

Sprawozdanie – Projektowanie algorytmów i metody sztucznej inteligencji, Projekt 2

Data oddania sprawozdania: 15.05.2019r.

Autor: Tomasz Filip 241630

Prowadzący: dr inż. Łukasz Jeleń

Termin zajęć: środa 11:15

1. Wprowadzenie

Zadanie polegało na zaimplementowaniu oraz przetestowaniu jednego z algorytmów służących do znalezienia najkrótszego połączenia między danym wierzchołkiem, a pozostałymi wierzchołkami. Wybrano algorytm Dijkstry i przetestowano jego działanie dla reprezentacji grafu w postaci macierzy sąsiedztwa i listy sąsiedztwa; dla następujących ilości wierzchołków: 10, 50, 100, 500, 1000; dla gęstości grafu: 25%, 50%, 75% oraz dla grafu pełnego.

2. Zastosowane algorytmy

Algorytm Dijkstry- algorytm służący do znajdowania najkrótszej ścieżki z danego wierzchołka do pozostałych w grafie o nieujemnych wagach krawędzi. Algorytm tworzy kolejkę priorytetową wszystkich wierzchołków grafu, gdzie priorytetem jest aktualnie wyliczona odległość od wierzchołka źródłowego. Następnie algorytm zdejmuję kolejno wierzchołki najbliższe źródła, które nie zostały jeszcze rozważone i dokonuje ewentualnych zmian w tablicy kosztów przejścia między wierzchołkami.

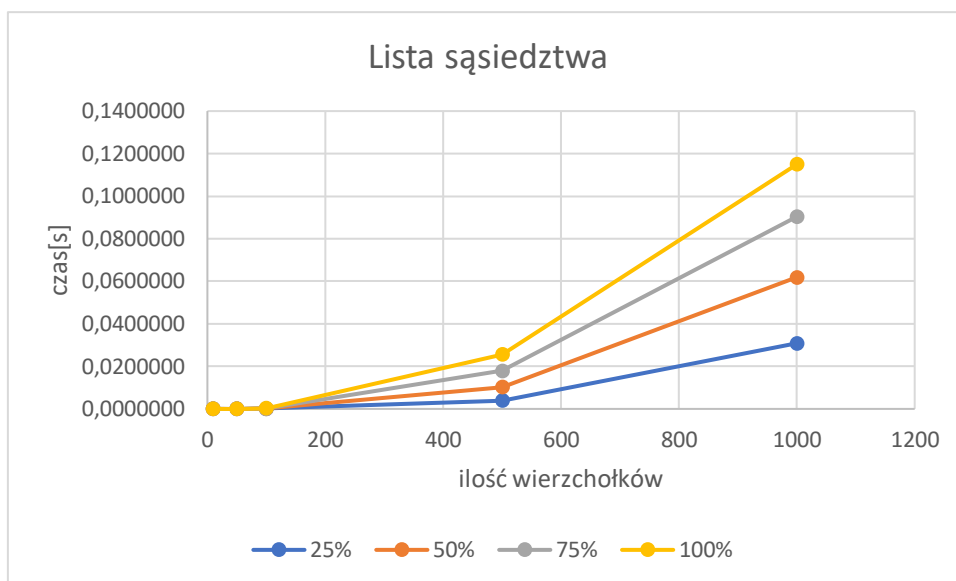
3. Testy algorytmów

Tabela 1. Testy algorytmu Dijkstry dla grafu w postaci listy sąsiedztwa

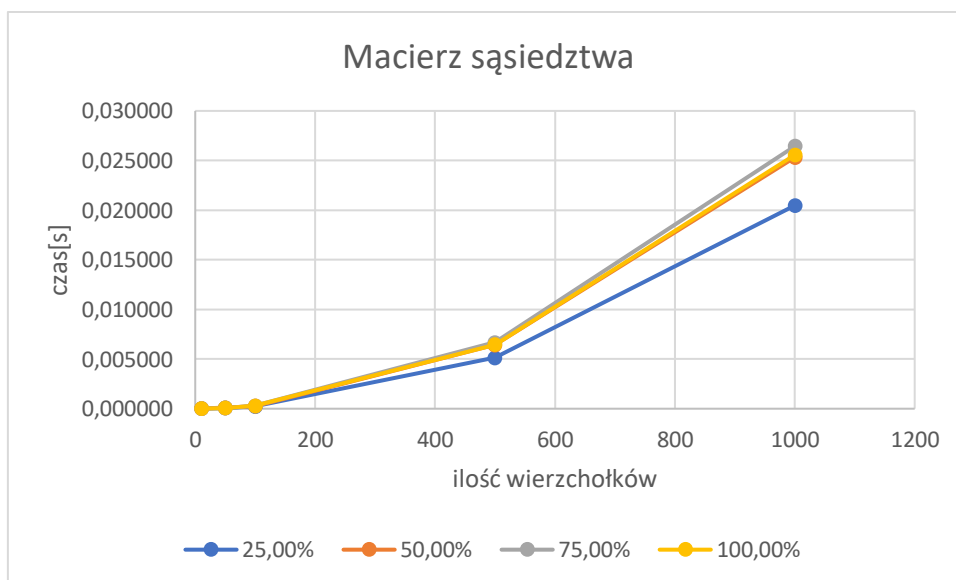
Lista sąsiedztwa					
wierzchołków:	10	50	100	500	1000
gęstość:	czas[s]				
25,00%	0,0000064	0,000037	0,00012	0,0039	0,031
50,00%	0,0000069	0,000042	0,00016	0,010	0,062
75,00%	0,0000075	0,000056	0,00020	0,018	0,090
100,00%	0,0000084	0,000058	0,00025	0,026	0,11

