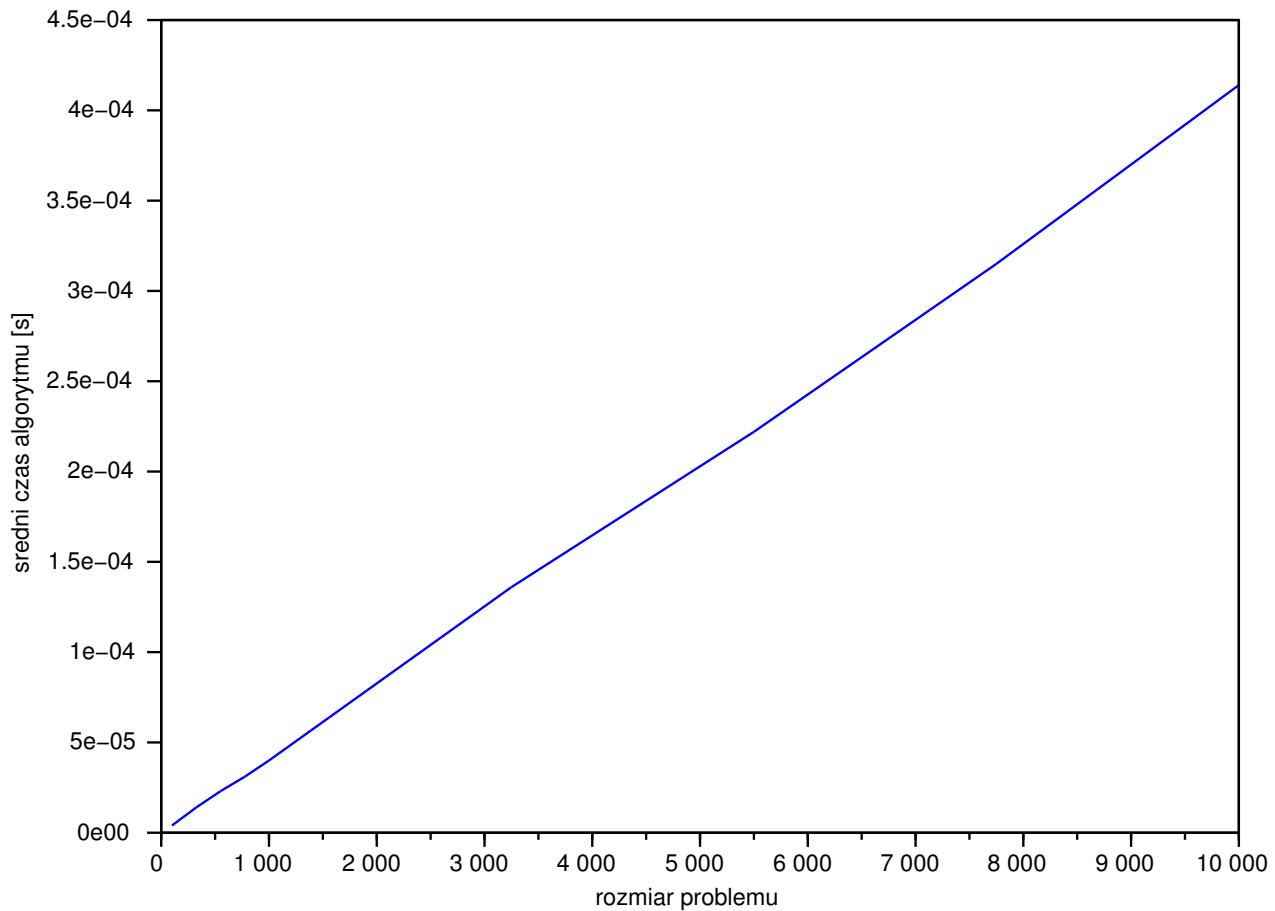
1. Cel programu

Zadaniem do wykonania było przygotowanie programu mierzącego czas wykonywania algorytmu dla różnych struktur danych. Wyniki jego działań można zobaczyć na wykresach w punkcie 2.

