

Лабораторная работа №7  
Динамические структуры данных. Бинарные деревья

| Оценка | Условия                                                                                                                                                 |
|--------|---------------------------------------------------------------------------------------------------------------------------------------------------------|
| 4-7    | Реализовать все необходимые методы для работы с бинарными деревьями — поиск, удаление, добавление, изменение, просмотр элементов помимо основной задачи |
| 8-10   | Выполнение <u>двух любых</u> доп заданий.                                                                                                               |

- 1) Найти расстояние между узлами в бинарном дереве.
- 2) Дано бинарное дерево. Все нулевые значения должны стать конечными — находиться внизу дерева. Преобразованное дерево не должно содержать узлов с нулевыми значениями, у которых есть потомки с ненулевыми значениями.
- 3) Алгоритм сжатия кода Хаффмана. Закодировать и декодировать сообщение, вывести таблицу и само дерево.
- 4) Найти максимальное независимое множество в бинарном дереве. Максимальное независимое множество — множество, элементы которого не являются соседями.
- 5) Спиральный обход бинарного дерева.
- 6) Обход границ бинарного дерева.
- 7) Найти путь с максимальной суммой между двумя узлами бинарного дерева. Вывести путь и сумму.
- 8) Для идеального бинарного дерева вывести значения чередующихся левых и правых узлов для каждого уровня сверху вниз.
- 9) Проверить сбалансированность бинарного дерева по высоте.
- 10) Преобразовать бинарное дерево в полное бинарное дерево.

---

ДОПОЛНИТЕЛЬНЫЕ ЗАДАНИЯ СДАЮТСЯ **ТОЛЬКО ВО ВРЕМЯ СДАЧИ ЛАБЫ:**

- 1) Реализовать словарь в бинарном дереве. В словаре храниться слово на английском и его перевод. Ввод происходит из файла со строками следующего типа: «cat=кот».
- 2) Дана матрица  $M \times N$  с нулями и единицами. Нули — вода, единицы — суша. Найти и раскрасить в разные цвета острова.
- 3) Дан массив строк. Найти максимальную подстроку присутствующую в каждой строке.