

# Sprawozdanie PAMSI

Zadanie: Przetestować czas dodawania nowych elementów na tablicę asocjacyjną w zależności od ilości elementów. Testy wykonać dla różnej ilości bucketów w tablicy.

Ilość bucketów: 26 + 1 bucket na niezidentyfikowane			
Ilość danych	Czas dodawania [s]	Czas wyszukiwania [s]	Ilość elementów w przeszukiwanym buckecie
10	0,007	0	2
	0,007	0	
	0,006	0	
	0,010	0	
	0,009	0	
	0,017	0	
	0,010	0	
	0,007	0	
	0,008	0	
	0,007	0	
100	0,069	0,001	6
	0,070	0,002	
	0,074	0,002	
	0,066	0,002	
	0,065	0,001	
	0,069	0,003	
	0,063	0,003	
	0,112	0,002	
	0,077	0,003	
	0,075	0,002	
1000	0,458	0,001	56
	0,471	0,001	
	0,444	0,001	
	0,452	0,001	
	0,476	0,003	
	0,481	0,001	
	0,436	0,001	
	0,374	0,001	
	0,405	0,001	
	0,390	0,001	
	0,374	0,001	
10000	3,401	0,001	631
	3,416	0,002	
	3,447	0,001	
	3,463	0,001	
	3,510	0,001	
	3,510	0,001	
	3,463	0,001	
	3,478	0,001	
	3,510	0,001	
	3,463	0,002	
100000	40,623	0,002	13053
	40,716	0,002	
	40,092	0,002	
	40,155	0,002	
	40,467	0,001	
	40,116	0,002	
	40,334	0,002	
	40,360	0,001	
	40,387	0,002	
	40,136	0,002	
1000000	311,515	0,015	26003
	308,615	0,016	
	303,982	0,015	
	306,79	0,015	
	304,23	0,015	
	313,42	0,015	
	307,882	0,014	
	316,384	0,015	
	311,095	0,015	
	303,124	0,015	

Tabela 1

Ilość bucketów: 13 + 1 bucket na niezidentyfikowane			
Ilość danych	Czas dodawania [s]	Czas wyszukiwania [s]	Ilość elementów w przeszukiwanym buckecie
10	0,003	0,001	3
	0,010	0,001	
	0,009	0,001	
	0,009	0,001	
	0,010	0,003	
	0,015	0,001	
	0,006	0,001	
	0,010	0,001	
	0,010	0,003	
	0,007	0,001	
100	0,046	0,003	14
	0,074	0,003	
	0,047	0,002	
	0,077	0,003	
	0,046	0,002	
	0,069	0,001	
	0,112	0,002	
	0,046	0,002	
	0,047	0,002	
	0,065	0,001	
1000	0,343	0,006	118
	0,358	0,002	
	0,358	0,002	
	0,390	0,002	
	0,405	0,002	
	0,343	0,001	
	0,390	0,002	
	0,390	0,002	
	0,359	0,001	
	0,358	0,002	
	0,358	0,002	
10000	3,37	0,002	1314
	3,354	0,003	
	3,322	0,002	
	3,307	0,003	
	3,291	0,002	
	3,354	0,002	
	3,354	0,003	
	3,260	0,002	
	3,354	0,003	
	3,292	0,002	
100000	40,997	0,002	13053
	40,887	0,005	
	40,887	0,002	
	40,294	0,002	
	45,541	0,005	
	40,123	0,016	
	40,731	0,002	
	39,639	0,016	
	41,246	0,002	
	40,116	0,005	
1000000	370,757	0,016	50653
	343,403	0,025	
	364,729	0,027	
	367,614	0,025	
	364,729	0,027	
	362,989	0,017	
	367,003	0,019	
	367,656	0,025	
	366,099	0,027	
	340,174	0,025	

