

## Общие требования

Задача про футбольную команду в этой домашней работе является одной из трех задач на ревью в этом семестре. То есть ее нужно не только сдать в систему, но и прислать теоретическое решение с доказательствами, а также прислать интерфейс, а затем код на Code Review.

В задачах этой домашней работы дополнительно нельзя пользоваться никакими стандартными алгоритмами сортировки, бинарного поиска и сложными структурами данных, например, кучей и деревом поиска. Если для решения какой-то задачи необходимо ими воспользоваться, их нужно реализовать самим.

**Задача 2-1 (35 баллов).** Дано две последовательности целых чисел:  $x_1, x_2, \dots, x_n$  и  $y_1, y_2, \dots, y_k$ . Нужно найти их наибольшую общую подпоследовательность, то есть такую последовательность  $a_1, a_2, \dots, a_m$ , что  $a_1 = x_{i_1} = y_{j_1}, a_2 = x_{i_2} = y_{j_2}, \dots, a_m = x_{i_m} = y_{j_m}, 1 \leq i_1 < i_2 < \dots < i_m \leq n, 1 \leq j_1 < j_2 < \dots < j_m \leq k$ .

В первой строке входа число  $n$ , во второй — последовательность  $x_1, x_2, \dots, x_n$ , в третьей — число  $k$ , в четвертой — последовательность  $y_1, y_2, \dots, y_k$ . Выведите одно число — длину наибольшей общей подпоследовательности.

*Ограничения.*  $1 \leq n, k \leq 2000$ . Числа не превосходят по модулю  $10^9$ .

Пример входа	Пример выхода
5 1 2 3 4 5 5 1 3 5 7 9	3
1 1 1 2	0

**Задача 2-2 (40 баллов).** Вы хотите набрать футбольную команду. У каждого игрока своя эффективность, она описывается одним целым числом. Чем больше число, тем больше эффективность футболиста. Обязательным условием для любой команды является сплоченность. Если один из игроков играет сильно лучше всех остальных, его будут недолюбливать, и команда распадется. Поэтому эффективность любого игрока команды не должна превышать сумму эффективностей любых двух других игроков. Ваша задача — набрать команду, которая будет удовлетворять условию сплоченности, и при этом иметь наибольшую суммарную эффективность.

В первой строке входа задано число  $n$  ( $1 \leq n \leq 100000$ ). Во второй строке — эффективности каждого из  $n$  игроков — положительные целые числа, не превосходящие  $2^{31} - 1$ . Выведите две строки. В первую запишите наибольшую возможную сумму эффективностей игроков в команде. Во вторую строку выведите номера всех игроков, которых нужно взять в команду, в порядке возрастания. Игроки пронумерованы от 1 до  $n$  в том порядке, в котором заданы их эффективности на входе. Если существует несколько способов набрать команду максимальной эффективности, выведите любой из них.

Пример входа	Пример выхода
5 3 2 5 4 1	14 1 2 3 4
5 1 2 4 8 16	24 4 5

**Задача 2-3 (65 баллов).** На плоскости даны  $n$  точек с координатами  $(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)$ . Необходимо накрыть по крайней мере  $k$  из них кругом с центром на оси  $Ox$ . Найдите наименьший возможный радиус такого круга с точностью до  $10^{-3}$ .

В первой строке входа даны числа  $n$  и  $k$ ,  $1 \leq k \leq n \leq 10000$ . В следующих  $n$  строках — по два целых числа, координаты соответствующей точки. Координаты не превосходят по модулю 1000. Выведите радиус круга ровно с 6 знаками после запятой. Ваш ответ должен отличаться от правильного не более, чем на  $10^{-3}$ , вывести нужно при этом 6 знаков, чтобы избежать ошибок из-за округления.

Пример входа	Пример выхода
3 3 0 5 3 4 -4 -3	5.000000
3 2 0 1 2 1 1 100	1.414214

**Задача 2-4 (35 баллов).** Вам дано  $n$  упорядоченных по возрастанию последовательностей целых чисел, каждая из которых имеет длину  $m$ . Необходимо слить их в одну упорядоченную по возрастанию последовательность целых чисел длины  $mn$ . Сложность алгоритма не должна превышать  $O(mn \log n)$ , затраты памяти —  $O(mn)$ .

В первой строке входа два целых числа  $n, m$  ( $1 \leq m, n \leq 1000$ ). В следующих  $n$  строках по  $m$  целых чисел в каждой, числа в каждой строке упорядочены по возрастанию. Числа не превосходят по модулю  $10^9$ . Выведите все  $mn$  чисел, упорядоченных по возрастанию.

Пример входа	Пример выхода
2 5 1 3 5 7 9 2 4 6 8 10	1 2 3 4 5 6 7 8 9 10
4 2 1 4 2 8 3 7 5 6	1 2 3 4 5 6 7 8