

# [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

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

$\alpha \approx 0.5$  oraz  $\alpha \approx 1$

Wstawianie elementu dla  $\alpha \approx 0.5$ . Szacuje  $O(1)$

n	t(n)	q(n)
1000	4.683828e-05	0.71
5000	6.114063e-05	0.93
10000	6.946511e-05	1.06
50000	6.578375e-05	1.00
100000	7.263148e-05	1.10
500000	6.377113e-05	0.97
1000000	6.625939e-05	1.01

Wstawianie elementu dla  $\alpha \approx 1$ . Szacuje  $O(1)$

n	t(n)	q(n)
1000	5.485463e-05	0.64
5000	9.475402e-05	1.10
10000	7.773022e-05	0.90
50000	8.617057e-05	1.00
100000	8.890803e-05	1.03
500000	9.222760e-05	1.07
1000000	9.936787e-05	1.15

Enumeracja tablicy dla  $\alpha \approx 0.5$ . Szacuje  $O(n)$

n	t(n)	q(n)
1000	0.039893	0.587234837
5000	0.419360	1.234616605
10000	0.596407	0.87792587
50000	3.396682	1
100000	6.425576	0.945860696
500000	32.020983	0.942713595
1000000	66.484603	0.978669817

Enumeracja tablicy dla  $\alpha \approx 1$ . Szacuje  $O(n)$

n	t(n)	q(n)
1000	0.043881	0.533502669
5000	0.309173	0.751781503
10000	0.743040	0.90338375
50000	4.112538	1
100000	8.474689	1.030347805
500000	41.940460	1.019819391
1000000	93.552564	1.137406682

## 8. Wnioski:

Jeżeli  $\alpha < 1$  to oczekiwana liczba porównań podczas wstawiania lub wyszukiwania jest mniejsza niż  $\frac{1}{1-\alpha}$