**Projektowanie Algorytmów i Metody Sztucznej Inteligencji**

Projekt II

Paweł Prucnal 248937

02.05.2020r.

Piątek 1315 mgr inż. Marta Emirsajłow

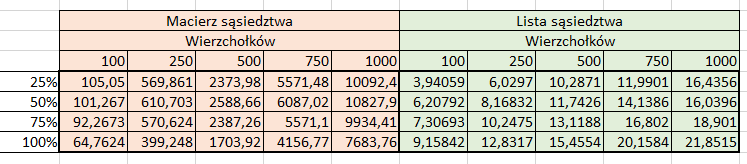
# Wprowadzenie

Celem projektu było zapoznanie się z jednym z algorytmów rozwiązujących problem najkrótszej drogi w grafie. Polega on na znalezieniu połączenia między dwoma wierzchołkami grafu ważonego o najniższej wadze. Problem ten jest często spotykany w sieciach komputerowych czy elektronice. Zdecydowano się na algorytm Dijkstry. Zaimplementowano tenże algorytm dla grafu reprezentowanego przez macierz sąsiedztwa jak i listę sąsiedztwa. Następnie przeprowadzono pomiary czasu dla różnych kombinacji ilości wierzchołków oraz zagęszczenia grafu.

# Przebieg eksperymentów

Testy polegały na zmierzeniu czasu wykonywania algorytmu na losowo wygenerowanych grafach. W tym celu utworzono prosty program generujący macierz sąsiedztwa, w której zawsze istnieje krawędź między wierzchołkiem *n* oraz *n+1*, co daje pewność, że wygenerowany graf jest połączony. Reszta macierzy uzupełniana jest w losowych miejscach losowymi wartościami, by uzyskać odpowiednią gęstość. Przed rozpoczęciem procesu oraz natychmiast po jego zakończeniu program pobierał dokładny czas z komputera przy pomocy biblioteki ctime. Następnie obliczał on różnicę między tymi dwoma punktami czasu w mikrosekundach i zapisywał ją do pliku. Proces ten wykonywany był sto razy dla każdej z kombinacji warunków. Należało bowiem sprawdzić jak algorytm poradzi sobie z grafami składającymi się z 100, 250, 500, 750 oraz 1000 wierzchołków, jak i grafami o gęstości 25, 50, 75 oraz 100%. Przy dwóch badanych reprezentacjach daje to razem 40 kombinacji, dla każdej z których zostało przeprowadzone 100 pomiarów. Wyniki spisane w plikach zaimportowano do programu Microsoft Excel w celu sporządzenia wykresów.

Otrzymano następujące wyniki pomiarów:



# Podsumowanie wyników i wnioski

Niestety algorytm Dijkstry dla listy sąsiedztwa został nieprawidłowo zaimplementowany, a błędu nie udało się znaleźć. W większości przypadków znajduje on takie samie ścieżki, co algorytm na macierzy sąsiedztwa. Niekiedy jednak wyszukuje on drogę dłuższą, lub nie znajduje jej wcale. Prawdopodobnie dzieje się tak, gdyż występuje problem z wyborem krótszej ścieżki, gdy podjęto już inną. Możliwe, że algorytm ten gubi niektóre krawędzie gdy dostanie się do danego wierzchołka przy użyciu innej. Można natomiast założyć, że algorytm na macierzy działa dobrze, gdyż nigdy nie zaistniała sytuacja, by ten drugi znalazł krótszą drogę. Wyniki pomiarów na sporządzonych wykresach układają się w kształt przypominający eksponentę, ponadto porównując średnie czasy np. dla 500 oraz 1000 wierzchołków, zauważamy, że są one 4 razy większe, czyli 22. Złożoność algorytmu Dijkstry na grafie reprezentowanym przez macierz sąsiedztwa wynosi zatem O(n^2), gdzie n to liczba wierzchołków.

# Literatura

* [www.geeksforgeeks.org](http://www.geeksforgeeks.org),
* [www.wikipedia.org](http://www.wikipedia.org),