Tabela 2

Ilość bucketów: 2 + 1 bucket na niezidentyfikowane			
Ilość danych	Czas dodawania [s]	Czas wyszukiwania [s]	Ilość elementów w przeszukiwanym buckecie
10	0,015	0,002	6
	0,015	0,002	
	0,015	0,001	
	0,015	0,003	
	0,015	0,002	
	0,015	0,002	
	0,015	0,002	
	0,015	0,002	
	0,015	0,002	
	0,015	0,002	
	0,015	0,003	
100	0,031	0,003	58
	0,031	0,003	
	0,031	0,002	
	0,046	0,003	
	0,031	0,002	
	0,031	0,001	
	0,031	0,002	
	0,031	0,002	
	0,031	0,002	
	0,046	0,001	
1000	0,343	0,003	562
	0,374	0,003	
	0,327	0,002	
	0,358	0,003	
	0,327	0,003	
	0,374	0,003	
	0,343	0,002	
	0,327	0,003	
	0,343	0,003	
	0,374	0,003	
	0,327	0,002	
10000	2,855	0,003	5782
	3,026	0,003	
	3,182	0,002	
	3,213	0,003	
	3,213	0,003	
	3,213	0,003	
	3,073	0,002	
	3,213	0,003	
	3,213	0,003	
	3,276	0,003	
100000	48,016	0,016	57955
	47,734	0,016	
	48,033	0,015	
	46,815	0,015	
	46,860	0,016	
	48,781	0,016	
	47,081	0,015	
	48,563	0,016	
	47,626	0,016	
	48,456	0,015	
1000000	477,345	0,048	224668
	475,239	0,075	
	475,531	0,081	
	477,342	0,075	
	476,234	0,081	
	477,234	0,051	
	474,938	0,057	
	463,664	0,075	
	477,766	0,081	
	476,983	0,075	

Tabela 3

Wartość średnia Tab1		
Ilość danych	Czas dodawania [s]	Czas wyszukiwania [s]
10	0,0088	0,0021
100	0,0740	0,0021
1000	0,4761	0,0023
10000	3,4661	0,0023
100000	40,3386	0,0024
1000000	308,7037	0,0150

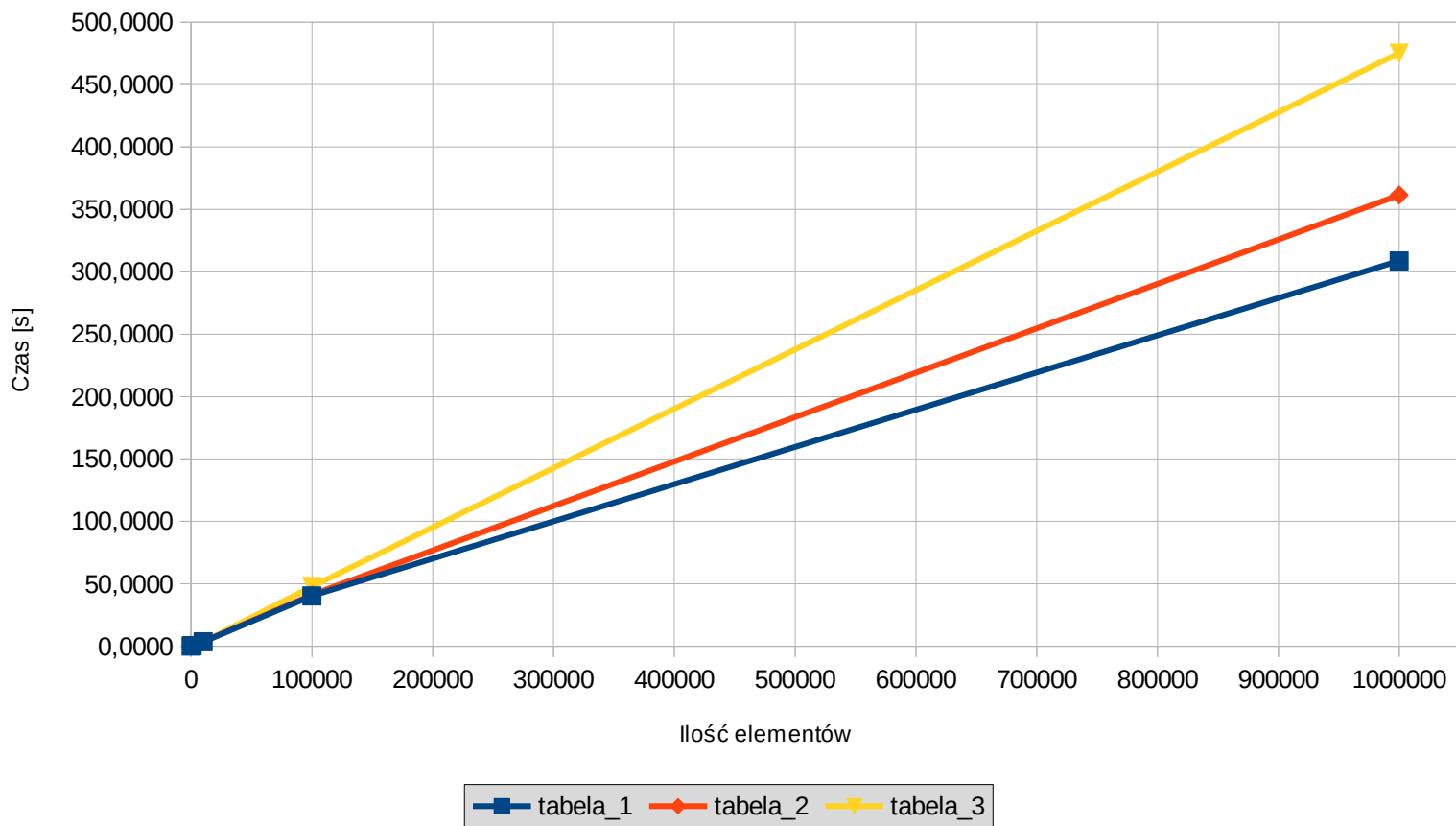
Wartość średnia Tab2		
Ilość danych	Czas dodawania [s]	Czas wyszukiwania [s]
10	0,0089	0,0014
100	0,0629	0,0021
1000	0,4052	0,0024
10000	3,3258	0,0024
100000	41,0461	0,0057
1000000	361,5153	0,0233