2. Wyniki działania programu — wykresy

2.1. Stos operujący na tablicach

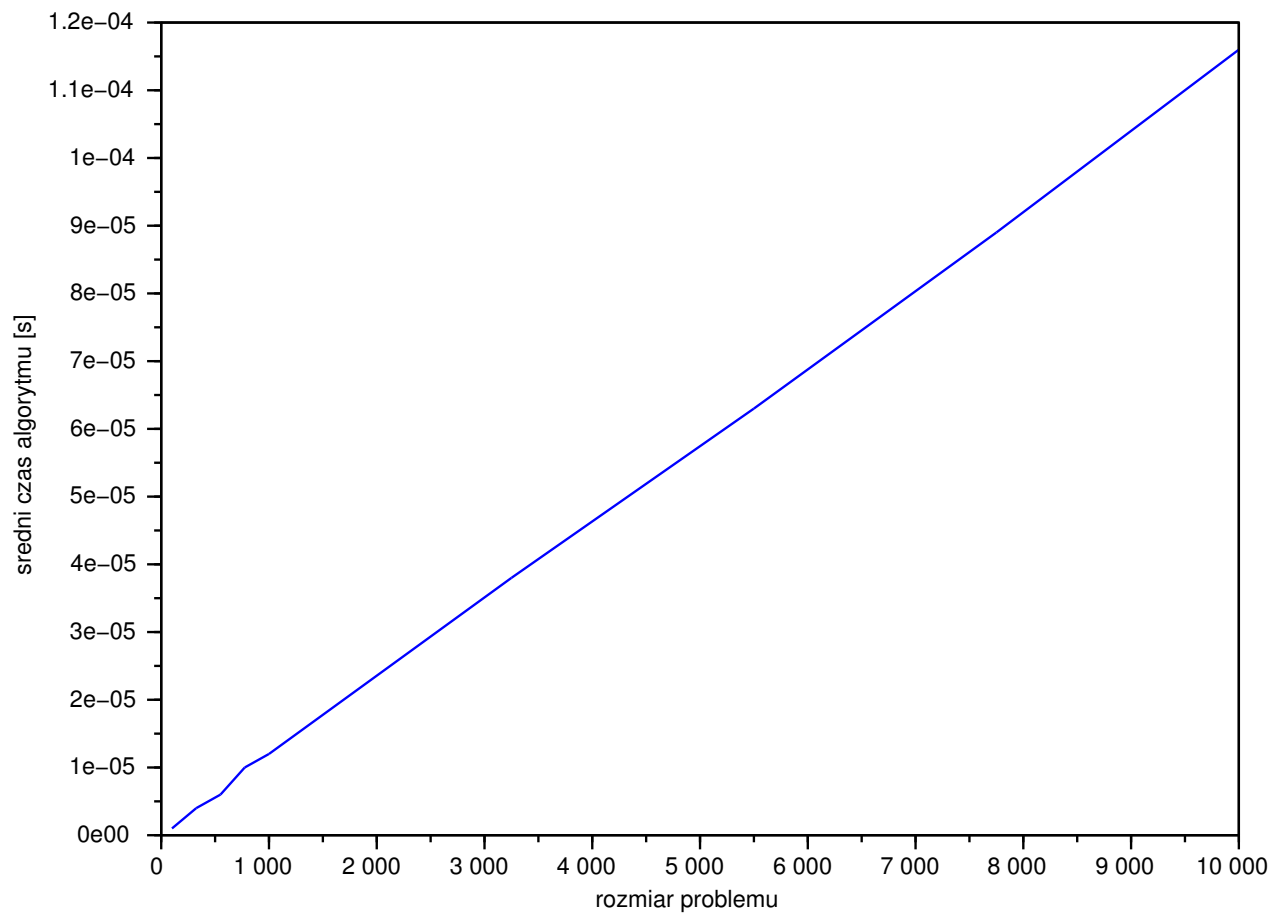
Wykonuje podstawowe operacje na stosie. W przypadku przekroczenia rozmiaru tablicy zwiększa jej rozmiar o 1.



Rysunek 1. Zależność czasu wykonywania algorytmu od rozmiaru problemu dla stosu.

