

Projektowanie Algorytmów i Metody Sztucznej Inteligencji

Projekt 2 - Grafy

Michał Maćkowiak 249464

28.04.2020

Prowadzący:

dr hab. inż. Andrzej Rusiecki

Zajęcia:

Wtorek godz. 13.15

1. Algorytm Bellmana-Forda

Problem najkrótszej drogi w grafie ważonym może być rozwiązany między innymi przez algorytm Bellmana-Forda. Problem polega na znalezieniu najkrótszej ścieżki między dwoma wierzchołkami grafu. W grafach droga o najmniejszej sumie wag drogi między wierzchołkami.

W owym algorytmie dopuszczalne są wagi ujemne, natomiast nie możliwe jest istnienie drogi o koszcie ujemnym.

Złożoność obliczeniowa w pesymistycznym wypadku tego algorytmu wynosi $\mathcal{O}(VE)$, gdzie V oznacza ilość wierzchołków, a E oznacza ilość krawędzi.

Algorytm ten w porównaniu do alg. Dijkstry jest wolniejszy, lecz obsługuje wagi ujemne co w algorytmie Dijkstry jest niemożliwe.

2. Przebieg eksperymentu

Algorytm był testowany dla pięciu różnych wartości wierzchołków:

- 5,
- 10,
- 50,
- 100,
- 500,

oraz dla różnego zagęszczenia grafu:"

- 25%,
- 50%,
- 75%,
- 100% - graf pełny.

Efektywność była badana w zależności od reprezentacji grafu:

- w postaci listy,
- w postaci macierzy.

Dla każdego przypadku generowane było po 100 grafów natomiast czasy zaprezentowane w dalszej części są czasami uśrednionymi.

Program również został wyposażony w odczytywanie grafów z pliku zapisanych w odpowiednim formacie przedstawionym w opisie do projektu. Także posiada opcje do generowania i zapisywania grafu do pliku w tym samym formacie oraz do zapisywania wyniku w pliku.

W prostym menu pojawiającym się włączeniu programy wybieramy opcje nas interesującą. Opcja pierwsza oznacza rozpoczęcie testu czyli generowanie po 100 grafów o każdej gęstości i ilości wierzchołków oraz znajdowaniu najkrótszej ścieżki przy pomocy algorytmu Bellmana-Forda. W wyniku tworzony jest plik „czas_y_lista.txt” z czasami alg. dla grafu w reprezentacji jako lista. Analogicznie plik „czas_y_macierz.txt” dla macierzy.

Opcja druga pozwala wczytać graf z pliku o nazwie „wejście.txt” i znalezieniu w nim najkrótszej ścieżki. W wyniku powstaje plik o nazwie „wynik.txt”, który zawiera najkrótszą ścieżkę do pozostałych wierzchołków oraz koszt drogi.

Opcja trzeci to generator grafów. Tworzy losowy graf o podanych parametrach(liczba wierzchołków, zagęszczenie, początkowy wierzchołek) przez użytkownika. Po skończonym działaniu tworzony jest plik „wynik.txt” z najkrótszą ścieżką do następnych wierzchołków.

UWAGA!

Plik wynik jest nadpisywany przy wybieraniu następnych opcji w menu.

3. Wyniki algorytmu

Czasy wykonywania algorytmu w zależności od gęstości i ilości wierzchołków oraz reprezentacji grafu można zaobserwować w tabelach poniżej.

