

# Struktury Danych i Złożoność Obliczeniowa

Autor: Alicja Wróbel

## 1. Wstęp

Celem projektowym było zapoznanie się z problemem wyznaczania minimalnego drzewa rozpinającego, oraz wyszukiwania najkrótszej ścieżki w grafie. Program zawiera implementacje następujących algorytmów:

- Algorytm Kruskala –  $O(E * \log V)$
- Algorytm Prima –  $O(E * \log V)$
- Algorytm Dijkstry –  $O(E * \log V)$
- Algorytm Bellmana-Forda –  $O(|V| * |E|)$

Minimalne drzewo rozpinające – drzewo zawierające wszystkie wierzchołki grafu oraz krawędzie, których suma jest o najniższej wadze z wszystkich pozostałych drzew rozpinających danego grafu. W projekcie wykorzystujemy do tego algorytm Kruskala oraz Prima.

Najkrótsza ścieżka w grafie – jest to ścieżka, w której dojście z danego punktu do innego jest o najniższym koszcie przejścia. W projekcie wykorzystujemy do tego algorytm Bellmana-Forda oraz Dijkstry.

Do obliczeń została wykorzystana funkcja `QueryPerformanceCounter()`, w klasie „Timer.h”. Każdy pomiar został wykonany dla innego zestawu danych losowych dla danego grafu. W przedstawionych poniżej pomiarach jest wartość średnia 100 pomiarów dla różnej ilości gęstości oraz liczby wierzchołków.

