

Задача А. Дубы

Имя входного файла: `oaks.in`
Имя выходного файла: `oaks.out`
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

На аллее перед зданием Министерства Обороны в ряд высажены n дубов. В связи с грядущим приездом главнокомандующего, было принято решение срубить несколько деревьев для придания аллее более милитаристического вида.

Внутренние распорядки министерства позволяют срубить дуб только в двух случаях:

- Если и ближайший дуб слева, и ближайший дуб справа строго ниже, чем данный дуб.
- Если и ближайший дуб слева, и ближайший дуб справа строго выше, чем данный дуб.

В частности, согласно этому правилу, нельзя срубить крайний левый и крайний правый дуб.

Министр хочет выработать такой план вырубki, чтобы в итоге осталось несколько дубов, высоты которых образуют неубывающую последовательность, то есть чтобы каждый дуб был не ниже, чем все дубы, стоящие слева от него. При этом, как человек любящий флору, министр хочет, чтобы было срублено минимальное возможное количество деревьев.

Помогите сотрудникам министерства составить оптимальный план вырубki аллеи или выяснить, что срубить дубы соответствующим образом невозможно.

Формат входного файла

Первая строка входного файла содержит целое число n — количество дубов, растущих на аллее ($2 \leq n \leq 200$). Вторая строка содержит n чисел — высоты дубов, приведенные слева направо. Высоты дубов — положительные целые числа, не превышающие 1000.

Формат выходного файла

Если оставить последовательность дубов с неубывающими высотами невозможно, выходной файл должен содержать только одно число -1 .

В случае, если искомым план существует, в первую строку выходного файла выведите целое число m — минимальное количество дубов, которые необходимо срубить. В следующие m строк выведите оптимальный план вырубki деревьев — номера дубов в том порядке, в котором их следует срубить, по одному номеру на строке.

Дубы нумеруются слева направо натуральными числами от 1 до n .

Если планов с наименьшим числом срубаемых дубов несколько, выведите любой из них.

Примеры

<code>oaks.in</code>	<code>oaks.out</code>
5	2
3 2 4 8 5	2
	4
5	0
4 5 5 5 6	
6	-1
1 1 3 3 2 2	
6	-1
400 300 310 300 310 500	

Задача В. Почтовые отделения

Имя входного файла: `post.in`
Имя выходного файла: `post.out`
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

Вдоль прямой дороги расположены деревни. Дорога представляется целочисленной осью, а расположение каждой деревни задается одним целым числом — координатой на этой оси. Никакие две деревни не имеют одинаковых координат. Расстояние между двумя деревнями вычисляется как модуль разности их координат.

В некоторых, не обязательно во всех, деревнях будут построены почтовые отделения. Деревня и расположенное в ней почтовое отделение имеют одинаковые координаты. Почтовые отделения необходимо расположить в деревнях таким образом, чтобы общая сумма расстояний от каждой деревни до ближайшего к ней почтового отделения была минимальной.

Формат входного файла

В первой строке содержатся два целых числа: количество деревень n , ($1 \leq n \leq 300$) и количество почтовых отделений m ($1 \leq m \leq 30$), ($m \leq n$). Вторая строка содержит n целых чисел в возрастающем порядке, являющихся координатами деревень. Для каждой координаты x верно, что ($1 \leq x \leq 10000$).

Формат выходного файла

Первая строка выходного файла должна содержать одно целое число — общую сумму расстояний от каждой деревни до её ближайшего почтового отделения. Вторая строка содержит m целых чисел в возрастающем порядке. Эти числа являются искомыми координатами почтовых отделений. Если для заданного расположения деревень есть несколько решений, то достаточно найти одно из них.

Примеры

post.in	post.out
10 5	9
1 2 3 6 7 9 11 22 44 50	2 7 22 44 50

Задача С. Интересное число

Имя входного файла: `number.in`
Имя выходного файла: `number.out`
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

Для заданного числа n найдите наименьшее положительное целое число s суммой цифр n , которое делится на n .

Формат входного файла

Во входном файле содержится целое число n ($1 \leq n \leq 1000$).

Формат выходного файла

Выходной файл должен содержать искомое число. Ведущие нули выводить не разрешается.

Пример

number.in	number.out
1	1
10	190