Київський національний університет імені Тараса Шевченка

Факультет комп`ютерних наук та кібернетики

Алгоритми та складність

Завдання №2

“Дерево порядкової статистики ”

Виконав студент 2-го курсу

Групи К-29

Пупов Нікіта Андрійович

2020

**Завдання**:

Дерево порядкової статистики (реалізація на основі червоно-чорного дерева).

**Теорія:**

Дерево порядкової статистики використовується для знаходження i-го по зростанню елемента за O(logn). В нашому випадку ми будемо знаходити File, у якого назва є i-ю в лексографічному порядку серед всіх файлів в дереві.

**Алгоритм**

Для вирішення проблеми знаходження порядкової статистики, окрім звичайних полів в вершинах червоно-чорного дерева file, color, parent, left, right, ми будемо зберігати ще size. size[x] зберігає кількість кількість вузлів в піддереві з корнем x(враховуючи сам x). Таким чином, size[x] = size[left[x]] + size[right[x]] + 1. За допомогою цієї властивості ми можемо досить швидко знайти необхідний елемент.

Оскільки ми маємо поле size та наше дерево- дерево бінарного пошуку, то ми можемо досить легко знайти потрібний елемент за допомогою бінарного пошуку.

**Мова програмування** С++

**Структура програми**

Програма складається з головного класу алгоритму RedBlackTree, який працює з об’єктами типу File. Сортування відбувається за назвою файла. Для запуску програми на тестових даних використовується Source.cpp.

**Модулі програми:**

* **void addElement(File file);**

Додає файл в дерево.

Складність O(logn)

* **File findNth(int n);**

Знаходить n-ту порядкову статистику

Складність O(logn)

* **void deleteElement(File file);**

Видаляє файл по його імені з дерева

Складність O(logn)

* **void printTree();**

Графічно відображає поточний стан дерева. Допоміжна функція.

**Матеріали**

Т. Кормен, Ч. Лейзер - Алгоритмы. Построение и анализ