

**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 1 - „Liczby rzeczywiste”**

Napisać program do zarządzania zbiorem liczb rzeczywistych nieujemnych (maksymalnie 8 cyfr dziesiętnych przed i po przecinku). Program ma umożliwiać szybkie wykonywanie następujących operacji:

- a) wstawienie do zbioru nowej liczby
- b) usunięcie ze zbioru zadanej liczby
- c) wyszukanie w zbiorze danej liczby
- d) znalezienie liczby liczb, których część całkowita jest równa zadanej przez użytkownika liczbie
- e) wyświetlenie struktury drzewa wraz z elementami
- f) wykonanie skryptu poleceń:
  - 1) W x – wstaw x
  - 2) U x – usuń x
  - 3) S x – szukaj x (odpowiedź: TAK/NIE)
  - 4) L x – wypisać, ile liczb posiada część całkowitą równą x

### **Przykład**

Plik wejściowy:

```
5 //liczba poleceń
W 1,6 //wstaw liczbę 1,5
W 1,8 //wstaw liczbę 1,8
S 2,5 //szukaj liczby 2,5
W 2,1 //wstaw liczbę 2,1
U 1,6 //usuń liczbę 1,6
L 2 //ile liczb posiada część całkowitą 2
```

Plik wyjściowy:

```
NIE //liczba 2,5 nie występuje w drzewie
1 //1 liczba posiada część całkowitą 2
```

## Problem 2 - „Liczby całkowite”

Napisać program do zarządzania zbiorem liczb naturalnych (z przedziału  $(0, 10^{18})$ ). Program ma umożliwiać szybkie wykonywanie następujących operacji:

- a) wstawienie do zbioru nowej liczby
- b) usunięcie ze zbioru zadanej liczby
- c) wyszukanie w zbiorze danej liczby
- d) znalezienie liczby liczb należących do przedziału  $\langle a, b \rangle$
- e) wyświetlenie struktury drzewa wraz z elementami
- f) wykonanie skryptu poleceń:
  - 1) W x – wstaw x
  - 2) U x – usuń x
  - 3) S x – szukaj x (odpowiedź: TAK/NIE)
  - 4) L x y – wypisać, ile liczb znajduje się w przedziale  $\langle x, y \rangle$

### Przykład

Plik wejściowy:

```
5 //liczba poleceń
W 2 //wstaw liczbę 2
W 3 //wstaw liczbę 3
S 3 //szukaj liczby 3
W 5 //wstaw liczbę 5
U 3 //usuń liczbę 3
L 2 4 //ile liczb z przedziału <2, 4>
```

Plik wyjściowy:

```
TAK //liczba 3 występuje w drzewie
1 //1 liczba należy do przedziału <2, 4>
```

## Problem 3 - „Słownik”

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

- a) wstawienie nowego słowa (co najwyżej 30 małych liter angielskich)
- b) usunięcie danego słowa
- c) wyszukanie w słowniku zadanego słowa
- d) obliczenie liczby słów o danym prefiksie
- e) wyświetlenie struktury drzewa wraz z elementami
- f) wykonanie skryptu poleceń:
  - 1) W x – wstaw x
  - 2) U x – usuń x
  - 3) S x – szukaj x (odpowiedź: TAK/NIE)
  - 4) L 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”
```