## 2. Pomiary:

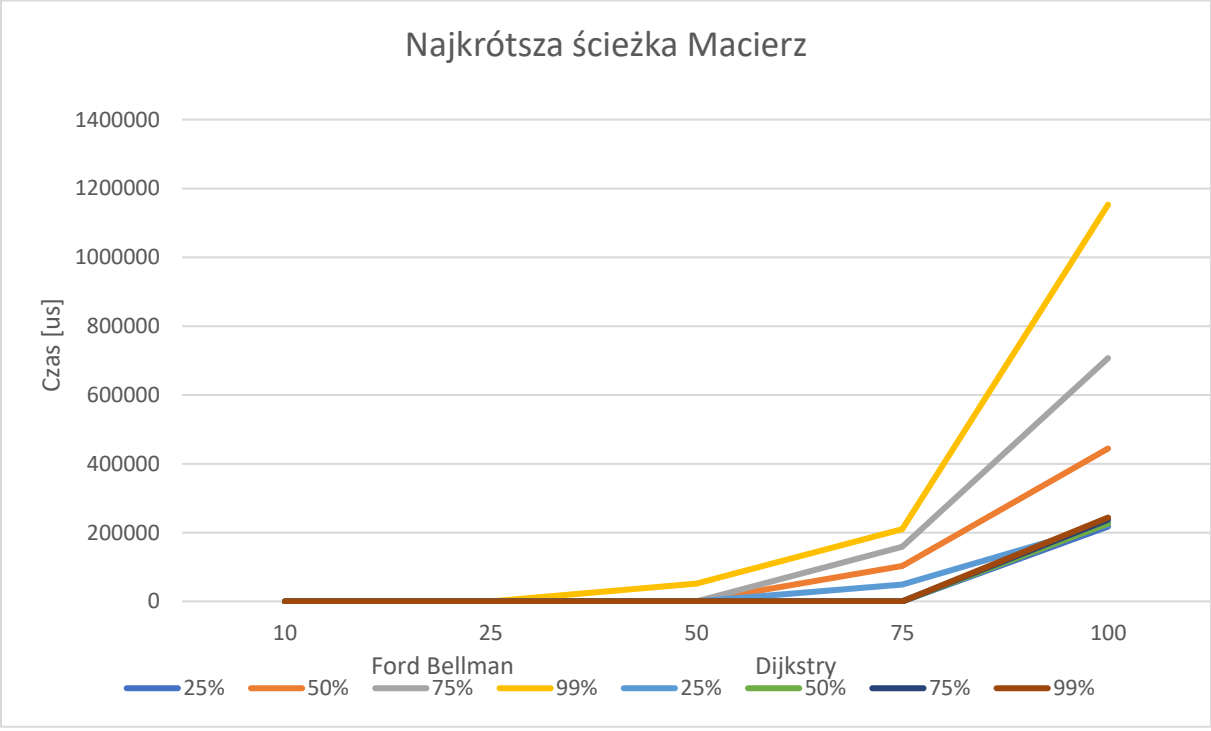
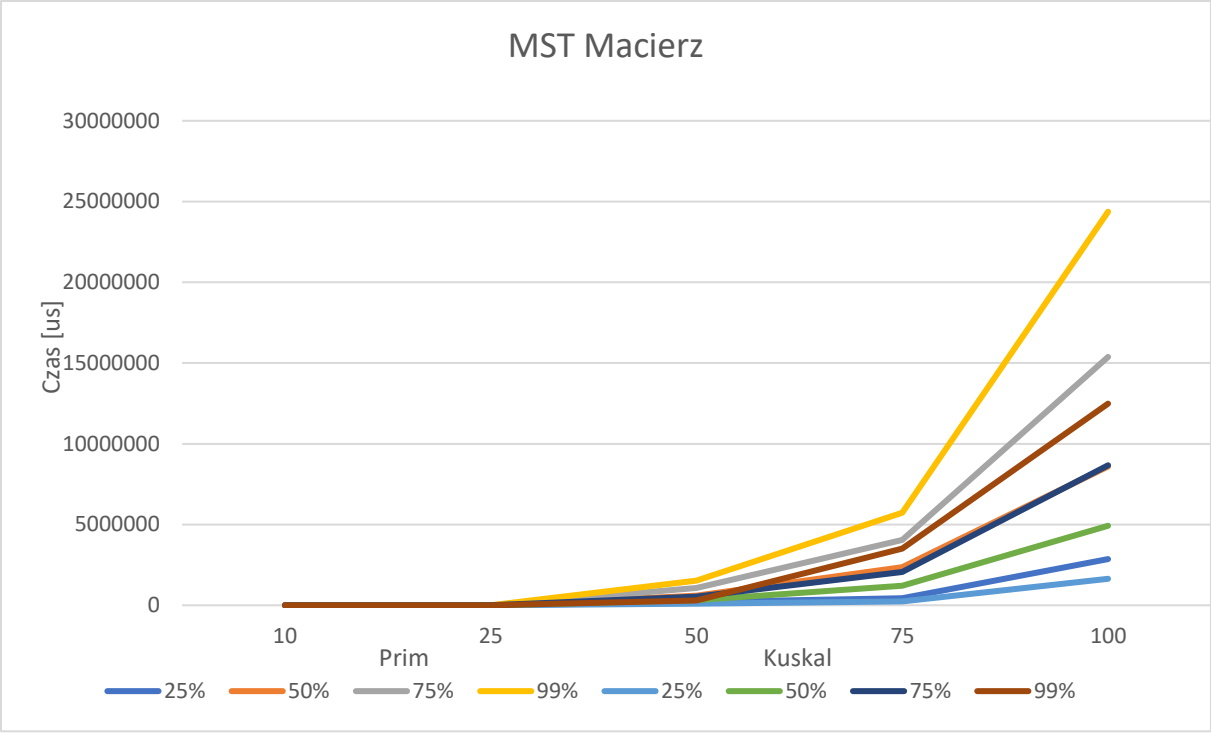
## 2.1 Macierz

PRIM				
wierzchołki	25%	50%	75%	99%
10	456.076	1623.26	1849.35	3196.68
25	14240.3	43045.7	73348.7	95631.9
50	204812	599771	1071690	1518780
75	434089	2365610	4053080	5724670
100	2859630	8595020	15377600	24369300

KRUSKAL				
wierzchołki	25%	50%	75%	99%
10	356.149	801.839	1107.7	1691.74
25	8391.17	22143.1	37648.5	50426.8
50	106278	315800	528331	791955
75	229002	1203640	2070350	295510
100	1638620	4921480	8666680	12484400