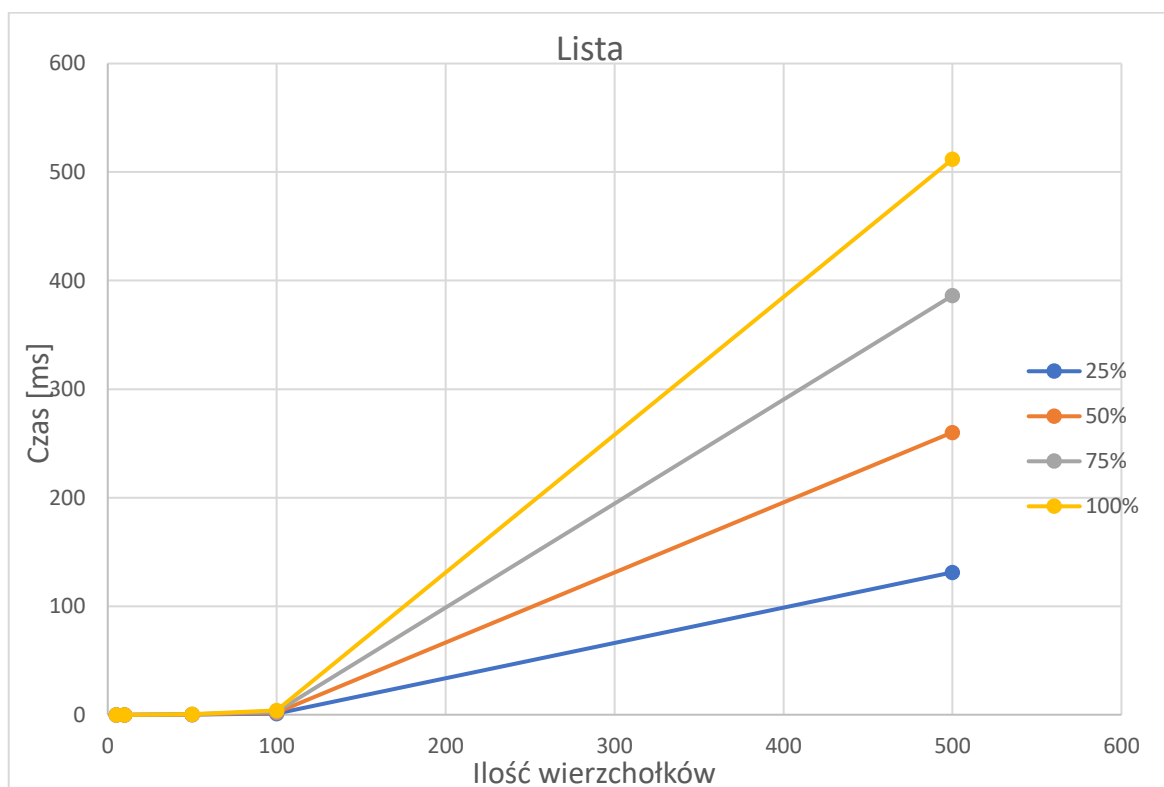
Gęstość grafu	Ilość wierzchołków	Czas wykonania [ms]
25%	5	0,000415
	10	0,001349
	50	0,123363
	100	1,00008
	500	131,202
50%	5	0,000499
	10	0,002052
	50	0,266393
	100	1,99256
	500	260,152
75%	5	0,000608
	10	0,002637
	50	0,356094
	100	2,94127
	500	386,241
100%	5	0,000604
	10	0,003371
	50	0,487277
	100	3,94514
	500	511,874

Tabela 1. Czasy wykonania algorytmu Bellmana-Forda w zależności od gęstości i ilości wierzchołków dla grafu jako lista.

