

Rozwiązanie każdego z zadań polega na:

- 1) Zaimplementowaniu struktury drzewa, która umożliwia efektywne wykonywanie operacji**
- 2) Przy implementacji należy wybrać drzewo AVL lub nie zrównoważone drzewo BST (połowa punktów do zdobycia)**

UWAGA:

- zakładamy, że w prawym poddrzewie znajdują się elementy większe, a w lewym mniejsze
- przy implementowaniu usuwania węzła, który ma dwóch synów, należy w jego miejsce wstawić najmniejszy element w jego prawym poddrzewie
- w drzewie nie dopuszcza się duplikatów

Problem 3 - „Słownik”

Napisać program do przechowywania wyrazów w słowniku. Program ma umożliwiać szybkie wykonywanie następujących operacji:

- a) stawienie nowego słowa (co najwyżej 30 małych liter angielskich)
- b) sunięcie danego słowa
- c) yszukanie w słowniku zadanego słowa
- d) o iczenie liczby słów o danym prefiksie
- e) yświetlenie struktury drzewa wraz z elementami
- f) ykonanie skryptu poleceń:
 - 1) W x – wstaw x
 - 2) x – usuń x
 - 3) x – szukaj x (odpowiedź: TAK/NIE)
 - 4) x – wypisać, ile słów zaczyna się prefiksem x

Przykład

Plik wejściowy:

```
5 //liczba poleceń
W kot //wstaw słowo „kot”
W kosa //wstaw słowo „kosa”
S kos //szukaj słowa „kos”
W kowal //wstaw słowo „kowal”
U kot //usuń słowo „kot”
L ko //ile słów o prefiksie „ko”
```

Plik wyjściowy:

```
NIE //słowa „kos” nie ma w drzewie
2 //2 słowa zaczynają się na „ko”
```