BELLMAN FORD				
wierzchołki	25%	50%	75%	99%
10	13.5891	33.2953	43.1074	62.0336
25	804.795	1639.19	2419.72	3514.86
50	12472.6	24755.3	38861.8	51702
75	26022.2	103326	158463	209792
100	217460	444237	706864	1152990

DIJKSTRY				
wierzchołki	25%	50%	75%	99%
10	313.411	314.971	319.077	333.857
25	3686.92	3685.03	3770.71	3777.11
50	27833.6	28272.4	28353.9	29529.4
75	48759	77782.7	78195.3	80793.2
100	223880	226281	237137	243682



## 2.2 Lista

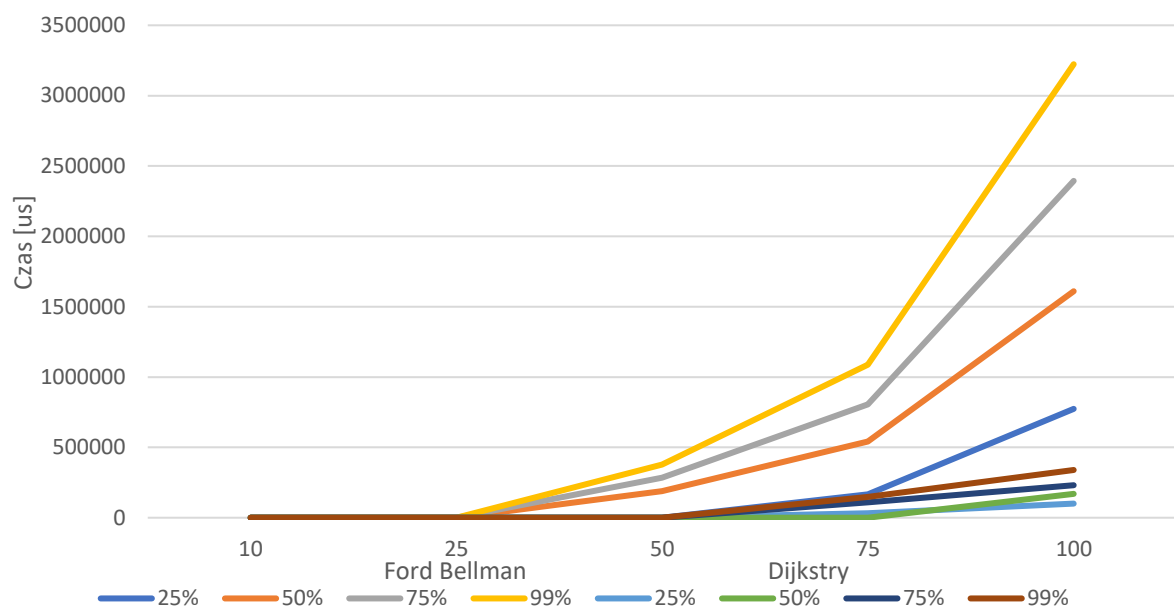
PRIM				
wierzchołki	25%	50%	75%	99%
10	1680.41	2703.08	4638.19	5420.98
25	29014.6	73120.8	125071	182225
50	369520	1015840	1851100	2351230
75	783059	3707280	6713890	9188480
100	5054620	14166100	24506100	37780100

KRUSKAL				
wierzchołki	25%	50%	75%	99%
10	698	1136	158855	210672
25	20971	50526	806724	106694
50	36012.2	147485	218783	288845
75	808450	308912	464506	604534
100	2911690	548075	1013003	1082290

