

# [AAL] Tablica mieszająca dokumentacja

## 1. Autorzy:

-Rafał Lewanczyk

-<https://github.com/rafallewanczyk/AAL>

## 2. Temat:

W14, W21, W31

Analiza oraz implementacja tablicy mieszającej spełniającej następujące założenia:

-W przypadku kolizji obliczamy nową lokalizację

-Testy przeprowadzić dla: listy słów języka polskiego wygenerowanych z zadanych tekstów

-Zastosować jedną funkcję mieszającą; dodatkowo przeprowadzić analizę dla enumeracji tablicy (wydobycia wszystkich elementów).

## 3. Założenia:

-Program został napisany w języku Python

-Dane testowe służące do przeprowadzenia analizy składają się z listy różnych polskich słów.

## 4. Opis funkcjonalności:

-klasa *HashMap* zawiera implementację tablicy mieszającej o adresowaniu otwartym. Udostępnia metody *add*, *delete*, *find*, *getall*

-klasa *Generator* zawiera implementację generatora. Generator czyta dane z pliku dostarczonego przez użytkownika lub wygenerowanego przez *wget* oraz usuwa z niego wszystkie słowa które zawierają znaki spoza polskiego alfabetu lub są zawarte w nawiasach <>. Zwraca listę słów.

## 5. Uruchomienie programu:

Opisane w readme.md

## 6. Rozwiązanie:

-Zamiana klucza na wartość całkowitoliczbową została zrealizowana poprzez jej reprezentację w systemie pozycyjnym o podstawie 380 (kod ASCII ostatniej polskiej litery). Aby uniknąć zbyt dużych wartości klucza już podczas zamiany obliczamy na bieżąco jego moduł K. W tym celu został zastosowany algorytm szybkiego potęgowania modulo.

-Tablica używa adresowania otwartego o funkcji haszującej danej wzorem

$$h(k, i) = (h(k) + \frac{1}{2}i + \frac{1}{2}i^2) \bmod K$$
, gdzie K jest rozmiarem tablicy, i jest liczbą kolizji

-Jeśli komórka tablicy haszującej nie posiadała nigdy żadnej wartości przechowuje w niej wartość '1'. Po dodaniu klucza, komórka przechowuje klucz, natomiast po usunięciu klucza przechowuje wartość '2'. Wartości '1' i '2' są potrzebne aby móc wyszukiwać w tablicy po wystąpieniu kolizji oraz usunięcia.

-Jeśli funkcja haszująca próbująca wstawić element do tablicy wykona pętlę, tablica odrzuci słowo oraz wyświetli komunikat "nie można dodać klucza {klucz}"

-Tablica nie pozwala na dodanie duplikatu klucza

## 7. Analiza:

Szybkość działania tablicy mieszającej zależy od jej współczynnika wypełnienia

$$\alpha = \frac{\text{liczba elementów w tablicy}}{\text{wielkość tablicy}}. \text{ Analizę złożoności przeprowadziłem dla dwóch}$$

przypadków

$$\alpha \approx 0.5 \text{ oraz } \alpha \approx 1$$

$$q(n) = \frac{t(n)T(n_{\text{medianaa}})}{t(n_{\text{mediana}})T(n)}$$

Wstawianie elementu dla $\alpha \approx 0.5$ . Szacuje $O(1)$		
n	t(n) [s*e-5]	q(n)
1000	4.60	0.92
5000	4.68	0.93
10000	4.58	0.91
50000	5.14	1.02
100000	5.03	1.00
500000	5.01	1.00
1000000	5.01	1.00
2000000	5.78	1.15
3000000	7.27	1.45

Wstawianie elementu dla $\alpha \approx 1$ . Szacuje $O(1)$		
n	t(n)[s*e-5]	q(n)
1000	8.70	1.04
5000	7.00	0.83
10000	7.22	0.86
50000	8.24	0.98
100000	8.38	1.00
500000	8.26	0.98
1000000	8.16	0.97
2000000	6.72	0.80
3000000	6.58	0.78

Enumeracja tablicy dla $\alpha \approx 0.5$ . Szacuje $O(n)$		
n	t(n) [s]	q(n)
1000	0.00	1.26
5000	0.01	0.92
10000	0.02	0.97
50000	0.10	0.88
100000	0.24	1.00
500000	1.34	1.13
1000000	2.40	1.01
2000000	4.73	0.99
3000000	7.02	0.98

Enumeracja tablicy dla $\alpha \approx 1$ . Szacuje $O(n)$		
n	t(n) [s]	q(n)
1000	0.00	1.42
5000	0.01	0.85
10000	0.02	0.75
50000	0.10	0.97
100000	0.21	1.00
500000	1.16	1.01
1000000	2.15	1.01
2000000	4.17	0.98
3000000	6.33	1.00

N wyszukiń dla $\alpha \approx 0.5$ . Szacuje $O(n)$		
n	t(n) [s]	q(n)
1000	0.04	0.91
5000	0.23	0.93
10000	0.46	0.93
50000	2.52	1.02
100000	4.95	1.00
500000	24.82	1.00
1000000	49.08	0.99
2000000	114.49	1.16
3000000	145.70	0.98

N wyszukiń dla $\alpha \approx 1$ . Szacuje $O(n)$		
n	t(n) [s]	q(n)
1000	0.05	0.77
5000	0.25	0.86
10000	0.50	0.85
50000	2.85	0.96
100000	5.94	1.00
500000	30.39	1.02
1000000	61.25	1.03
2000000	121.55	1.02
3000000	184.37	1.03

## 8. Wnioski:

Pomimo że w przypadku wykonania n wyszukiń oraz enumeracji tablicy złożoność jest liniowa czasy tych operacji znacznie się różnią. Jest to spowodowane koniecznością obliczania wartości indeksu elementu w tablicy przy wyszukiwaniu co zwiększa czas wykonywania, ale nie zmienia klasy złożoności.

Enumeracja tablicy nie zależy od współczynnika zapęnlennia  $\alpha$ .

Jeżeli  $\alpha < 1$  to oczekiwana liczba porównań podczas wstawiania lub wyszukiwania jest mniejsza niż  $\frac{1}{1-\alpha}$ . Wynika to ze wzoru  $\frac{1}{1-\alpha} = 1 + \alpha + \alpha^2 + \alpha^3 + \dots$ . Pierwsze porównanie musimy wkonać za każdym razem, drugie porównujemy z prawdopodobieństwem  $\alpha$ , trzecie z prawdopodobieństwem  $\alpha^2$  itd.