

Projekt P.A.M.S.I.

Jan Bronicki

Piątek 13:15-15:00

1. Wprowadzenie

Celem zadania było napisanie samemu programu wykonującego stworzenie grafu używając dwóch metod, Macierzowej oraz Listowej. Następnie stworzenie algorytmu Bellmana-Ford'a w celu znalezienia najkrótszej ścieżki z jednego wierzchołka grafu do innego danego wierzchołka grafu i zmierzenie wydajności metod.

2. Algorytm

Używamy tutaj algorytmu Bellmana-Forda, który wykorzystuje metodę relaksacji do wyznaczenia najkrótszej ścieżki, dlatego metoda polega na sprawdzaniu, gdy przechodzimy przez daną „krawędź” nie znajdziemy najkrótszej ścieżki. Bellman'owy algorytm jest bardziej złożony obliczeniowo niżeli ten Dijkstry, ale umożliwi znalezienie ścieżki również kiedy krawędzie posiadają wagi ujemne. Minusem jest to, że algorytm Bellmana jest ograniczony możliwością wystąpienia cyklu w grafie który jest ujemny.

Jego złożoność obliczeniowa, dla różnych implementacji:

- Macierz: $O(W^3)$
- Lista: $O(W * K)$

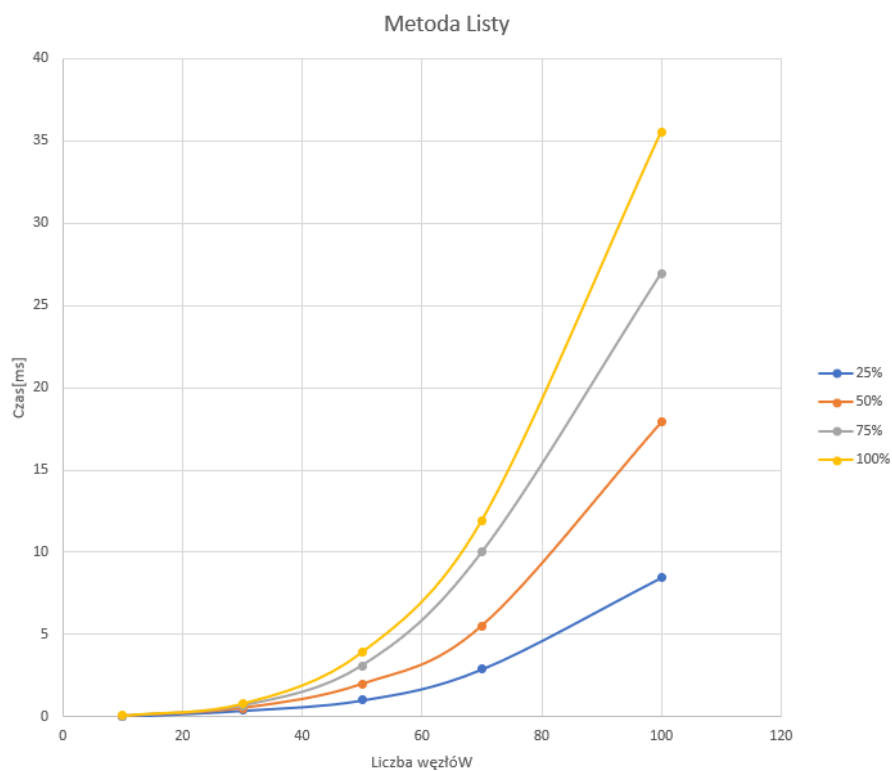
Gdzie W – wierzchołki, K – krawędzie

3. Wyniki Eksperymentów

Oto otrzymane wyniki eksperymentów:

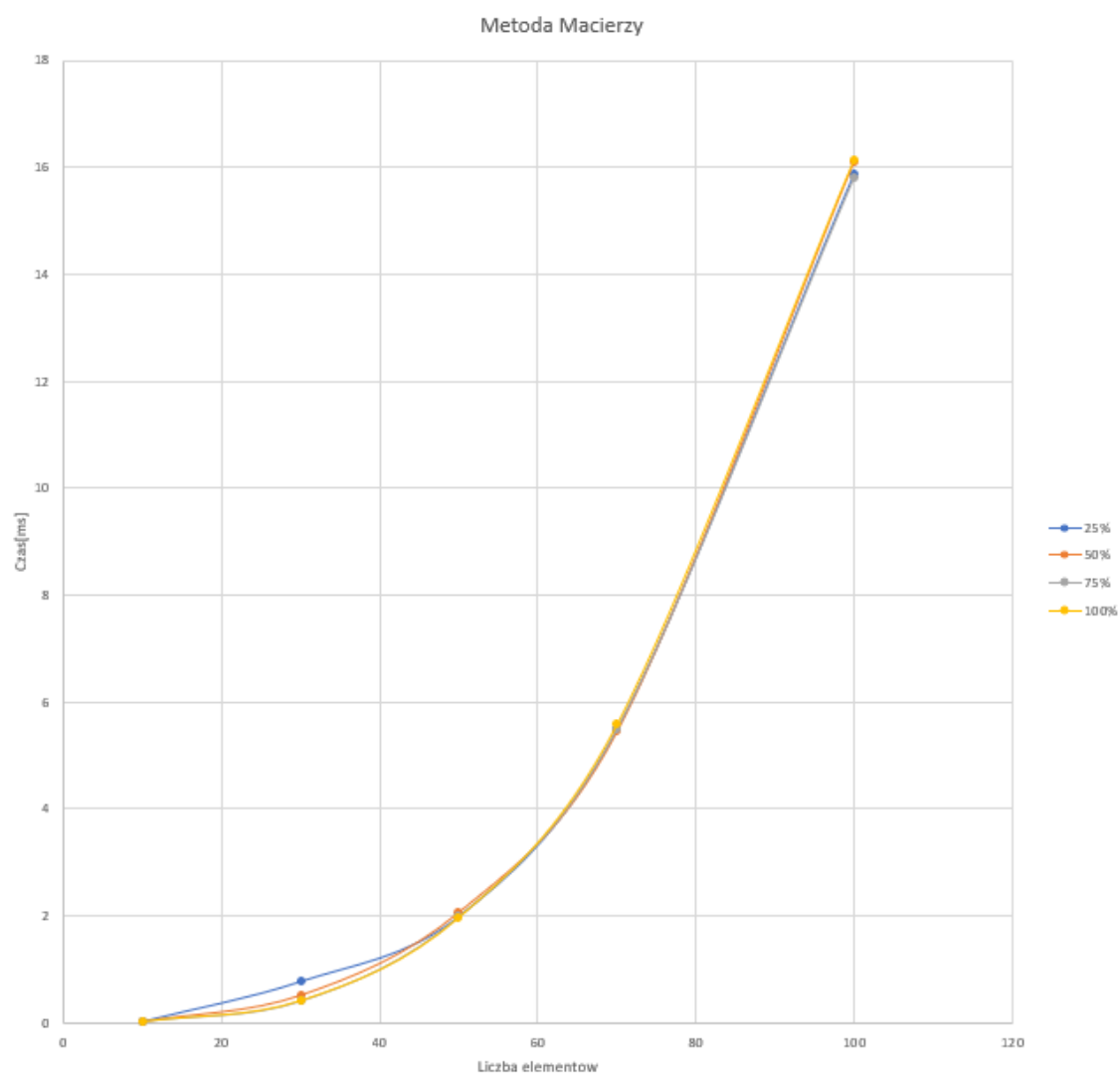
1. Listy

Metoda Listy		Gęstość			
		0.25	0.5	0.75	1
Liczba elementów	10	0.01686	0.02807	0.03818	0.09024
	30	0.38764	0.50971	0.66802	0.8152
	50	1.01331	1.97756	3.10737	3.94732
	70	2.88813	5.51966	9.99706	11.9505
	100	8.44205	17.9176	26.9655	35.5397