Wartość średnia Tab3		
Ilość danych	Czas dodawania [s]	Czas wyszukiwania [s]
10	0,0150	0,0021
100	0,0340	0,0021
1000	0,3817	0,0030
10000	3,1477	0,0028
100000	47,7965	0,0156
1000000	475,2276	0,0699

Zmiana pojemności bucketów		
Tabela 1	Tabela 2	Tabela3
2	3	6
6	14	58
56	118	562
631	1314	5782
6652	13053	57955
26003	50653	224668

## Średnie czasy sortowania

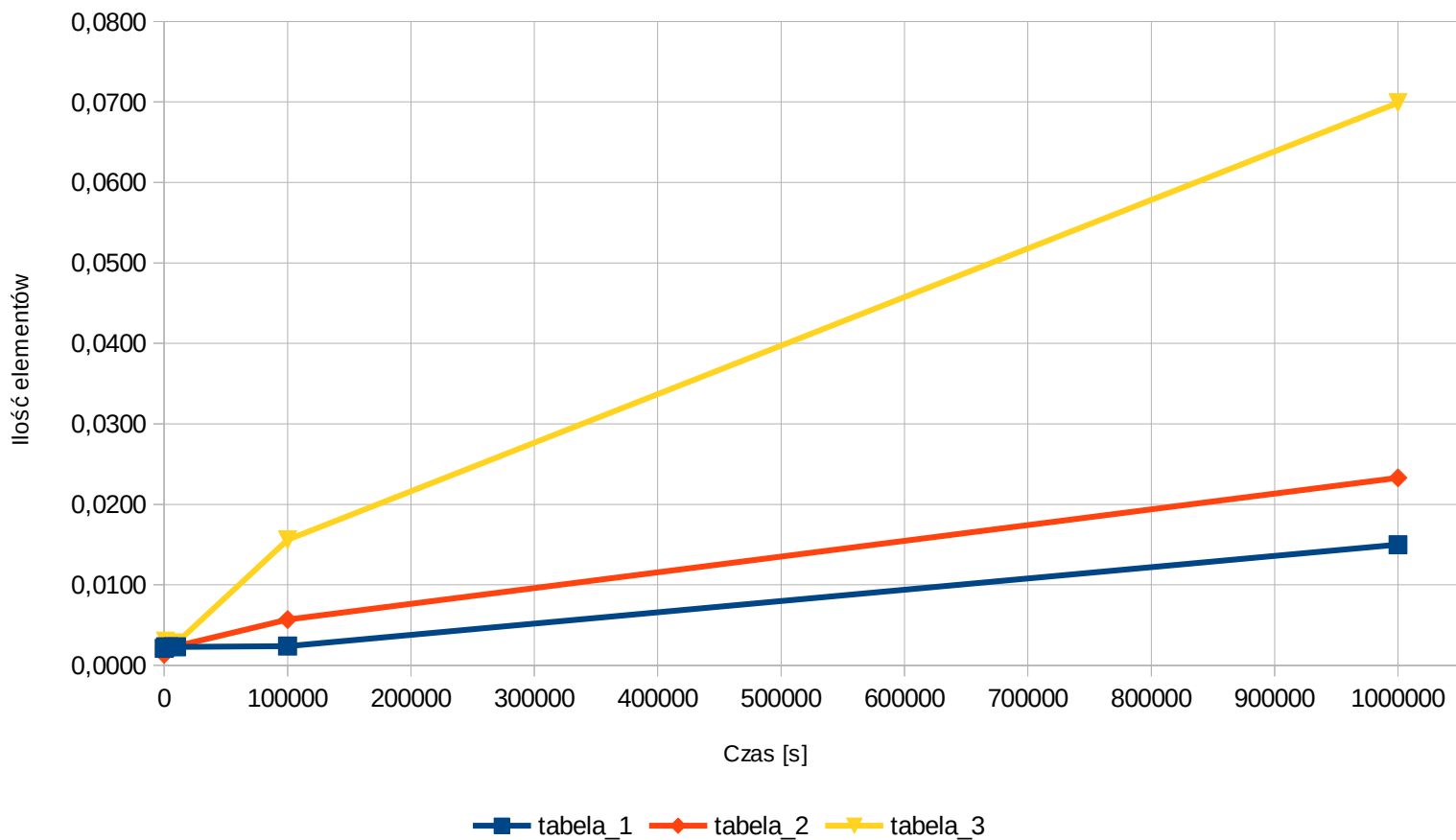
Porównanie średnich czasów dodawania dla tabel 1,2,3



