

Лабораторная работа 5. Деревья поиска без балансировки.

Реализовать структуру данных “Бинарное дерево поиска”. Реализовать функции:

- Создание дерева
- Добавление элемента
- Удаление элемента
- Поиск элемента
- Поиск максимума дерева
- Поиск минимума дерева

Реализовать структуру данных “Декартово дерево”. Реализовать функции:

- Создание дерева (инициализации)
- Поиск элемента
- Функция Split
- Функция Merge
- Реализовать добавление элемента используя неоптимизированный алгоритм (1 Split, 2 Merge)
- Реализовать добавление используя оптимизированный алгоритм (1 Split)
- Реализовать удаление элемента используя неоптимизированный алгоритм (2 Split, 1 Merge)
- Реализовать удаление используя оптимизированный алгоритм (1 Merge)
- Удаление элементов дерева (очистка памяти)

Демонстрация возможностей программы на основе меню.

Требования к коду:

Код должен быть написан согласно стандарту оформления кода RSDN (<https://rdsn.org/article/mag/200401/codestyle.XML>). Обращайте внимание на разделы кроме разделов с class.

Код программы должен быть оформлен в различных файлах (.h, .cpp), выделенных каждый для своей задачи.

Программа должна быть организована таким образом, чтобы части с консольным вводом-выводом не использовались в файлах, описывающих СД. Консольный ввод

Требования к среде разработки:

Предпочтительная IDE - Visual Studio. Можно любую другую, но в случае ошибок/вопросов помощи будет меньше.

Что необходимо повторить:

- На лекции разбирались: СД «Бинарное дерево», СД «Декартово дерево»
- Для обеих СД и их операций разбирались примеры кода

Что необходимо изучить:

- Кормен Т. и др. Алгоритмы. Построение и анализ:[пер. с англ.]. – Издательский дом Вильямс, 2009. (гл. 12, бинарные деревья поиска)
- Цикл статей «Декартовы деревья» <https://habr.com/ru/post/101818/>