

Контрольные вопросы:

- ☐ (5 б.) В каких ситуациях используются контейнеры типа множества и отображения?
- ☐ (5 б.) Каким требованиям должна удовлетворять качественная хэш-функция?
- ☐ (5 б.) Из-за чего в хэш-таблицах возникают коллизии и как можно их разрешать?
- ☐ (5 б.) Почему сложность основных операций хэш-таблиц в худшем случае $O(N)$?
- ☐ (5 б.) В чем заключается преимущество интерфейсов контейнера `Boost.Multiindex`?

Упражнения:

- ☐ (25 б.) Операция вставки нового элемента в контейнер типа `std::set` имеет сложность $O(\log N)$. Сгенерируйте N случайных чисел без повторений и добавьте их в указанный контейнер. Данный процесс имеет сложность $O(N \log N)$. Далее создайте контейнер типа `std::vector`, добавьте в него те же самые N чисел, после чего отсортируйте контейнер с помощью стандартного алгоритма `std::sort`. Данный процесс также имеет сложность $O(N \log N)$, отличие от сложности первого будет только в константе. Используя таймер, определите лидера.
- ☐ (25 б.) На семинаре мы рассмотрели хэш-функцию из `Boost`, применяемую для вычисления хэш-кода по набору значений. Исследуйте равномерность данной хэш-функции. Создайте структуру с полями различных типов, сгенерируйте достаточно большое количество случайных уникальных экземпляров данной структуры и для каждого вычислите хэш-код. Постройте график зависимости числа коллизий от кол-ва экземпляров.
- ☐ (25 б.) Аналогично предыдущей задаче исследуйте еще 9 хэш-функций из этой [статьи](#). Данные хэш-функции предназначены для строк, поэтому вам потребуется написать генератор уникальных случайных строк. Рекомендую ограничиться английским алфавитом в нижнем регистре – из 6 букв можно составить 308915776 уникальных строк. Для проверки своих результатов можете посмотреть некоторые результаты в этой [статье](#).
- ☐ (25 б.) Вам необходимо написать контейнер для хранения записей телефонного справочника, однако разные клиенты собираются использовать его по-разному. Городская типография собирается напечатать справочник и ей нужны записи в отсортированном по фамилии человека порядке. Рекламное агентство нуждается в произвольном доступе к записям справочника. Регулярный пользователь хочет за максимально короткое время находить нужную ему запись. Удовлетворите желания клиентов, создав контейнер на базе `Boost.Multiindex`. Напишите демонстрационный код с использованием всех интерфейсов этого контейнера.