Федеральное государственное автономное образовательное учреждение высшего образования «Национальный исследовательский университет ИТМО»

Факультет инфокоммуникационных технологий

**ОТЧЕТ**

по лабораторной работе №2. «Двоичные деревья поиска» по дисциплине «Алгоритмы и структуры данных»

Выполнил студент 1 курса, группы К3140

Байков Иван

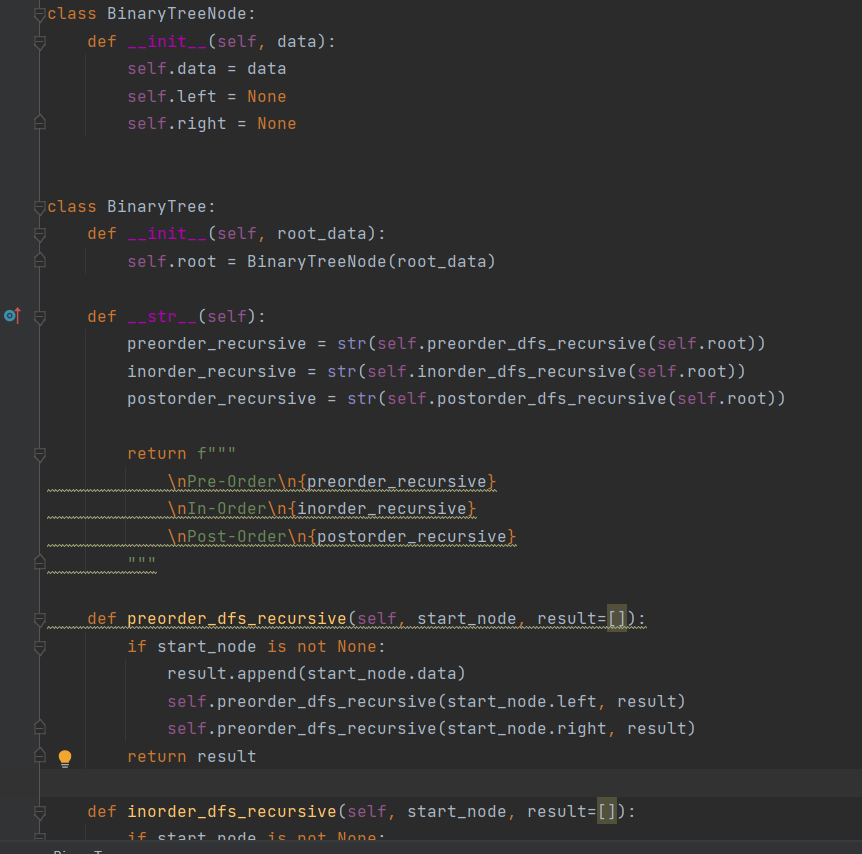
Преподаватель: Харьковская Татьяна Александровна

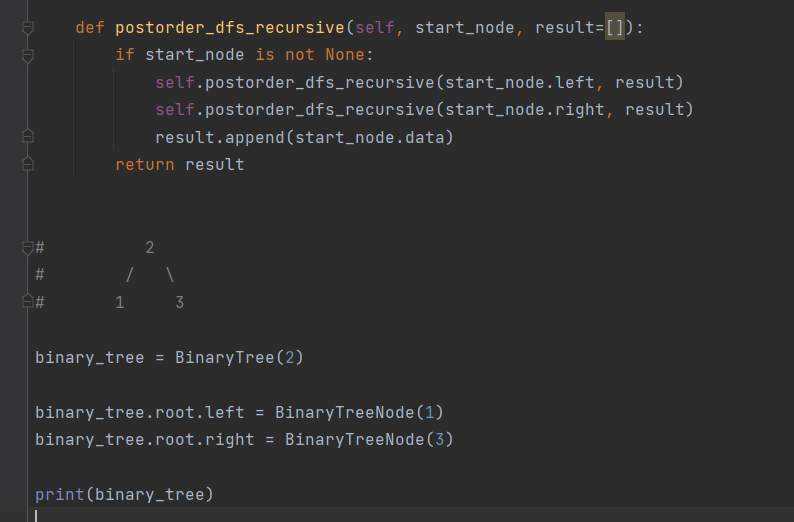
05 Июня, 2022 года, г. Санкт-Петербург

**Задание 1**

**Описание**

В этой задаче вы реализуете три основных способа обхода двоичного дерева «в глубину»: центрированный (in-order), прямой (pre-order) и обратный (post-order).