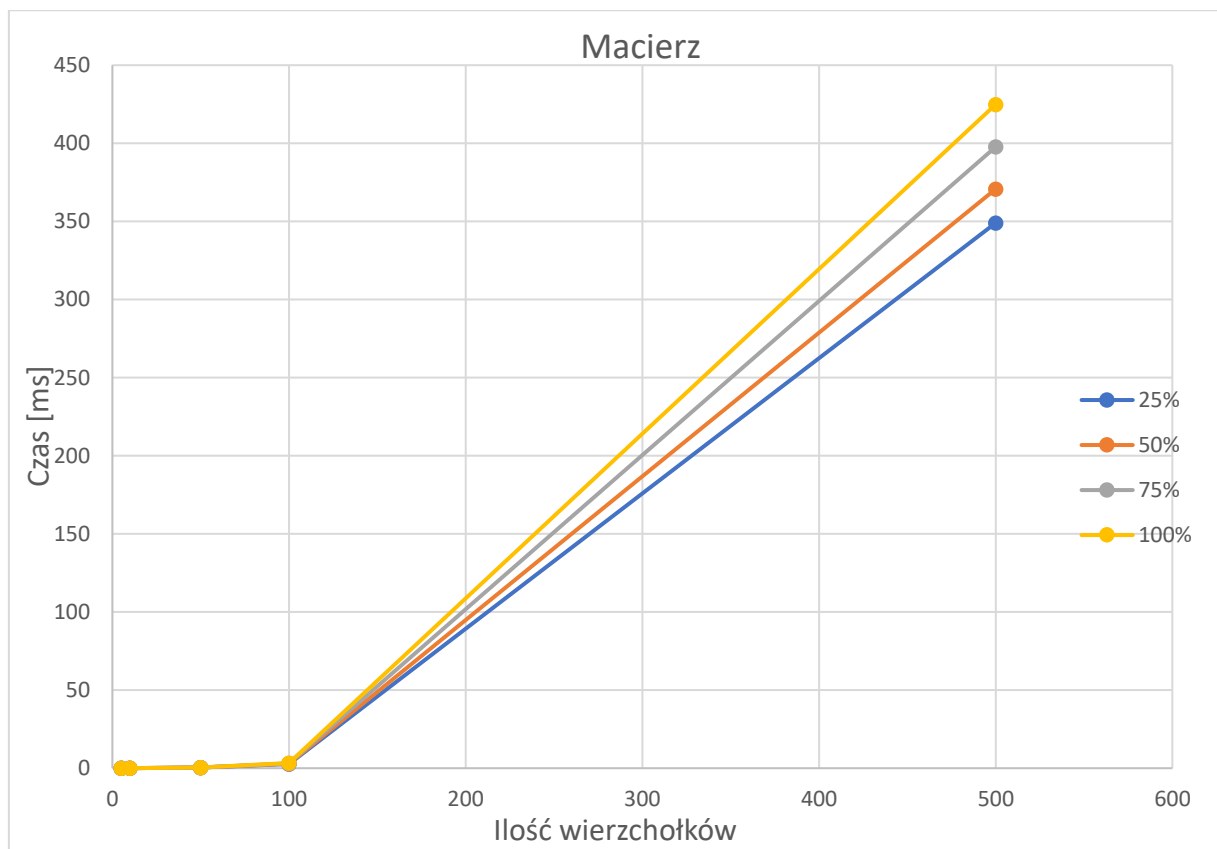
Gęstość grafu	Ilość wierzchołków	Czas wykonania [ms]
25%	5	0,000689
	10	0,003824
	50	0,373195
	100	2,69064
	500	349,037
50%	5	0,0007
	10	0,00379
	50	0,386121
	100	2,92661
	500	370,748
75%	5	0,000707
	10	0,004185
	50	0,405281
	100	3,12473
	500	397,724
100%	5	0,000696
	10	0,003976
	50	0,432119
	100	3,35413
	500	424,874

Tabela 2. Czasy wykonania algorytmu Bellmana-Forda w zależności od gęstości i ilości wierzchołków dla grafu jako macierz.

Na podstawie tabeli 1 oraz 2 narysowano wykresy obrazujące porównanie czasy w zależności od reprezentacji.

