

Sprawozdanie z laboratorium 2 – struktury trie oraz drzewo sufiksów

1. Sprawdzenie algorytmów przeprowadzono na następujących danych
 - a. txt1 bbbd
 - b. txt2 aabbabd
 - c. txt3 ababcd
 - d. txt4 abcbccd
 - e. txt5 1997_714_head.txt (fragment ustawy)
2. Na końcu każdego tekstu został dodany znak unikalny: \$
3. Została zaimplementowana struktura trie
4. Została zaimplementowana struktura drzewa sufiksów (bez użycia linków)
5. Dla danych wejściowych z punktu 1 przeprowadzono sprawdzenie algorytmów. Każdy możliwy sufiks został wyszukany poprawnie. Zaimplementowana została także prymitywna funkcja print służąca do wizualnego sprawdzenia drzewa dla prostych przykładów

```
Test txt1, trie structure is correct: True
Test txt2, trie structure is correct: True
Test txt3, trie structure is correct: True
Test txt4, trie structure is correct: True
Test txt5, trie structure is correct: True

Test txt1, suffix tree structure is correct: True
Test txt2, suffix tree structure is correct: True
Test txt3, suffix tree structure is correct: True
Test txt4, suffix tree structure is correct: True
Test txt5, suffix tree structure is correct: True
```

6. Został zmierzony czas tworzenia reprezentacji badanych tekstów

structure	txt1	txt2	txt3	txt4	txt5
trie	0	0	0	0	9014379
suffix tree	0	0	0	0	124204

Czas tworzenia w ms

7. Został zmierzony czas wyszukiwania wszystkich możliwych prefiksów tekstu w strukturze

structure	txt5
trie	719791
suffix tree	75332

Czas wyszukiwania w ms

8. Zgodnie z oczekiwaniami struktura drzewa sufiksowego okazała się szybsza zarówno w inicjalizacji jak i w działaniu. Będzie ona miała także lepszą złożoność pamięciową.