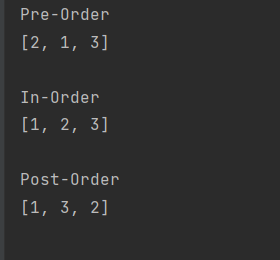
**Исходный код**



**Решение**

Мы обходим дерево по определенным порядкам, стартовой точкой (entry point) всегда является корень, но он не обязан быть первым в списке обхода. Порядок обходов на рисунке:

**Тесты**



Взял простое дерево [2,1,3], чтобы легче показать на примере.

**Вывод**

Было полезно попрактиковаться в реализации алгоритмов обхода, чтобы лучше понимать такую полезную структуру данных и как с ней взаимодействовать.

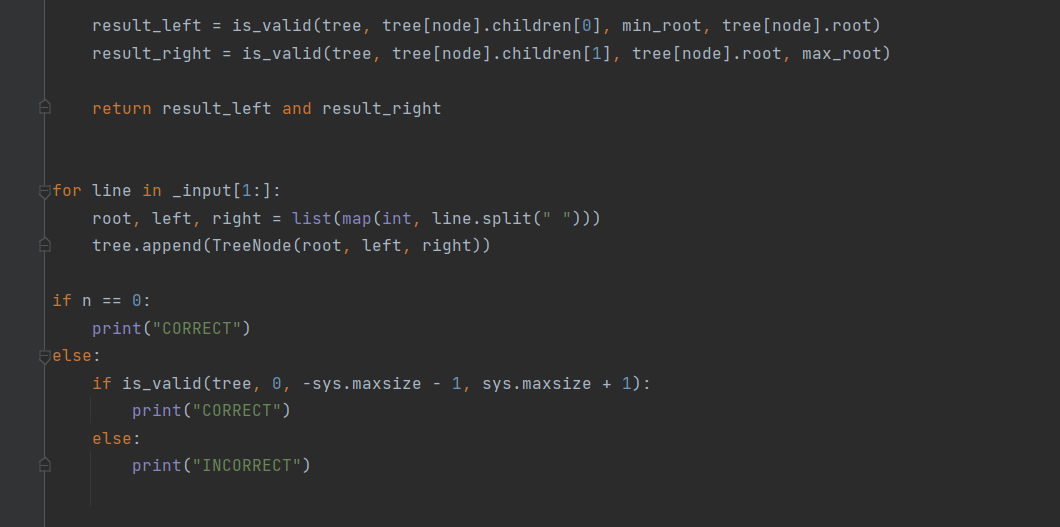
**Задание 6-7**

**Описание**

В данной задаче необходимо проверить, правильно ли реализована структура BST. В 7 задаче усложнение – деревья могут содержать равные ключи.

**Исходный код**





**Решение**

При выполнении In-order DFS каждый текущий узел во время обхода можно добавлять в массив, тогда в конце обхода можно наглядно убедиться, является ли данное дерево двоичным деревом поиском. Суть в том, что при таком обходе, каждый следующий элемент массива будет больше или равен(7) предыдущему, поэтому простым циклом можно это проверить.

**Тесты**

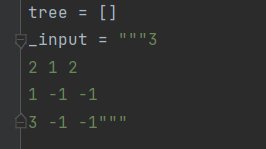


Рисунок 12

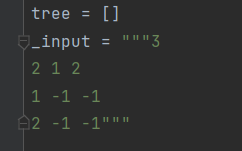


Рисунок 14

Были проверены два теста из условия, один на CORRECT, другой на INCORRECT.

**Вывод**

Не самая сложная задача, если решать через рекурсивный подход. Помогает детальнее изучить BST и узнать некоторые свойства. Асимптотика алгоритма – O(n), где n – количество узлов.

**Задание 11-15**

**Описание**

**Исходный код**

**Решение**

**Тесты**

**Вывод**