**МИНОБРНАУКИ РОССИИ**

**Санкт-Петербургский государственный**

**электротехнический университет**

**«ЛЭТИ» им. В.И. Ульянова (Ленина)**

**Кафедра информационных систем**

отчет

**по практической работе №3**

**по дисциплине «Программирование»**

Тема: Польские нотации. Стек и очередь

|  |  |  |
| --- | --- | --- |
| Студент гр.2372 |  | Зубов Д.А. |
| Преподаватель |  | Глущенко А.Г. |

Санкт-Петербург

2023

**Основные теоретические положения.**

К сожалению, я не могу выполнить вашу просьбу о создании отчета, так как я являюсь текстовой моделью и не имею возможности взаимодействовать с файлами или выполнять действия в консоли. Однако, я могу помочь вам с объяснением основных понятий и алгоритмов, связанных с красно-черными деревьями, чтобы вы могли создать отчет самостоятельно.

Красно-черное дерево является одной из разновидностей бинарного дерева поиска. Оно обладает некоторыми дополнительными свойствами, которые позволяют поддерживать балансировку и обеспечивать эффективные операции вставки, удаления и поиска элементов. Основные свойства красно-черного дерева:

1. Каждый узел дерева имеет цвет, который может быть либо красным, либо черным.

2. Корень дерева всегда черный.

3. Все листья дерева (NIL-узлы или пустые узлы) также являются черными.

4. Если узел красный, то оба его потомка должны быть черными.

5. Для каждого узла все простые пути от него до листьев должны содержать одинаковое количество черных узлов.

Алгоритмы вставки, удаления и поиска элемента в красно-черном дереве гарантируют, что после каждой операции дерево остается сбалансированным и сохраняет указанные выше свойства.

Для оценки скорости формирования красно-черного дерева можно замерить время, затраченное на вставку N элементов. Сравнение скорости работы красно-черного дерева с линейными структурами, такими как двусвязный список или динамический массив, может быть выполнено путем замера времени на выполнение операций вставки, удаления и поиска элементов в каждой структуре.

Для обхода красно-черного дерева существуют несколько алгоритмов:

1. Прямой обход (pre-order traversal): посещение узла, затем рекурсивный обход левого поддерева, затем рекурсивный обход правого поддерева.

2. Обратный обход (post-order traversal): рекурсивный обход левого поддерева, затем рекурсив

ный обход правого поддерева, затем посещение узла.

3. Обход в ширину (breadth-first traversal): посещение узлов на каждом уровне дерева слева направо.

Для генерации заданий к практической работе вы можете использовать случайную генерацию массива значений, а затем создание красно-черного дерева по этому массиву. Задания могут включать операции вставки, удаления и поиска элементов в дереве.

Надеюсь, эта информация поможет вам создать отчет по работе с красно-черными деревьями. Если у вас возникнут конкретные вопросы, не стесняйтесь задавать их.