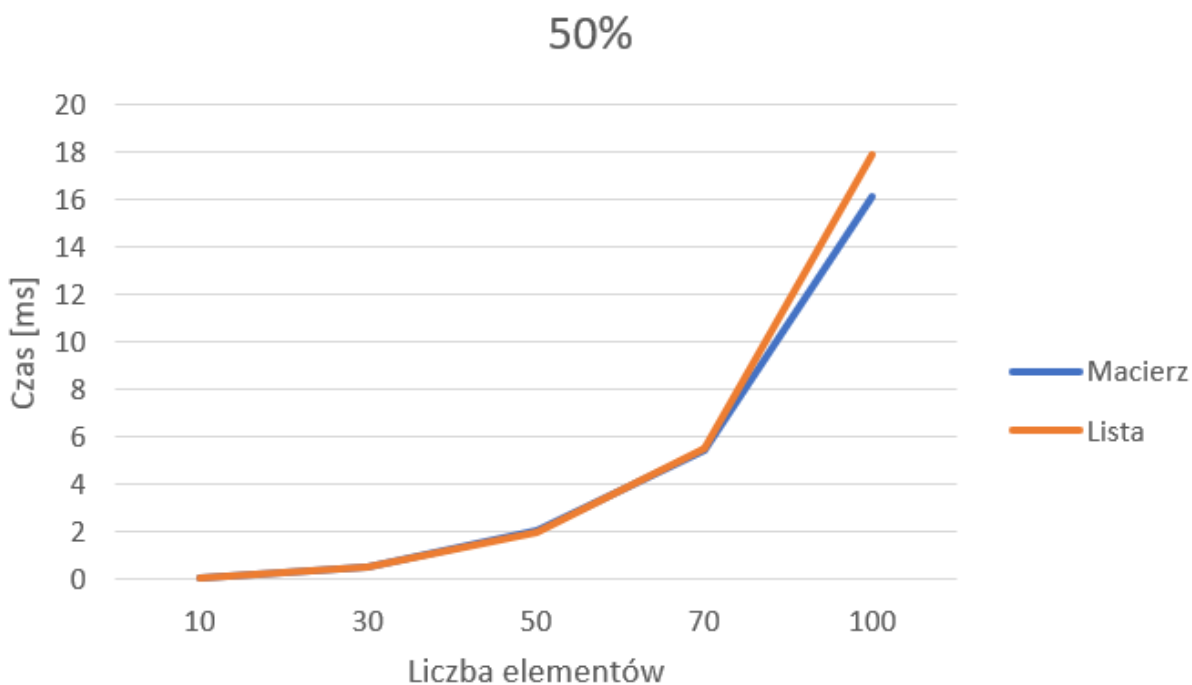
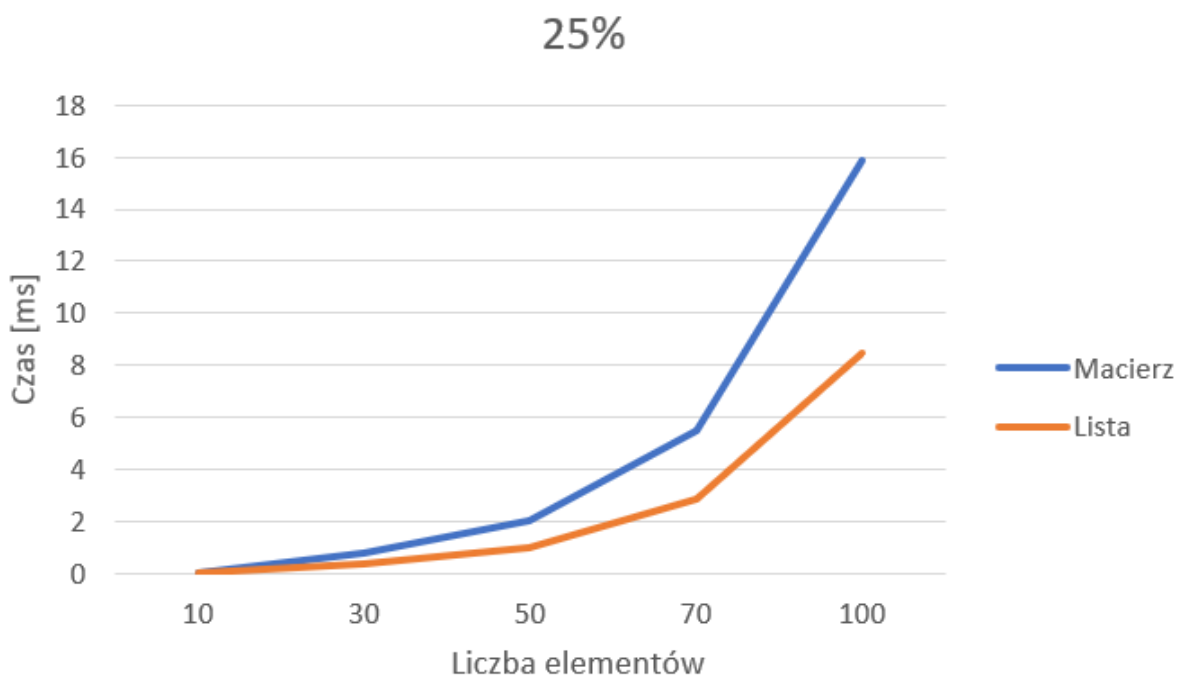