2.2. Stos operujący na tablicach — wydajniejszy

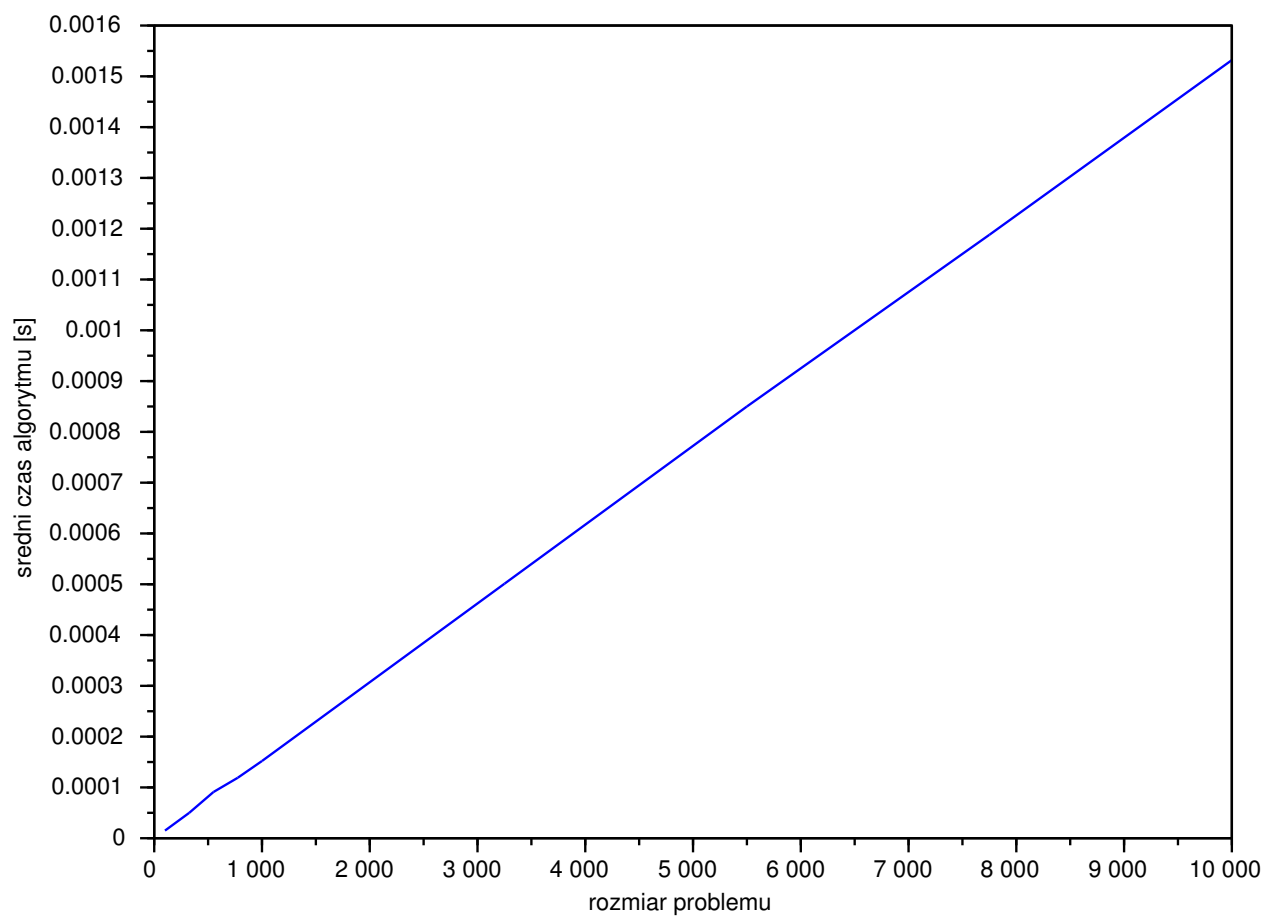
Wykonuje podstawowe operacje na stosie. W przypadku przekroczenia rozmiaru tablicy zwiększa jej rozmiar dwukrotnie.



Rysunek 2. Zależność czasu wykonywania algorytmu od rozmiaru problemu dla stosu — wersja zoptymalizowana.

