Задание 2.1. Реализовать алгоритмы бинарного и интерполяционного поиска числа x в массиве длины N, элементы которых - случайные целые числа в диапазоне от 0 до M. Вывести число операций сравнения, выполненных алгоритмом для заданных N, M и x.

Задание 2.2. Реализовать алгоритмы построения, обхода и балансировки дерева бинарного поиска (BST). На вход алгоритма подается последовательность целых положительных чисел a\_1, a\_2, ..., a\_n. Программа должна строить BST, добавляя узлы в порядке последовательности. Реализовать обходы дерева по возрастанию узлов и по убыванию узлов. Реализовать алгоритм нахождения k-го минимального ключа в дереве; на его основе сбалансировать построенное дерево (ротациями вправо и влево n/2-минимальный элемент помещается в корень, затем данная процедура рекурсивно повторяется для поддеревьев с корнями в дочерних узлах).

Задание 2.3. Реализовать алгоритм хеширования методом умножения с разрешением коллизий цепочками переполнения, линейного зондирования и двойным хешированием. В вычислительном эксперименте подобрать свою константу для метода умножения, сравнить ее с константой Кнута по наибольшей длине цепочек коллизий (множества ключей с равным хеш значением) для P наборов из N случайных ключей от 1 до R, при длине хеш-таблицы M.