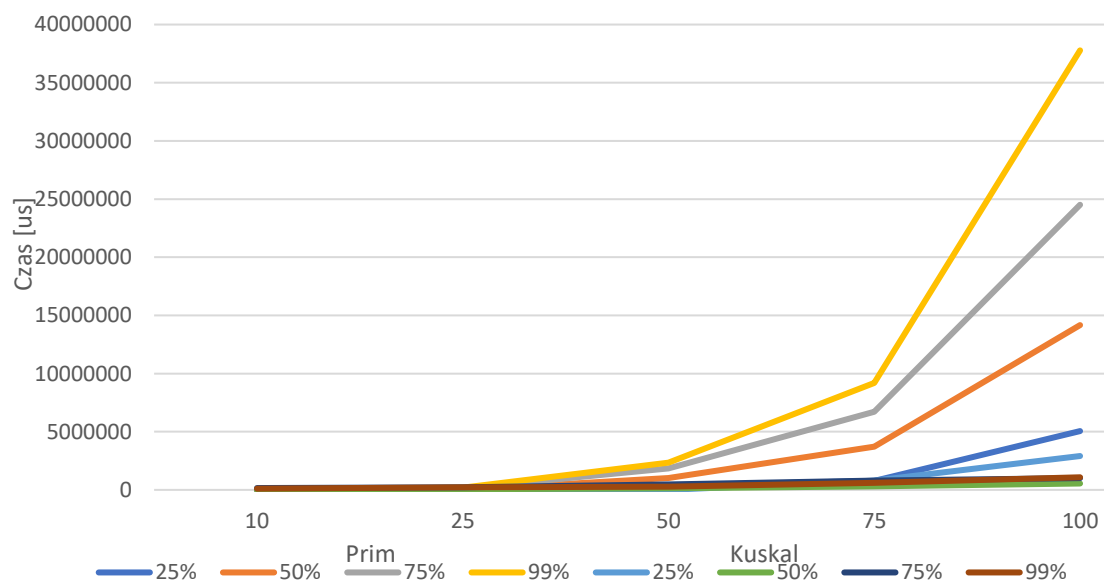
BELLMAN FORD				
wierzchołki	25%	50%	75%	99%
10	780.408	1303.86	1973.95	2590.59
25	11363.1	34031.8	44986.5	44986.5
50	94689.9	189260	284675	376901
75	167191	541820	806096	1088450
100	77397	1609740	2394020	3223330

DIJKSTRY				
wierzchołki	25%	50%	75%	99%
10	1031.13	1641.94	2392.09	3141.83
25	4978.21	9974.4	14295.7	18390.3
50	19289.1	38353.7	56697.9	73700.9
75	32413	79053.5	110248	146835
100	100313	169632	230958	339308