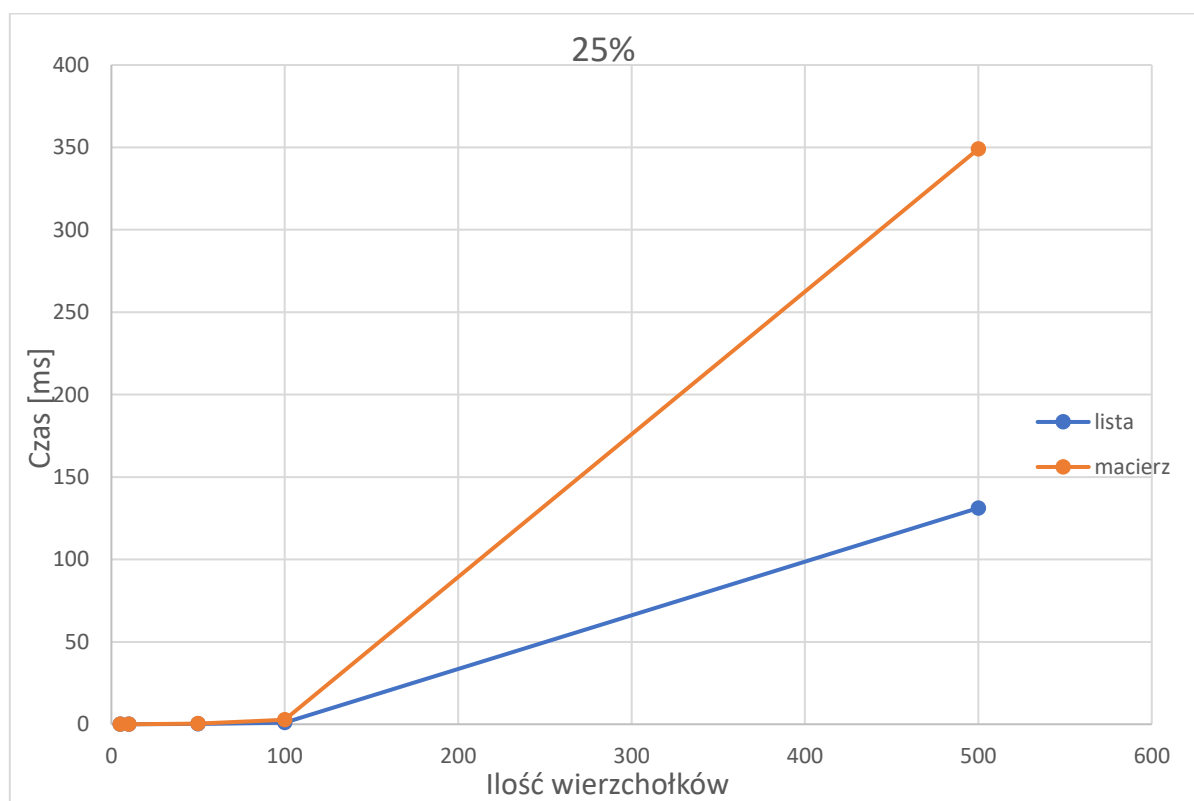


Wykres 1. Czas wykonania w zależności od gęstości dla reprezentacji jako lista.

