

Algorytmy numeryczne Zadanie 2 Działania na macierzach

Działania na macierzach zostały przeprowadzone przy użyciu języka Java oraz języka C++, z użyciem biblioteki Eigen.

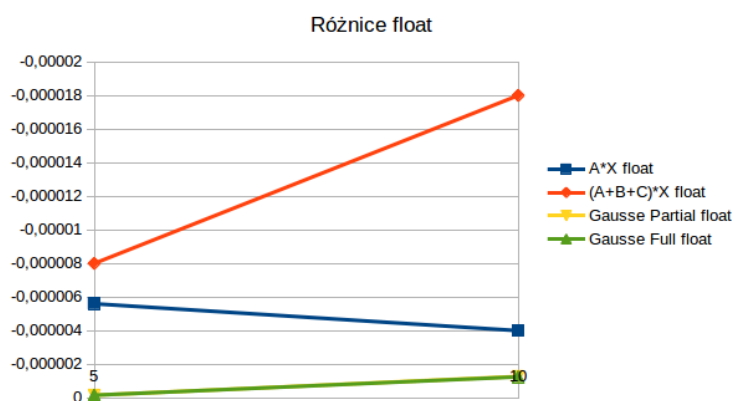
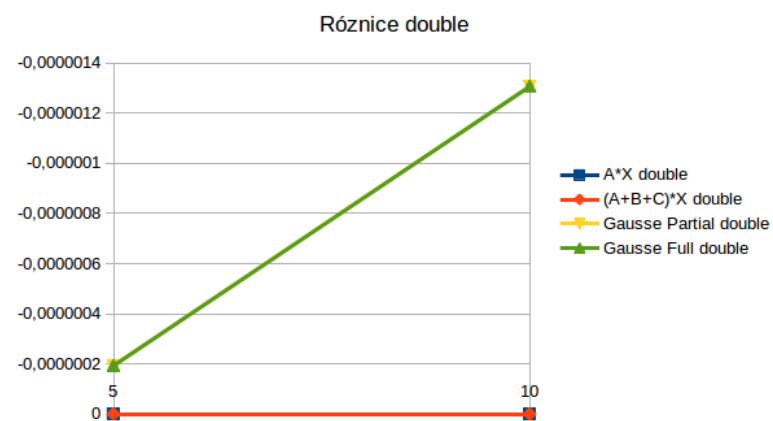
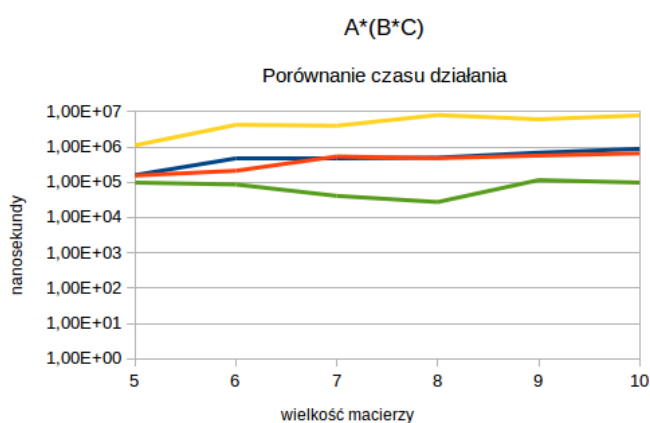
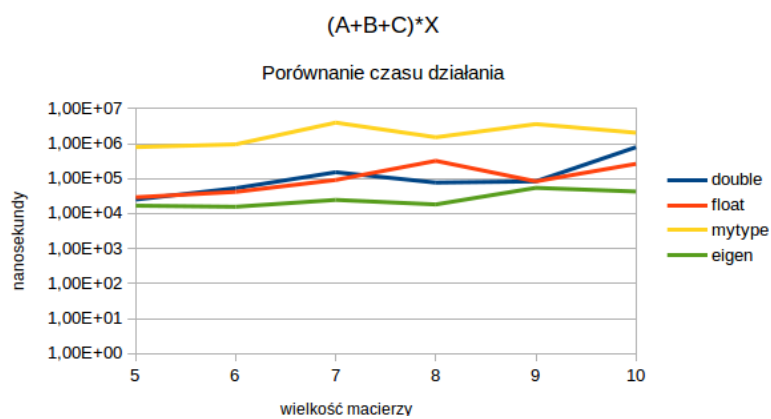
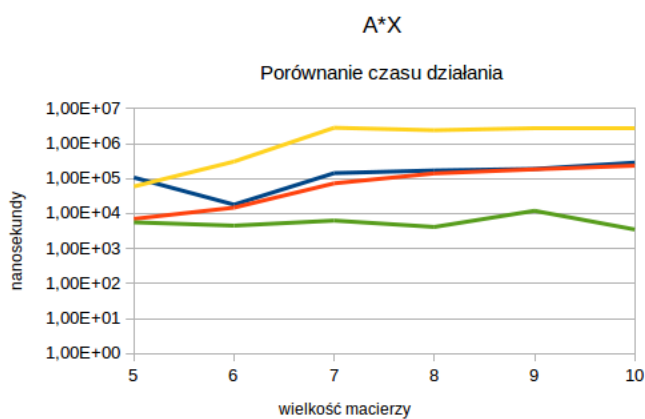
Testy przeprowadzono dla 3 wzorów:

- $A * X$
- $(A + B + C) * X$
- $A * B * C$

Litery A, B, C to macierze kwadratowe, a X to wektor. Użyto 3 typy zmiennych:

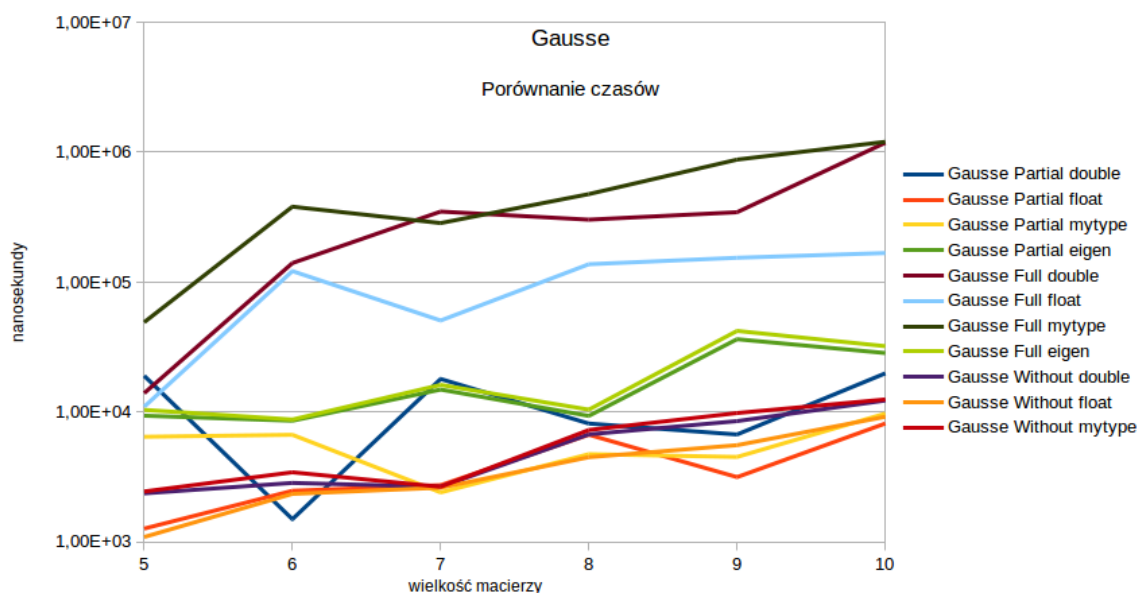
- double
- float
- typ własny (nie tracący dokładności)

Sprawdzam czasy oraz dokładność obliczeń, dla ogólnienia czasów wykonano 10 powtórzeń metody.