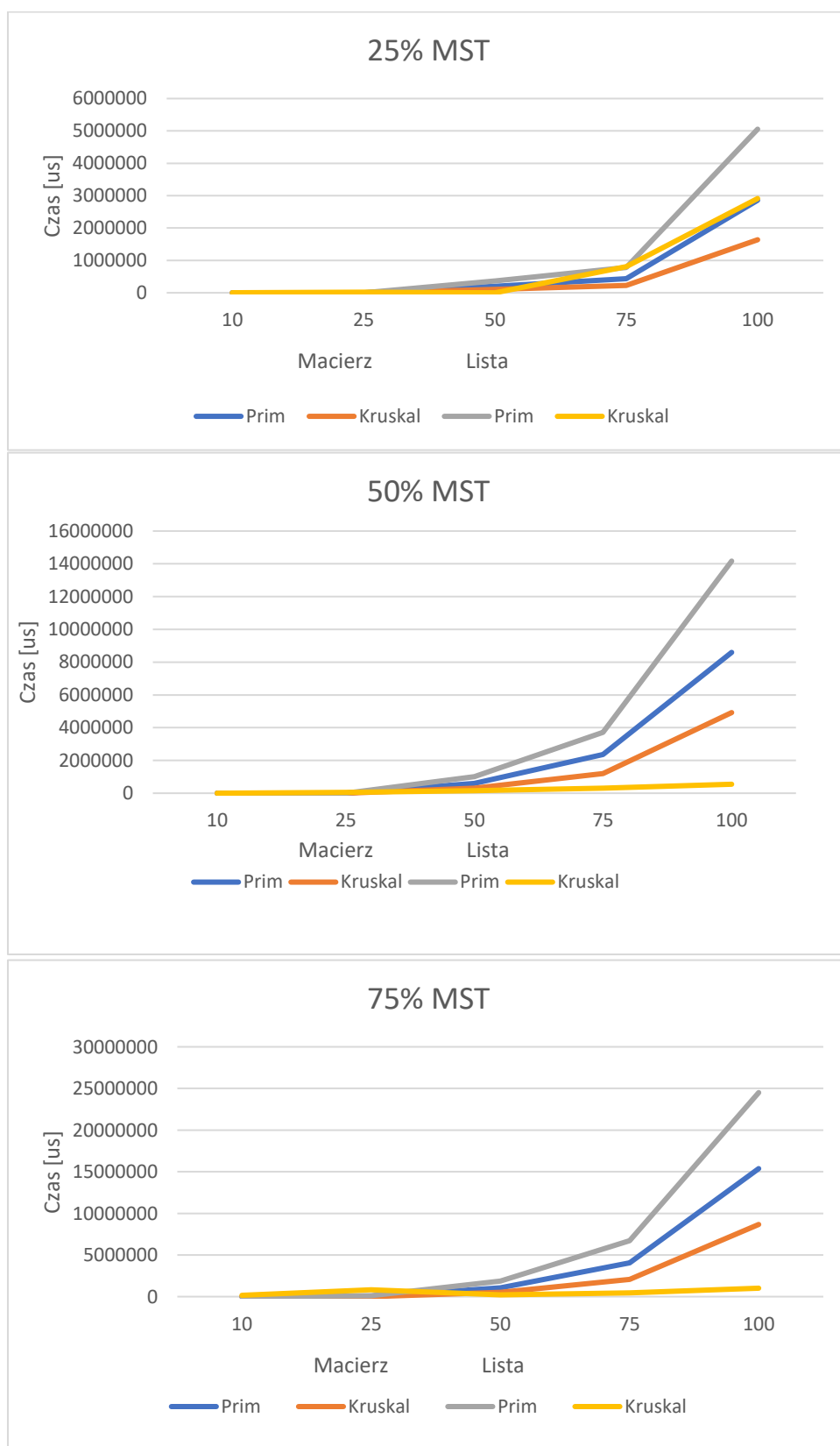
## Najkrotsza ścieżka Lista

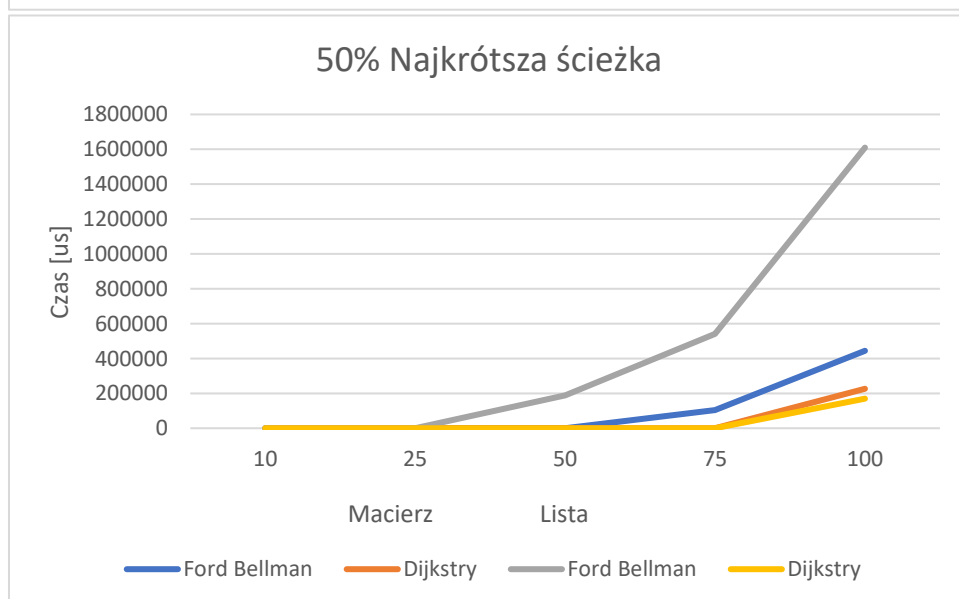