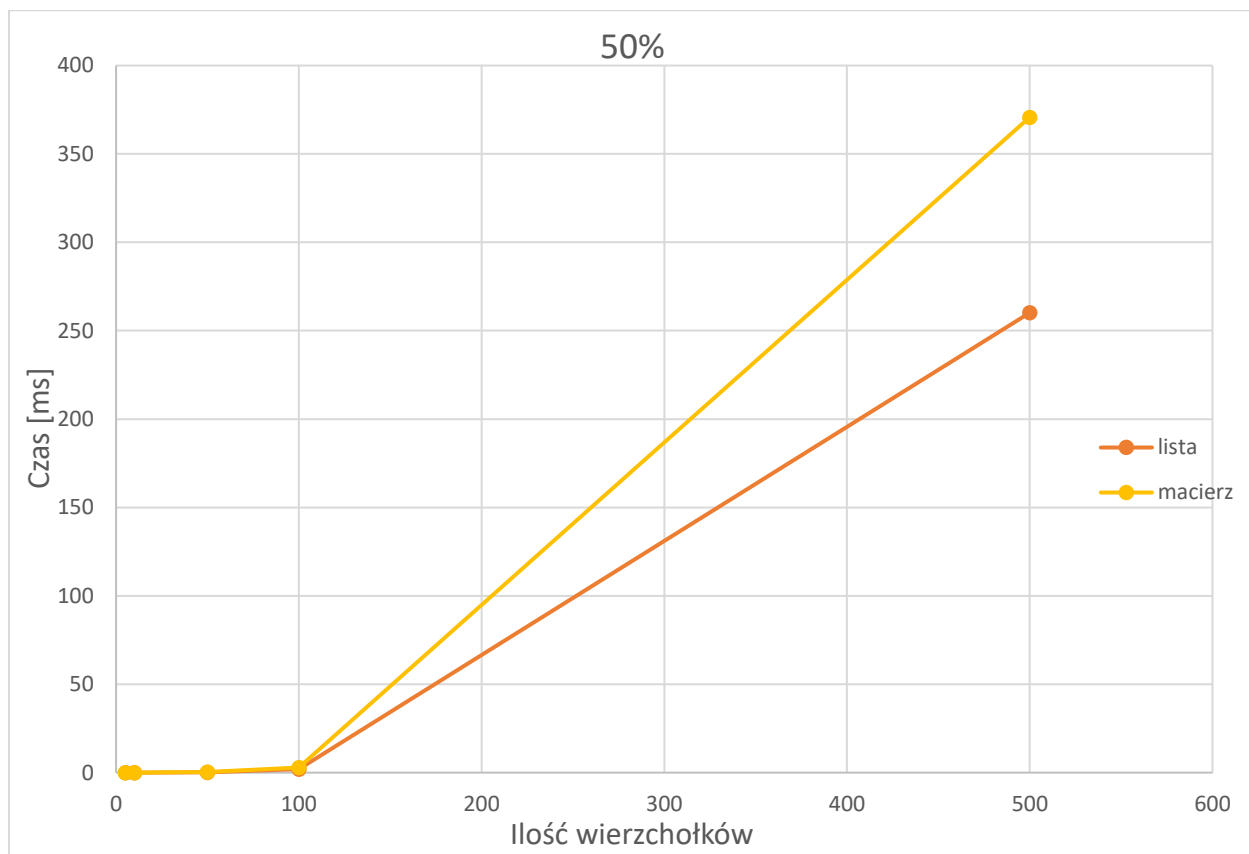


Wykres2 . Czas wykonania w zależności od gęstości dla reprezentacji jako macierz.

