Собеседование в Сбер

Поймем, что если на каком-то круге Ваня не решил ни одной задачи, то он и дальше не сможет ничего решить. Значит, Ваня сделает не больше n кругов решения задач. Тогда можем сделать решение за $O(n^2)$. Будем n раз идти по списку задач и решать задачи, которые можем.

Сделаем более быстрое решение. Запомним для каждой сложности задачи все позиции, где она встречается. Будем поддерживать структуру данных, в которой для каждой сложности задачи, меньше или равной текущему уровню Вани, запомним индекс нерешенной задачи, которую Ваня решит раньше всего. Изначально в ней лежит индекс самой первой 1 в массиве. Затем, пока структура не пустая, нужно найти минимальный индекс задачи, который больше последней решенной задачи, и достать его из структуры. После решения задачи навык увеличивается на 1, значит, должны добавить в структуру минимальный индекс задачи, равный текущему навыку и больший, чем текущий индекс. Аналогично нужно добавить задачу, у которой сложность равна сложности последней решенной.

Для решения будем использовать структуру данных std::set, тогда время работы будет $O(n \log(n))$, так как проводим $O(n)$ операций с сетом, и каждая требует $O(\log(n))$ действий.