2. Macierze

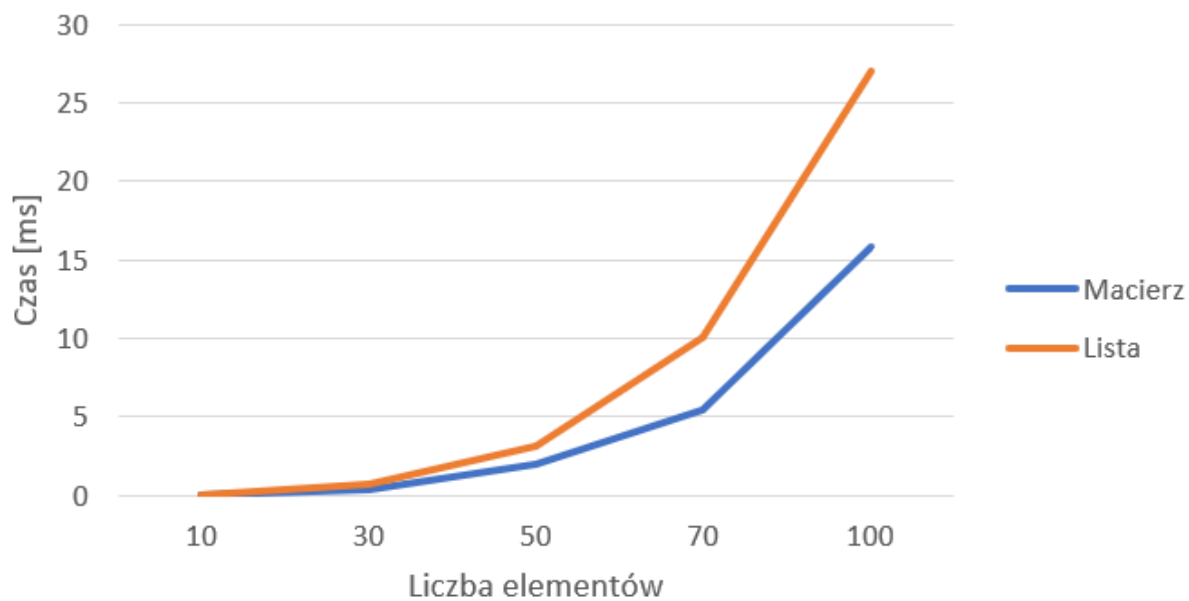
Metoda Macierzy		Gęstość			
		0.25	0.5	0.75	1
Liczba elementów	10	0.02863	0.02883	0.02898	0.03707
	30	0.77636	0.52038	0.40977	0.42676
	50	1.9944	2.07121	1.98881	1.97993
	70	5.48484	5.4459	5.49647	5.57908
	100	15.8848	16.1146	15.8247	16.1342



