Для каждой задачи необходимо предоставить код решения на любом языке программировния. Код должен содержать поясняющие комментарии (или быть самодокументируемым). В начале кода должен содержаться комментирй с теоретической оценкой сложности алгоритма и затрат по памяти.

Задачи:

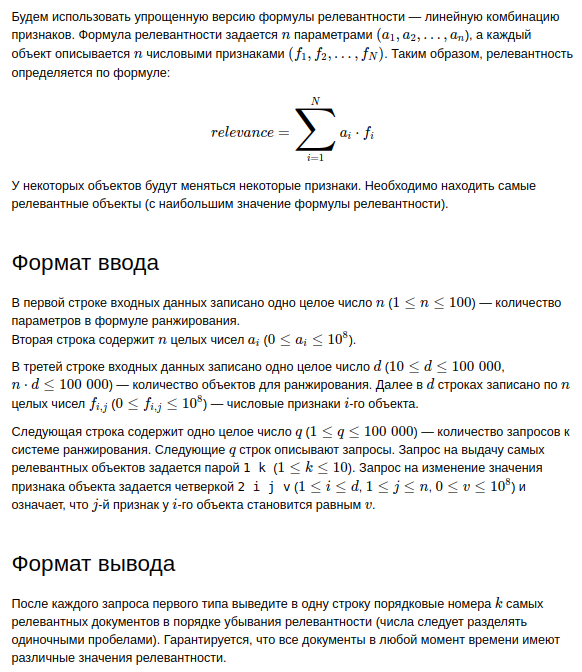
1. Сделать реверс односвзяного списка.

Пример:

Входные данные: 1, 2, 3, 4, 5

Результат: 5, 4, 3, 2, 1

2.



Пример:

Входные данные

2

1 100

10

1 2

2 1

3 1

4 1

5 1

6 1

7 1

8 1

9 1

10 1

4

1 2

1 10

2 4 1 1000

1 10

Результат

1 10

1 10 9 8 7 6 5 4 3 2

4 1 10 9 8 7 6 5 3 2

3. Инвертировать бинарное дерево поиска. Инвертировать дерево – значит прекомпоновать его элементы таким образом, чтобы узлы справа от материнского узла были больше, а слева-меньше.

4. Реализовать балансировку красно-черного дерева.

5. Написать класс, реализующий структуру данных на базе красно-черного дерева для хранения пар “ключ-значение”

6. Дана информация о времени заезда и отъезда посетителей отеля. Необходимо определить, в какой день посетителей в отеле единомоментно находилось больше всего.

Пример входных данных (один элемент данного листа – кортеж, содержащий дату заезда и отъезда одного посетителя): [(“2024-09-15”, “2024-09-15”), (“2024-09-14”, “2024-09-21”)]

7. Реализовать алгоритм Беллмана-Форда для произвольного графа

8. Релизовать венгерский алгоритм для решения задач комбинаторной оптимизации через графы