2.3. Stos operujący na liście

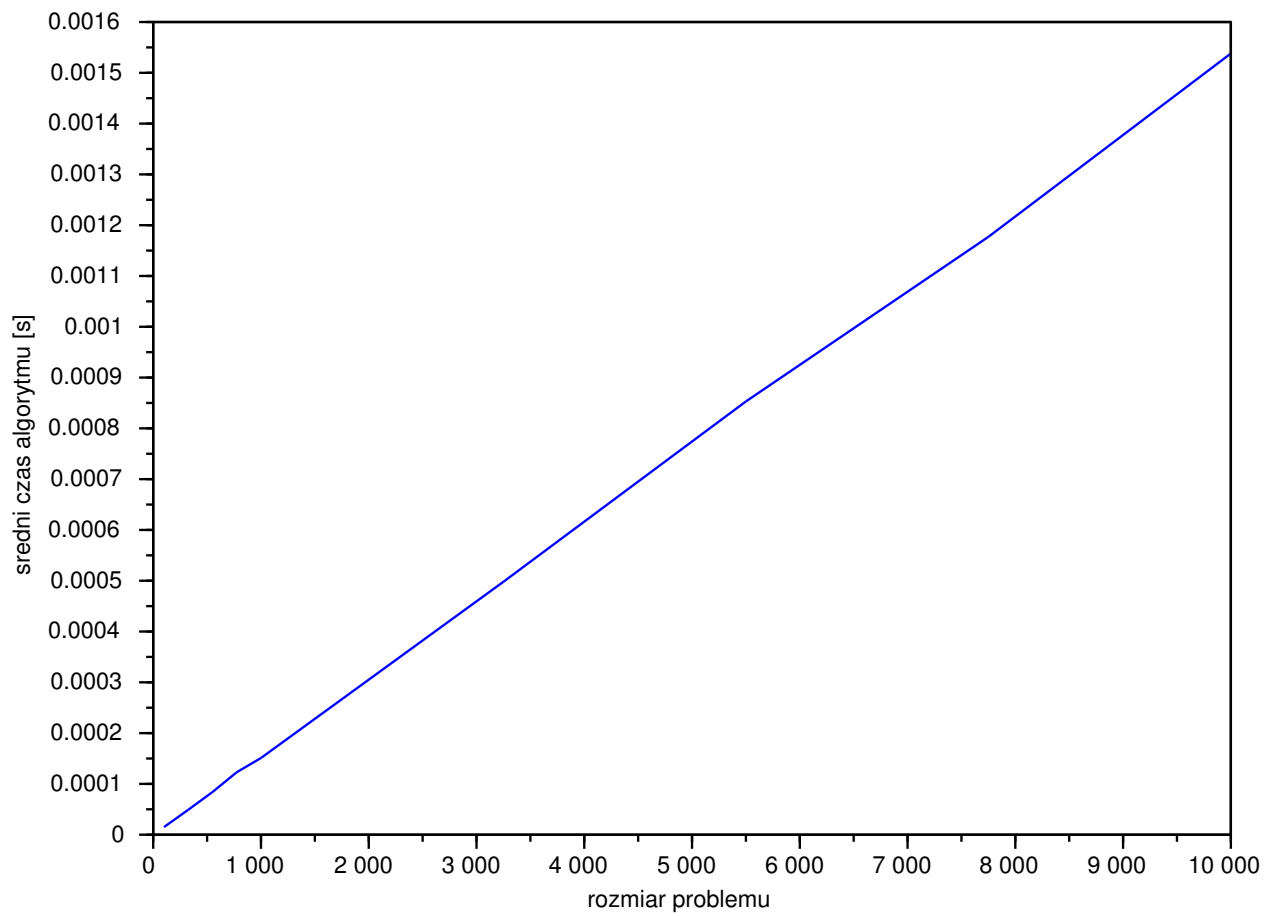
Wykonuje podstawowe operacje stosu na liście.



Rysunek 3. Zależność czasu wykonywania algorytmu od rozmiaru problemu dla stosu na liście.