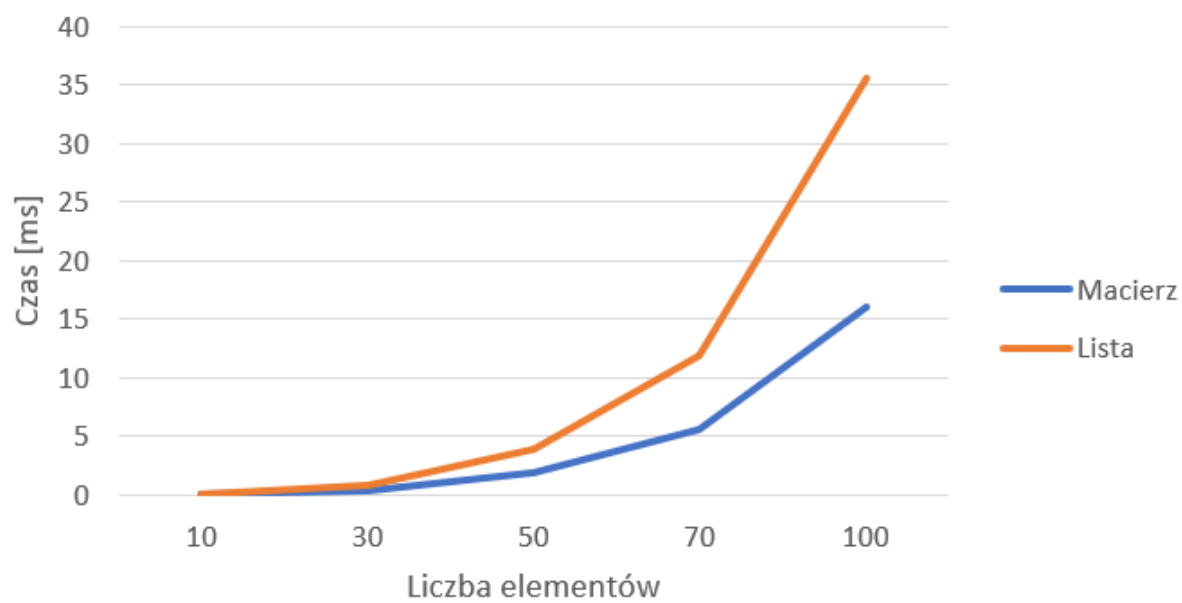
Teraz porównanie bezpośrednie między listami a macierzami:



75%



100%



4. Wnioski

- Efektywności metody Listowej spada wraz ze wzrostem liczby krawędzi ze względu na złożoność obliczeniową
- Graf oparty na metodzie Listowej jest lepszy, dla grafu o małej ilości krawędzi
- Czas, który algorytm potrzebuje na znalezienie drogi, dla grafu metody macierzowej nie zależy od ilości krawędzi
- Do gęstości 50% obie metody produkują niemal identyczne wyniki

Źródła:

- <https://www.geeksforgeeks.org/bellman-ford-algorithm-dp-23/>
- https://en.wikipedia.org/wiki/Bellman%E2%80%93Ford_algorithm
- <https://www.geeksforgeeks.org/detect-negative-cycle-graph-bellman-ford/>
- https://www-m9.ma.tum.de/graph-algorithms/spp-bellman-ford/index_en.html
- <https://brilliant.org/wiki/bellman-ford-algorithm/>
- <https://www.programiz.com/dsa/bellman-ford-algorithm>
- https://eduinf.waw.pl/inf/alg/001_search/0138a.php