Задача 2-1 (30 баллов). Пусть заданы n ключей $a_1, \ldots, a_n$ и m запросов на поиск k-й порядковой статистики $k_1, \ldots, k_m$, $m \geq 2$. Предложите алгоритм, который выполняет все эти запросы за время $O(n \log m + m)$. Указание: вообразите процесс одновременного (sic!) поиска ответов для всех k_i с помощью алгоритма Quick-Select, в котором на каждом вызове используется линейный детерминированный алгоритм поиска медианы. Очевидно, что одновременное знание всех k_i позволяет переиспользовать значительную часть результатов для разных k_i.

Задача 2-2 (25 баллов). Фиксируем натуральное k. Предложите структуру данных, способную выполнять два действия: Enqueue(x) — принять на вход ключ x; Get-Kth() — найти k-ю порядковую статистику среди принятых ключей. Учетная сложность Enqueue должна быть $O(1)$, сложность Get-Kth — $O(k)$ в худшем случае. Совет: группируйте вставки, используйте линейный алгоритм нахождения порядковой статистики.

Задача 2-3 (25 баллов). Пусть задана бинарная куча (min-heap) из n элементов. Придумайте алгоритм, находящий k минимальных элементов в ней за время $O(k \log k)$. Исходная куча при этом должна остаться без изменений, однако разрешается в процессе работы использовать дополнительную память. Совет: удаление элемента из бинарной кучи состоит в отделении корня и пары поддеревьев.

Задача 2-4 (40 баллов). Пусть задана бинарная куча из n элементов (представленная массивом). Предположим, что в конец массива были добавлены k новых элементов. Покажите, как преобразовать полученный массив длины $n + k$ в бинарную кучу (представленную массивом) за время $O(k + \log^2 n)$ (20 баллов), за время $O(k + \log n \log \log n)$ (40 баллов). Совет: используйте идеи из линейного алгоритма построения кучи.

Задача 2-5 (25 баллов). Предложите реализацию кучи, позволяющую добавить новый ключ за учетное время $O(1)$ и извлечь минимальный за время $O(\log n)$ в худшем случае (здесь n обозначает текущее число элементов в куче). Совет: группируйте вставки.

Задача 2-6 (20 баллов). Семейство $\mathcal{H}$ хеш-функций h, принимающих m значений, называется (C, k)-*независимым*, если $P[h(x_1) = y_1, \ldots, h(x_k) = y_k] \leq Cm^{-k}$, где x_i — произвольные различные ключи, y_i произвольные (не обязательно различные) хеш-значения, а вероятность берется относительно случайного и равномерного выбора $h \in \mathcal{H}$. Пусть выбрано простое $p > m$. Рассмотрим случайно, независимо и равномерно выбранные коэффициенты $a_0, a_1, \ldots, a_{k-1} \in \mathbb{F}_p$ $(a_{k-1} \neq 0)$ и образуем хеш-функцию на множестве ключей $\mathbb{F}_p$:

$$h(x) = ((a_0 + a_1 x + \ldots + a_{k-1} x^{k-1}) \bmod p) \bmod m.$$

Докажите, что для любого k существует константа C, такая что для всех p, m (удовлетворяющих указанным требованиям) данное семейство хеш-функций является (C, k)-независимым.

Задача 2-7 (30 баллов). Семейство $\mathcal{H}$ хеш-функций h, принимающих m значений, называется C-*универсальным*, если для произвольных различных ключей x и y справедливо $P[h(x) = h(h)] \leq C/m$. Рассмотрим семейство хеш-функций

$$h_a(x) = \lfloor ((ax) \bmod 2^w) / 2^{w-l} \rfloor,$$

определенных на w-битных ключах x ($0 \leq x < 2^w$) и дающих l-битные хеш-значения, где a — случайное нечетое число на отрезке $[1, 2^w]$. Покажите, что существует абсолютная константа C, такая что данное семейство является C-универсальным.