Воспользуемся структурой “декартово дерево” со случайными приоритетами. Ключами (x) будут являться сами числа из множества. Помимо этого в каждой вершине(v) будем хранить два вспомогательных значения – l\_val – самый маленький ключ в поддереве v и cont\_len – длину наибольшей непрерывной последовательность, содержащийся в поддереве v, начинающийся с элемента l\_val.

Очевидно, что для листа v значение l\_val = v.x, а cont\_val = 1;

Покажем, как обновлять значения для вершин. Пусть значения для сыновей v посчитаны. Тогда разберем случаи:

1. Детей у вершины нет. Как было сказано v.l\_val = v.x; а v.cont\_len = 1;
2. У вершины только левый сын. Тогда очевидно v.l\_val = v.child\_left.l\_val; Если v.child\_left.cout\_len == (v.x – x.child\_left.x), то v.x является следующим в непрерывной последовательность, начинающейся с v.l\_val, т.е. v.count\_len = v.child\_left.cont\_len + 1. Иначе v.count\_len = v.child\_left.cont\_len.
3. У вершины только правый сын. Тогда v.l\_val = v.x. Если v.child\_right.l\_val == v.x + 1, то последовательность, начинающаяся с v.child\_right.l\_val дополняется числом v.x слева т.е. v.cont\_len = 1 + v.child\_right.cont\_len. Иначе v.cont\_len = 1;
4. У вершины есть оба сына. Обновим вершину так, словно у нее есть только левый сын. Если v.l\_val + v.cont\_len = v.x – 1, то это значит, что последовательность заканчивается на элементе v.x, а значит мы можем продлить ее элементами из правого поддерева: Если v.child\_right.l\_val == v.x + 1, то последовательность, дополняется элементами из правого поддерева т.е. v.cont\_len += v.child\_right.cont\_len. Иначе дополнить ее нельзя, обработка завершена.

Отметим, что обновление происходит за O(1);

Добавление и удаление производится так же, как в обычном декартовом дереве.

Для ответа на вопрос выполним split() по ключу x и будем рассматривать только правое из полученных деревьев, т.е. дерево, в котором все ключи >= x. По свойствам декартового дерева, в корне полученного дерева значения l\_val и l\_cont\_len будут посчитаны корректно, а значит ответить на вопрос можно следующим образом:

1. Если root.l\_val > x, то значения x в множестве нет, а значит x – ответом.
2. Иначе максимальная последовательность в множестве, начинающаяся с x имеет длину root.cont\_len, а значит ответом будет x + root.cont\_len;

Исходя из того, что обновление вершины происходит за O(1), получим, что все операции будут выполняться за O(log n);