Wykres 1

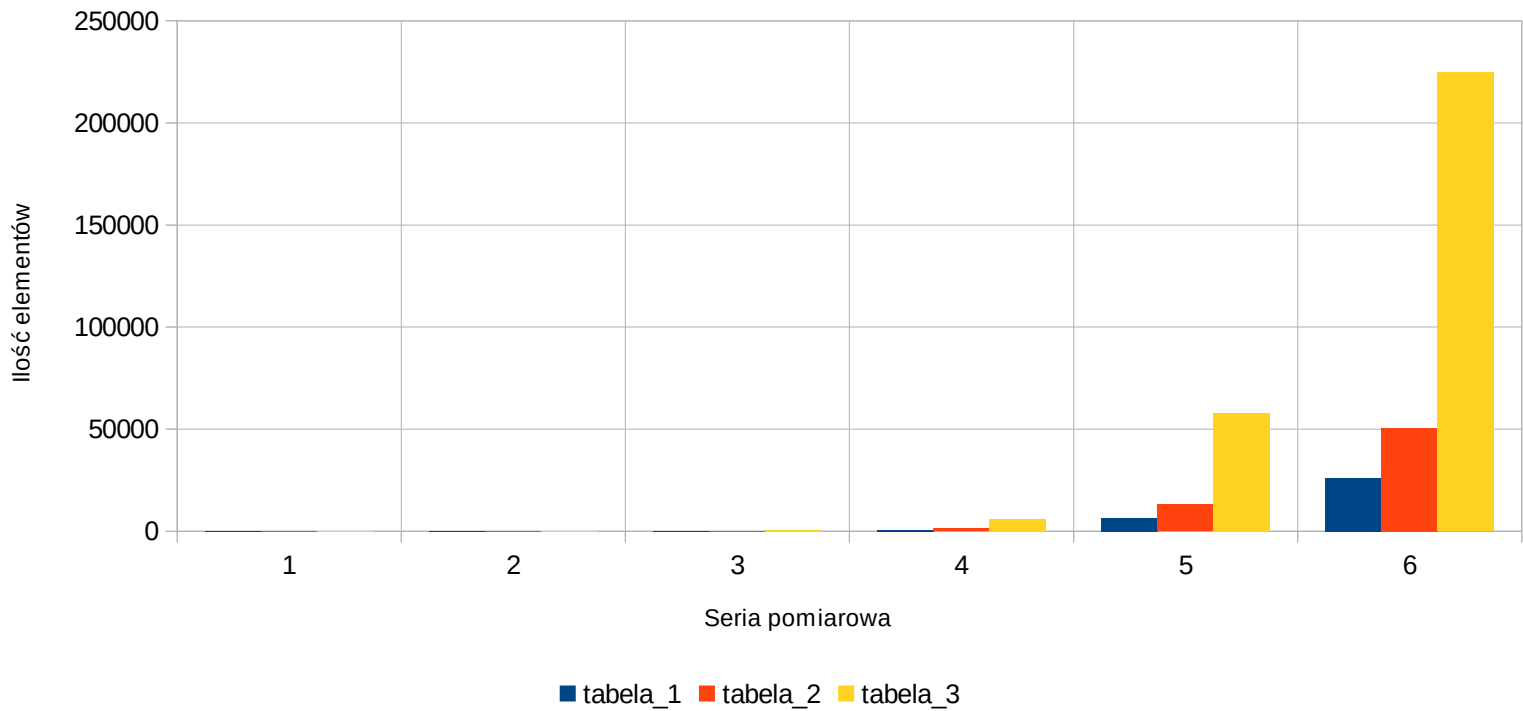
## Średnie czasy wyszukiwania

Porównanie średnich czasów wyszukiwania dla tabel 1,2,3



Wykres 2

## Zmiana pojemności przeszukiwanych bucketów



Wykres 3

### Wnioski:

Zgodnie z wynikami odczytanymi z Wykres 2, opisującego zmiany czasu dodawania elementów w zależności od ilości elementów do dodania, charakterystyka złożoności obliczeniowej wydaje się być liniowa  $O(n)$ . Jest to w miarę logiczne, ponieważ każdy element dodawany jest w następującej sekwencji: wczytaj, odszyfruj hasz elementu, umieść w odpowiednim polu tablicy **na początku listy**. Wstawianie na początek listy unika zbędnego „przechodzenia” po wskaźnikach listy i dla każdego elementu wymaga tyle samo czasu. Zastanawiający jest jednak fakt, czemu w takim razie linie dla Tabela 1, Tabela 2, Tabela 3, nie pokrywają się. Ten fakt sugeruje nam, że wraz z ilością elementów aktualnie zapisanych w bucketcie, wzrasta czas dodawania na początek listy. Może to być też spowodowane obciążeniem procesora podczas obliczeń przez inny zewnętrzny niezwiązany z nimi program, lecz jest to mało prawdopodobne.