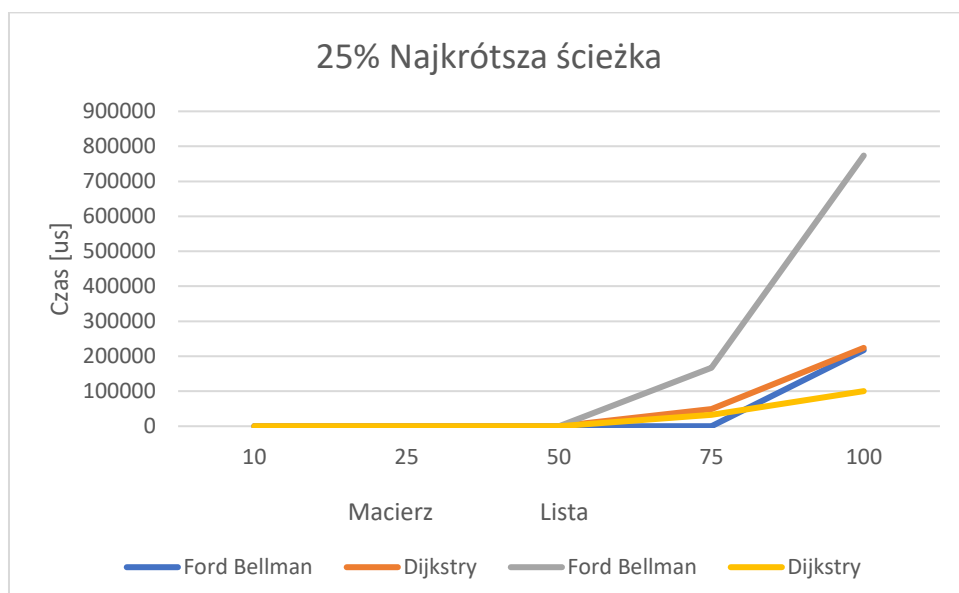
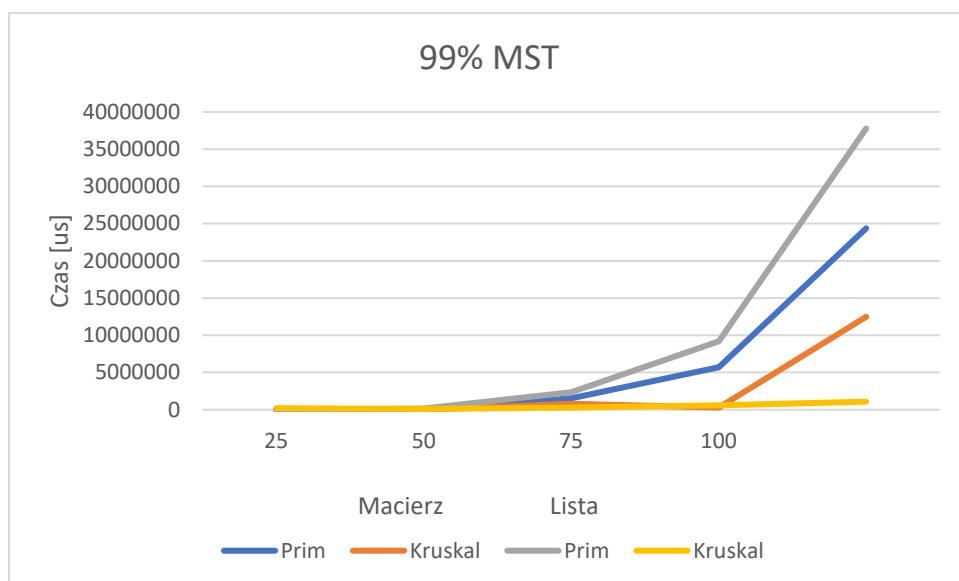