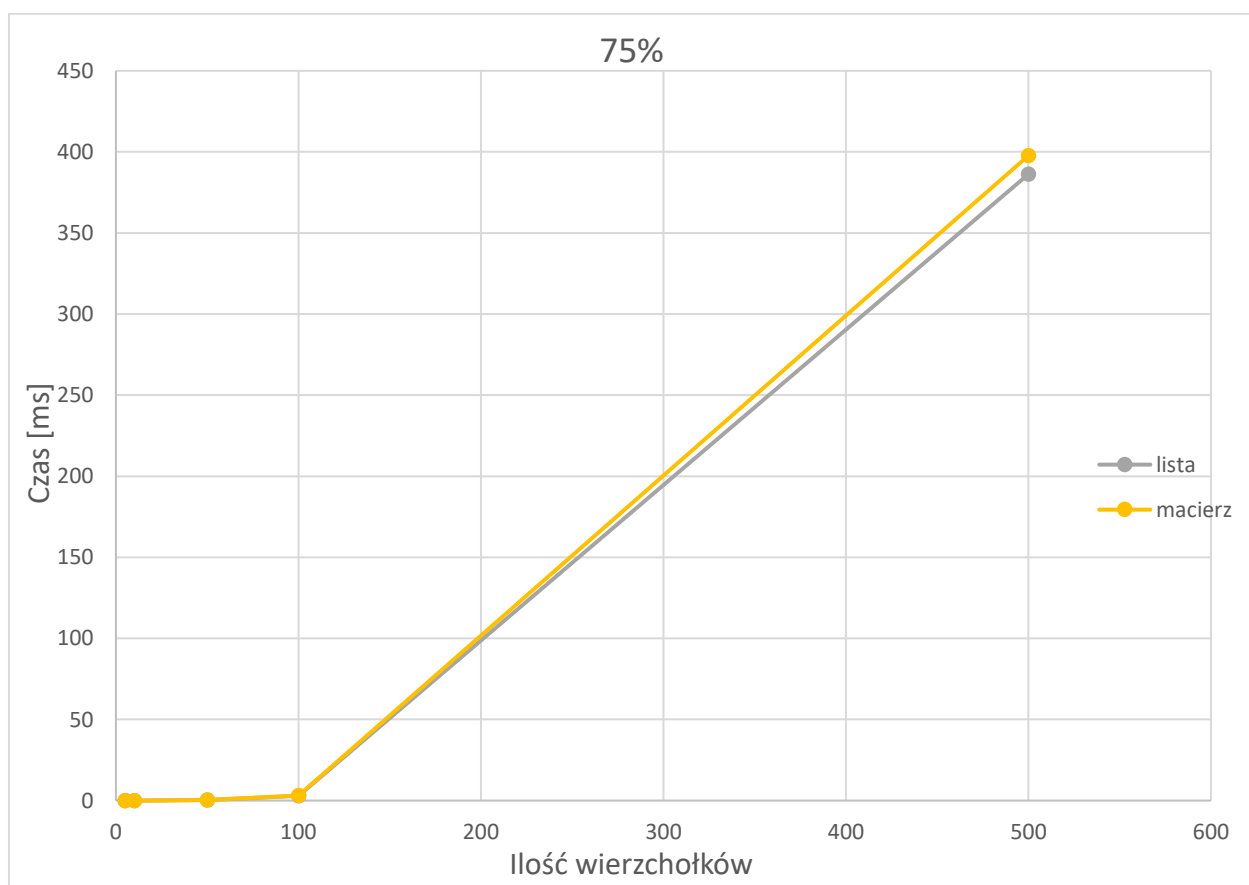
Wyniki z tabel można również przedstawić na wykresach porównujących czas w zależności od gęstości.