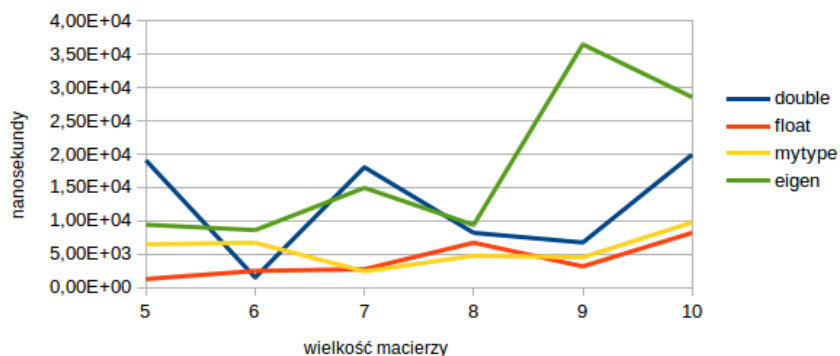
Sprawdzone został również wpływ rodzaju eliminacji Gaussa:

- bez wyboru elementu podstawowego,
- z częściowym wyborem elementu podstawowego,
- z pełnym wyborem elementu podstawowego



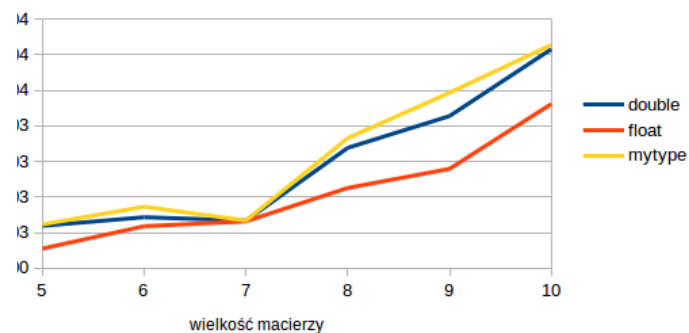
Gausse Partial Pivoting

Porównanie czasu działania



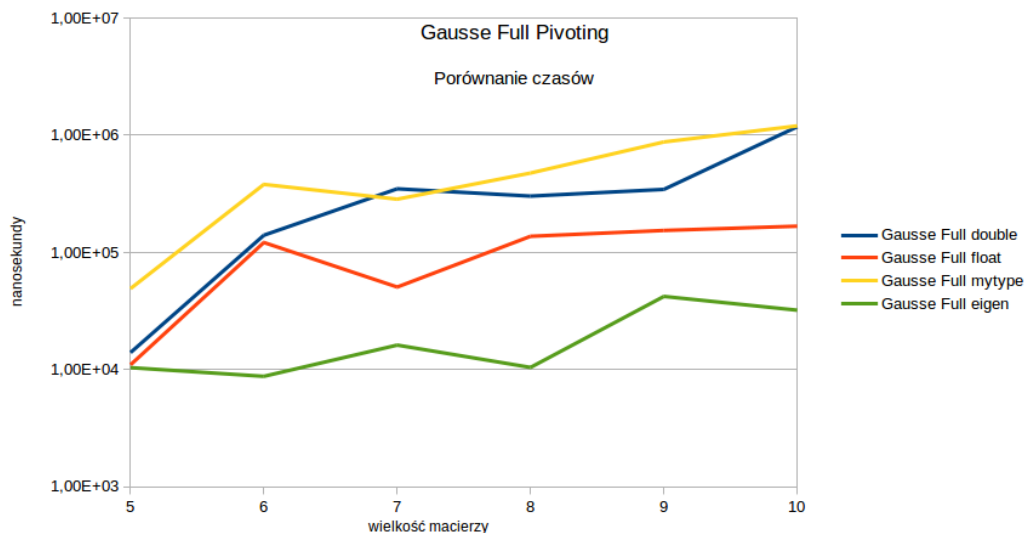
Gausse Without Pivoting

Porównanie czasu działania



Gausse Full Pivoting

Porównanie czasów



Jak można zauważyć,

najbardziej czasochłonnym rozwiązywaniem układu równań jest Full Pivoting Gausse, wpływ na to ma duże zużycie czasu na zmianę kolumn oraz wierszy.

Dzięki stworzeniu swojego typu, nie tracimy w ogóle dokładności, jednak najmniej dokładnym jest typ float. Jest to spowodowane utratą kilku miejsc po przecinku co ostatecznie ma wpływ na wynik.

Jak widzimy coraz większe macierze oczywiście mają wpływ na czas działania, lecz są to niewielkie różnice, rosną stopniowo powoli. Rozmiar macierzy ma większy wpływ na dokładność wyniku.