

Яндекс. Тренировки по алгоритмам 2.0, занятие 8 (B)

A. Бинарное дерево (вставка, поиск, обход)

Ограничение времени	1 секунда
Ограничение памяти	64Mb
Ввод	стандартный ввод или input.txt
Вывод	стандартный вывод или output.txt

Напишите программу, которая будет реализовывать действия в бинарном дереве поиска «вставить» и «найти» (по значению). Программа должна обрабатывать запросы трёх видов:
ADD n — если указанного числа еще нет в дереве, вставлять его и выводить слово «DONE», если уже есть — оставлять дерево как было и выводить слово «ALREADY».
SEARCH — следует выводить слово «YES» (если значение найдено в дереве) или слово «NO» (если не найдено). Дерево при этом не меняется.
PRINTTREE — выводить все дерево, обязательно используя алгоритм, указанный в формате вывода результатов.

Формат ввода

В каждой строке входных данных записан один из запросов ADD n или SEARCH n или PRINTTREE. Гарантируется, что запросы PRINTTREE будут вызываться только в моменты, когда дерево не пустое. Общее количество запросов не превышает 1000, из них не более 20 запросов PRINTTREE.

Формат вывода

Для каждого запроса выводите ответ на него. Для запросов ADD и SEARCH — соответствующее слово в отдельной строке. На запрос PRINTTREE надо выводить дерево, обязательно согласно такому алгоритму:
1) Распечатать левое поддереву
2) Вывести количество точек, равное глубине узла
3) Вывести значение ключа
4) Распечатать правое поддереву

Пример

Ввод <input type="text"/>	Вывод <input type="text"/>
ADD 2	DONE
ADD 3	DONE
ADD 2	ALREADY
SEARCH 2	YES
ADD 5	DONE
PRINTTREE	2
SEARCH 7	.3
	..5
	NO

Язык Python 3.12.1

Набрать здесь

Отправить файл

```
1 class Node:
2     """
3     Класс Узел
4     """
5     def __init__(self, val=None, left=None, right=None):
6         """
7         Конструктор узла
8         :param val: значение
9         :param left: левый потомок
10        :param right: правый потомок
11        """
12        self.val = val
13        self.left = left
14        self.right = right
15
16
17 class BST:
18     """
19     Класс - Бинарное Дерево Поиска
20     """
21     def __init__(self):
22         """
23         Конструктор BST. Создается пустой корень.
24         """
25        self.root = None
26
27    def search(self, num):
28        """
29        Функция поиска элемента в BST
30        :return: bool найден или нет элемент
31        """
32        def _search(node, num):
33            """
34            Инкапсулированная рекурсивная функция поиска элемента в BST
35            :param node: текущий узел
36            :param num: искомое значение
37            :return: bool
38            """
```

Отправить

Предыдущая

Следующая