## MST Lista

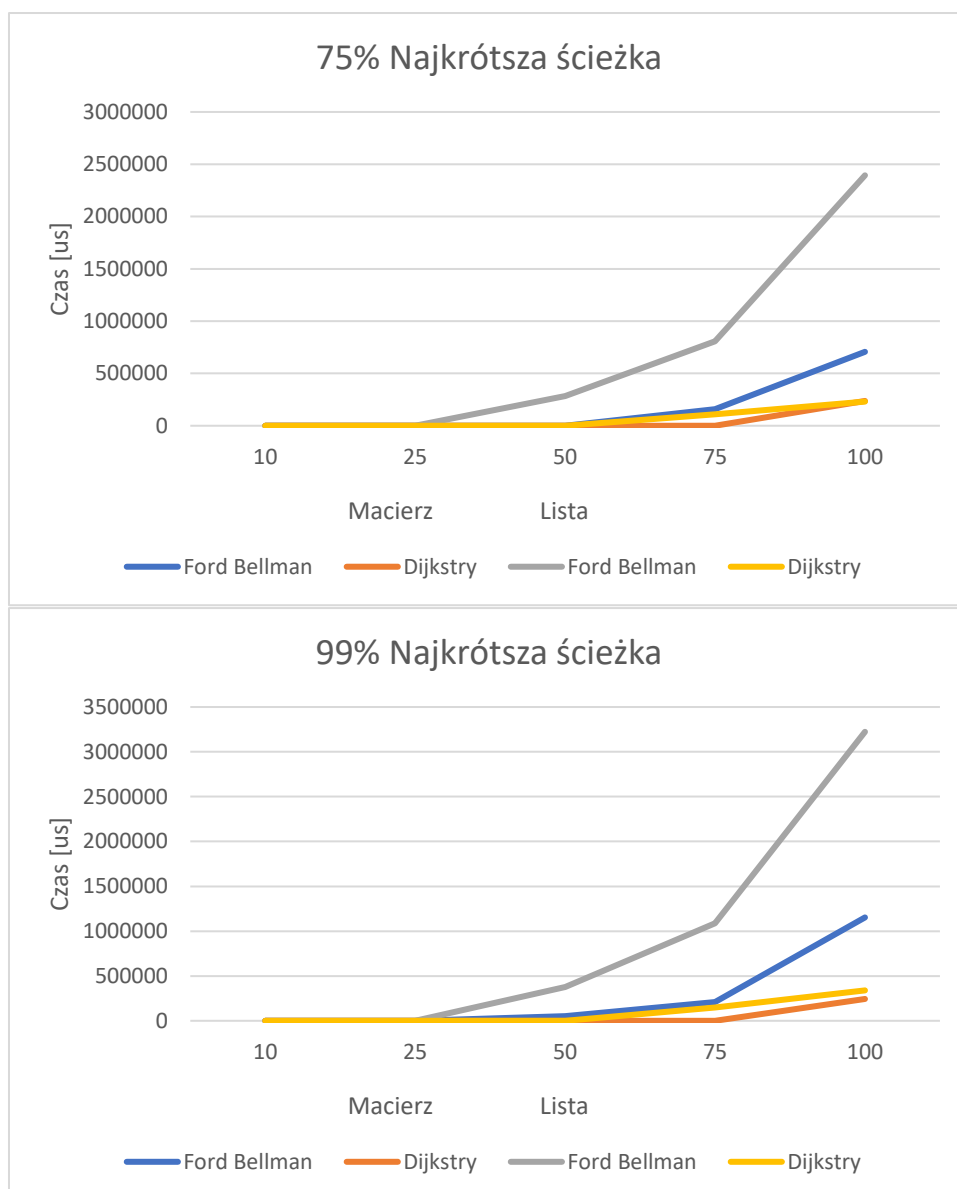


## 2.3 Wykresy dla poszczególnych gęstości









### 3. Wnioski

Większość danych zgadzała się założeniami teoretycznymi. Jedynie rozbieżność można zauważyć przy algorytmie Kruskala dla listy. Najgorszą wydajność miał algorytm Bellmana-Forda dla najkrótszej ścieżki przy wykorzystaniu listy oraz algorytm Prima dla MST również przy liście. Jest to związane prawdopodobnie z koniecznością iteracji po każdym elemencie.