2.4. Kolejka operująca na liście jednokierunkowej

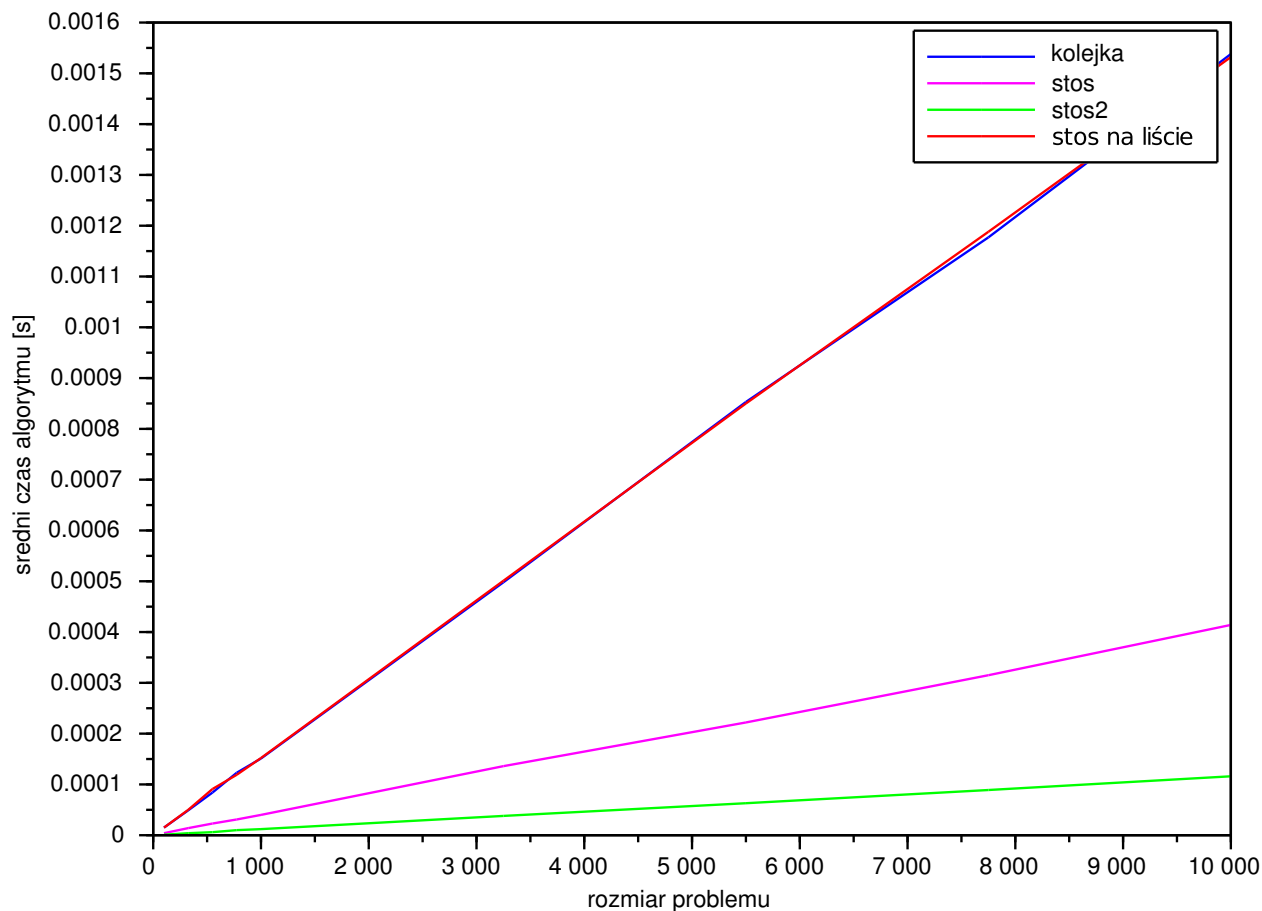
Wykonuje podstawowe operacje na kolejce (liście jednokierunkowej).



Rysunek 4. Zależność czasu wykonywania algorytmu od rozmiaru problemu dla kolejki.

2.5. Wykres zbiorczy

Porównanie czasu wykonywania poszczególnych implementacji.



Rysunek 5. Zależność czasu wykonywania algorytmu od rozmiaru problemu dla wszystkich implementacji.

3. Wnioski

- Wszystkie zależności są w przybliżeniu liniowe. Drobne odkształcenia wynikają z błędów pomiarowych (np.: przez programy pracujące w tle).
- Najszybszy jest algorytm operujący na tablicy, której rozmiar w przypadku braku miejsca zwiększamy dwukrotnie.
- Najwolniej działają algorytmy zaimplementowane na listach. W związku z tym lepszym rozwiązaniem jest operowanie na tablicach.
- Algorytmy kolejki i stosu działają z taką samą szybkością, ponieważ są implementowane w ten sam sposób.

4. Załączniki

- Dokumentacja z Doxygena "Dokumentacja.pdf"