Tabela 2. Testy algorytmu Dijkstry dla grafu w postaci macierzy sąsiedztwa

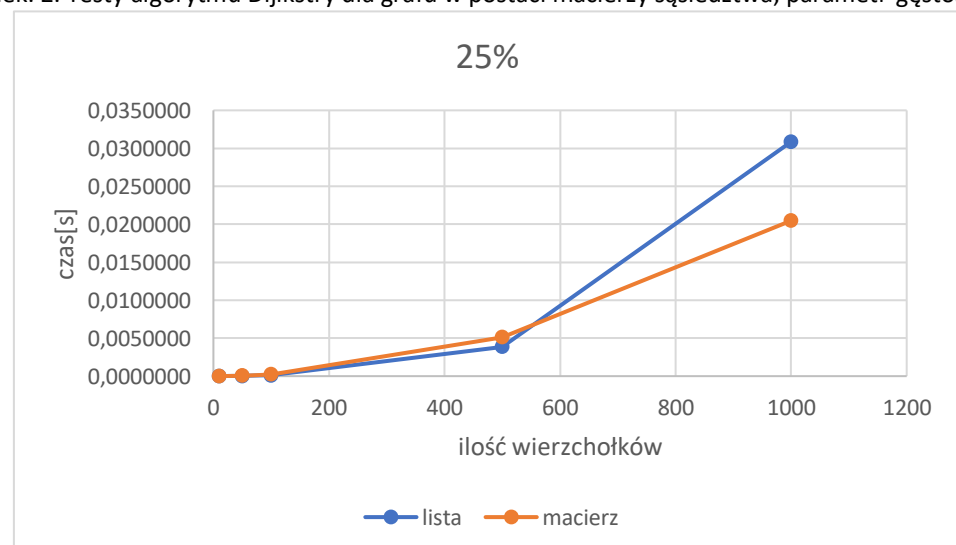
Macierz sąsiedztwa					
wierzchołków:	10	50	100	500	1000
gęstość:	czas[s]				
25,00%	0,000009	0,000081	0,00022	0,0051	0,020
50,00%	0,000011	0,000086	0,00028	0,0064	0,025
75,00%	0,000012	0,000084	0,00029	0,0067	0,026
100,00%	0,000013	0,000084	0,00028	0,0065	0,026



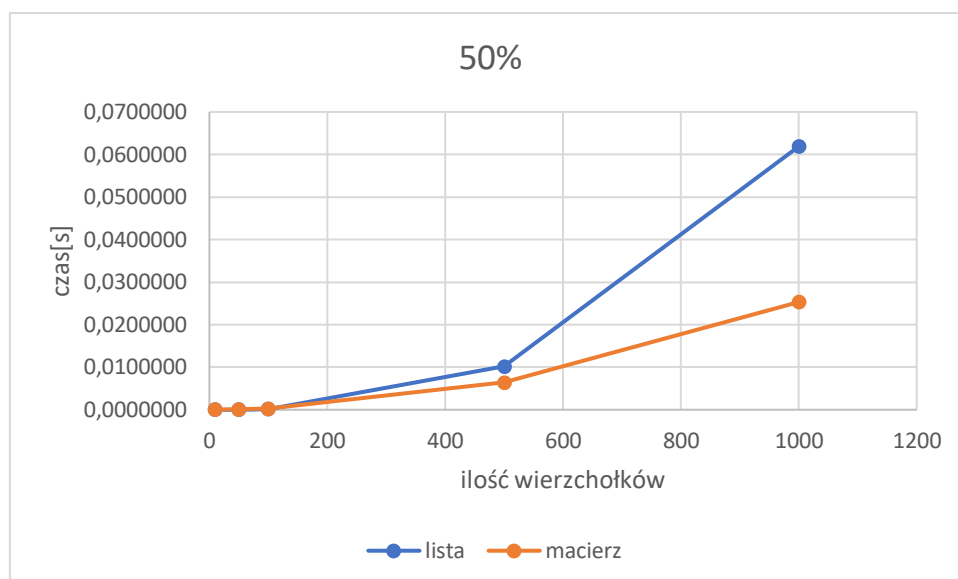
Rysunek. 1. Testy algorytmu Dijkstry dla grafu w postaci listy sąsiedztwa, parametr-gęstość grafu