Wyniki odczytane z Wykres 2, opisującego zmiany czasu przeszukiwania listy w zależności od ilości dodanych elementów, są zbyt mało miarodajne aby móc na ich podstawie oszacować złożoność obliczeniową algorytmu. Jedynie dwa ostatnie uśrednione wyniki uważam za prawidłowe. Wcześniejsze są zbyt małe aby odczytać je z odpowiednią dokładnością. Z tego wykresu możemy odczytać jednak jedną prawidłowość. Wraz ze wzrostem ilości elementów w bucketcie zwiększa się czas przeszukiwania go w celu znalezienia elementu umiejscowionego na końcu bucketa. Spowodowane jest to koniecznością przeszukiwania w następującej sekwencji: (os początku listy) sprawdź czy wskazany element jest poszukiwanym, jeżeli nie → sprawdź następny element. Taka sekwencja wiąże się z koniecznością przechodzenia po wskaźnikach, co zwiększa czas przeszukiwania.

Wyniki z Wykres 3 są logicznie poprawne. Wraz ze zmniejszeniem ilości bucketów, będą one przechowywały większą ilość danych.

Na bazie otrzymanych wyników możemy wywnioskować, że uzyskanie najlepszej złożoności obliczeniowej  $O(1)$  może zostać osiągnięte przez zwiększenie ilości bucketów, czyli przyporządkowania każdemu dodanemu elementowi unikalnego hashu. Wymagało by to jednak unikalnego miejsca w tablicy asocjacyjnej dla każdego elementu.