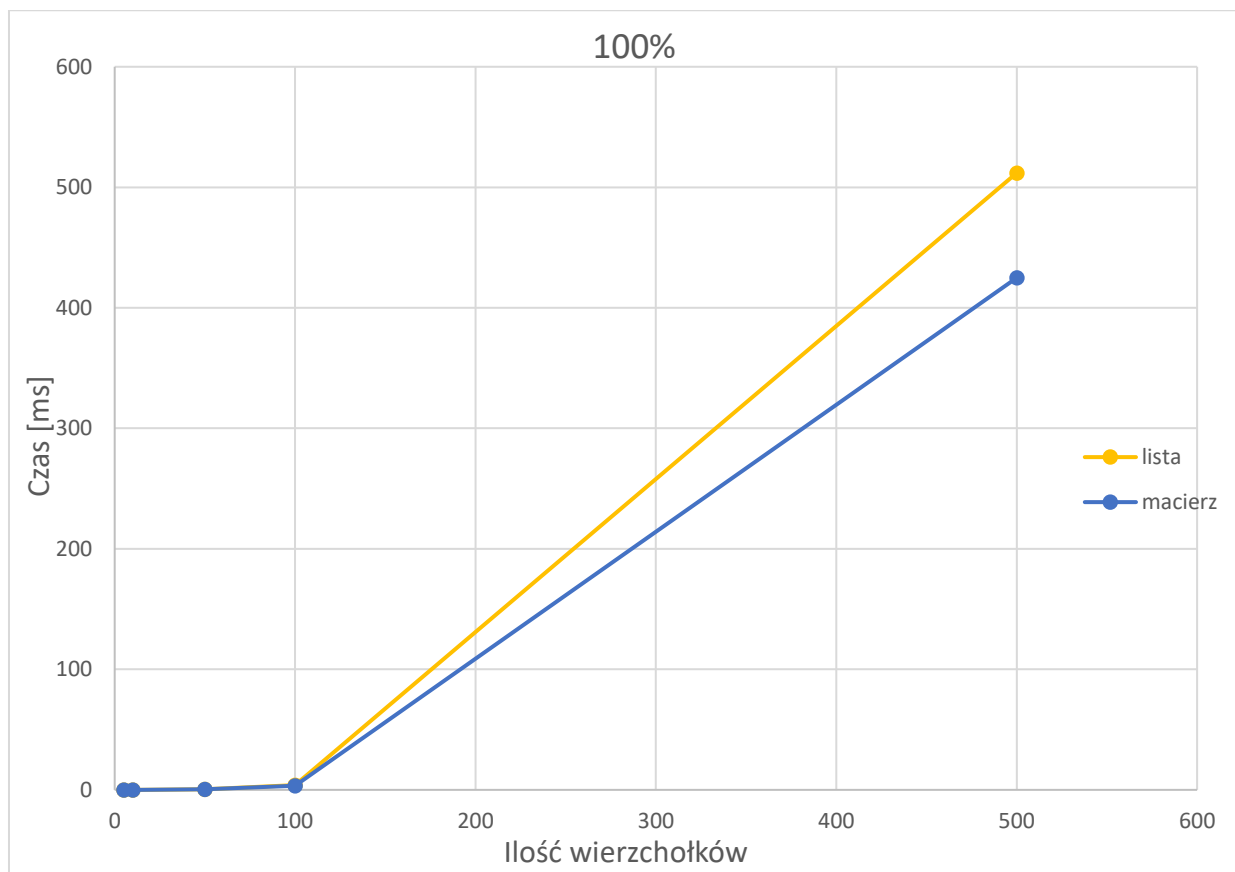
Wykres 3. Czas wykonania dla zagęszczenia 25%.



Wykres 4. Czas wykonania dla zagęszczenia 50%.



Wykres 5. Czas wykonania dla zagęszczenia 75%.



Wykres 6. Czas wykonania dla zagęszczenia 100%.

4. Wnioski

W grafie reprezentowanym jako macierz długość wykonania miała niewielkie różnice w zależności od gęstości. Różnica dla listy jest zauważalna. Wynika to z faktu, że złożoność obliczeniowa dla listy sąsiedztwa zależy od ilości wierzchołków jak i ilości krawędzi $\mathcal{O}(VE)$. Natomiast dla macierzy zależy tylko od ilości wierzchołków $\mathcal{O}(V^3)$. Zatem gęstość nie ma znaczenia przy reprezentacji grafu jako macierz. Można to zaobserwować na wykresie 1 oraz 2.

Jednakże ilość wierzchołków ma znaczenie w obu reprezentacjach. Przewidywanie było, że algorytm wykona się szybciej przy reprezentacji jako lista. W większości przypadków taka sytuacja miała miejsce.

Na wykresie 5 można zauważyć że czasy są dość zbliżone do siebie. A na wykresie 6, że czas dla macierzy jest szybszy niż w przypadku listy. Wynika to z faktu, że przy dużej ilości wierzchołków oraz dużym zagęszczeniu czas znajdowania najkrótszej drogi jest wydłużany dla listy, lecz nie ma takiego znaczenia dla macierzy.

Na czasy wykonania algorytmu miał też generator grafów który mógł w pewnych przypadkach przyspieszyć czas wykonania algorytmu.

5. Bibliografia

https://eduinf.waw.pl/inf/alg/001_search/0138a.php

https://pl.wikipedia.org/wiki/Algorytm_Bellmana-Forda

<https://www.geeksforgeeks.org/bellman-ford-algorithm-dp-23/>