## Algorytmy Numeryczne

**Zadanie 3**

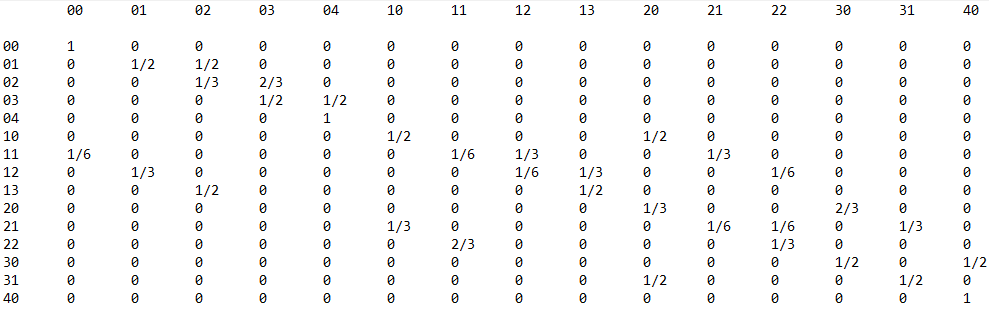
**Paulina Żurawska i Paweł Szczupak**

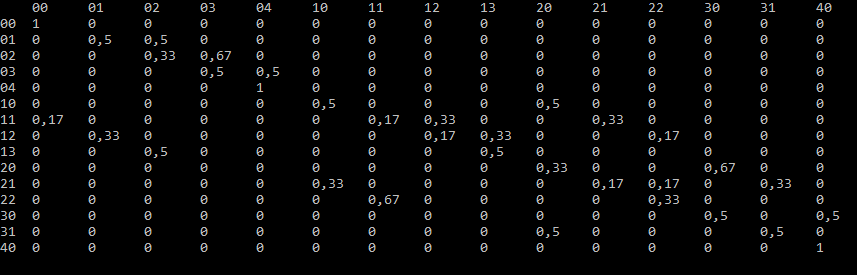
**Grupa 1, Aplikacje internetowe i bazy danych**

Zadanie polega na obliczeniu prawdopodobieństwa, tego że głosowanie zakończy się sukcesem przy z góry ustalonej liczbie agentów i stałych zasadach zmiany statusów wylosowanej pary. Prawdopodobieństwo wylosowania pary jest równomiernie rozłożone.

Wynik jest uzyskiwany dzięki macierzy opisującej układ równań liniowych i metodą Gaussa, Jacobiego i Gaussa-Seidela na danych typu *double*.

Układ równań dla macierzy 4x4 obliczony manualnie i (poniżej) układ równań uzyskany dzięki programowi:





Na wykresach 1-6 przedstawiono błędy przy konkretnych precyzjach pomiędzy algorytmem Monte Carlo a odpowiednio Jacobiego i Gaussa-Seidela. Algorytmy te uzyskują takie same, bądź tak bliskie sobie wyniki, że różnica pomiędzy nimi i ich precyzjami jest minimalna, dlatego też wykresy są identyczne.

Zostały także przeprowadzone testy polegające na zmierzeniu czasu działania obu algorytmów. Rezultaty zostały ukazane na wykresach 7-9. Wynika z nich, że algorytm Gaussa-Seidela jest szybszy, tak więc ten algorytm okazał się najbardziej optymalnym.

Figure Błąd pomiędzy algorytmami Monte Carlo a Jacobim dla precyzji 10^(-6) i wzrastającej liczby agentów

Figure Błąd pomiędzy algorytmami Monte Carlo a Jacobim dla precyzji 10^(-10) i wzrastającej liczby agentów

Figure Błąd pomiędzy algorytmami Monte Carlo a Jacobim dla precyzji 10^(-14) i wzrastającej liczby agentów

Figure Błąd pomiędzy algorytmami Monte Carlo a Seidlem dla precyzji 10^(-6) i wzrastającej liczby agentów

Figure Błąd pomiędzy algorytmami Monte Carlo a Seidlem dla precyzji 10^(-10) i wzrastającej liczby agentów

Figure Błąd pomiędzy algorytmami Monte Carlo a Seidlem dla precyzji 10^(-14) i wzrastającej liczby agentów

Figure 7 Różnica czasów pomiędzy algorytmami w milisekundach w zależności od rosnącej liczby agentów(testy przeprowadzone dla precyzji 10^(-6))

Figure 8 Różnica czasów pomiędzy algorytmami w milisekundach w zależności od rosnącej liczby agentów(testy przeprowadzone dla precyzji 10^(-10))

Figure 9 Różnica czasów pomiędzy algorytmami w milisekundach w zależności od rosnącej liczby agentów(testy przeprowadzone dla precyzji 10^(-14))

Zakres prac:

Paulina Żurawska: implementacja kodu, tworzenie sprawozdania.

Paweł Szczupak: implementacja algorytmów, tworzenie sprawozdania.