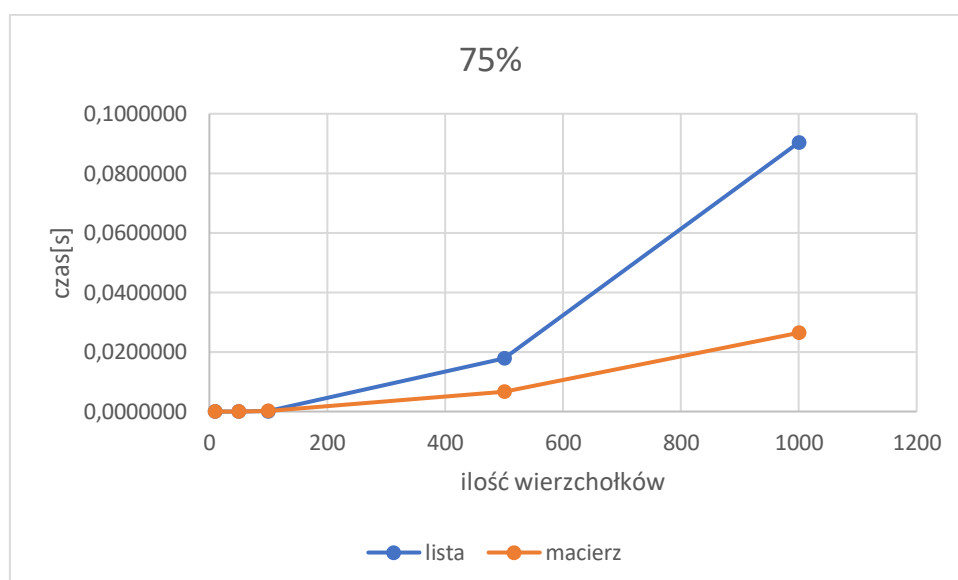
Rysunek. 2. Testy algorytmu Dijkstry dla grafu w postaci macierzy sąsiedztwa, parametr-gęstość grafu



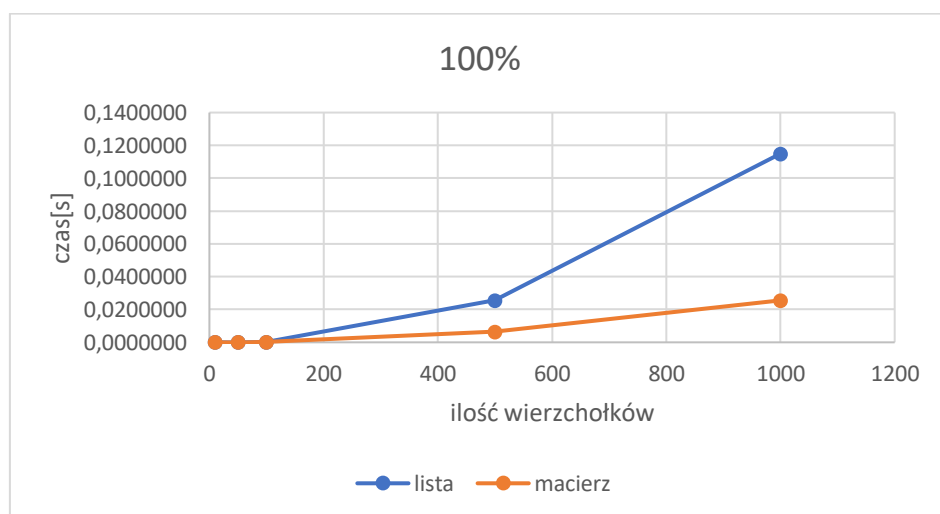
Rysunek. 3. Testy algorytmu Dijkstry dla grafu gęstości 25%, parametr-reprezentacja grafu



Rysunek. 4. Testy algorytmu Dijkstry dla grafu gęstości 50%, parametr-reprezentacja grafu



Rysunek. 5. Testy algorytmu Dijkstry dla grafu gęstości 75%, parametr-reprezentacja grafu



Rysunek. 6. Testy algorytmu Dijkstry dla grafu gęstości 100%, parametr-reprezentacja grafu

4. Podsumowanie i wnioski

Jak można zaobserwować złożoność obliczeniowa algorytmu w przypadku implementacji w postaci listy sąsiedztwa jest niższa niż w przypadku macierzy sąsiedztwa tylko dla małych rozmiarów grafów. Przy 500 wierzchołkach zazwyczaj algorytm Dijkstry działa szybciej dla implementacji w postaci macierzy. Dla 1000 wierzchołków różnica ta jest już znacząca. Inną obserwacją jest bardzo mały wzrost złożoności obliczeniowej dla rosnącej gęstości grafu przy implementacji w postaci macierzy sąsiedztwa. Dodatkowym plusem tej reprezentacji jest łatwość zrozumienia działania algorytmu.

Źródła:

[1] https://pl.wikipedia.org/wiki/Algorytm_Dijkstry [dostęp 15.05.2019]

[2] https://eduinf.waw.pl/inf/alg/001_search/0138.php [dostęp 15.05.2019]