

Задачи на 3 **Задача А. Наибольшая 7-складная подпоследовательность**

Имя входного файла: `sequence2.in`
Имя выходного файла: `sequence2.out`
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

Для заданной числовой последовательности a_1, a_2, \dots, a_n требуется найти максимальную 7-складную подпоследовательность.

Для 7-складной подпоследовательности $a_{k_1}, a_{k_2}, \dots, a_{k_t}$ ($k_1 < k_2 < \dots < k_t$) верно, что $a_{k_i} + a_{k_{(i+1)}} \equiv 0 \pmod{7}$ при $1 \leq i < t$ (утверждение « $a \equiv 0 \pmod{7}$ » означает, что « a кратно 7»). Подпоследовательность из одного элемента полагается 7-кратной по определению.

Формат входного файла

В первой строке входного файла записано одно натуральное число N ($1 \leq N \leq 1000$) — количество чисел в исходной последовательности. Далее следует N натуральных чисел, не превосходящих 10^9 — сама последовательность.

Формат выходного файла

Выведите 7-кратную подпоследовательность максимальной длины. Если решений несколько — выведите любое.

Пример

<code>sequence2.in</code>	<code>sequence2.out</code>
4	3 4 17
3 6 4 17	

Задачи на 4 **Задача В. Телепортация**

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

Великий учитель Шри Япутра упорным трудом и медитацией овладел древней методикой телепортации. Теперь, по дороге из монастыря на плантацию какао, он может не останавливаться в каждом кабаке на своём пути, а телепортироваться от кабака i до кабака j ($i < j$), минуя все промежуточные кабаки.

Кабаки на пути Шри Япутры пронумерованы числами от $1 \dots N$ (для простоты можете считать, что монастырь это кабак с номером 0, а плантация какао — кабак

номер $N + 1$). Шри Япутра, стоя у кабака с номером i , может телепортироваться до кабаков с номерами $i + y_j$ ($1 \leq j \leq K$). Если Шри Япутра оказывается около кабака, то не зайти в него он не может, где учитель обязательно выпьет x_i грамм какао за разговорами о природе Дао.

Будучи обескураженным количеством вариантов, которыми теперь великий учитель может добраться от своего монастыря до плантации какао, Шри Япутра просит вас помочь найти вариант, при котором он выпьет меньше всего какао.

Формат входного файла

В первой строке входного файла вводится два натуральных числа $N \leq 100$ и $K \leq 10$ — количество кабаков на пути Шри Япутры и количество способов телепортироваться. В следующей строке вводятся K чисел $2 \leq y_i \leq 20$ — расстояния в кабаках, на которые может телепортироваться великий учитель. В третьей строке вводятся N натуральных чисел x_i , не превосходящих 100 — количество какао в граммах, которое великий учитель выпьет если зайдёт в i -й из кабаков на своём пути.

Формат выходного файла

Выведите сначала количество грамм, которые выпьет Шри Япутра, а затем номера кабаков в порядке возрастания, которые он посетит. Если ответов несколько — выведите любой. Если ответа не существует — выведите число -1 .

Пример

<code>stairs.in</code>	<code>stairs.out</code>
8 3	11
1 2 3	1 4 6
1 2 5 4 7 6 9 9	
5 1	-1
100	
1 2 3 4 5	

Задача С. Кино

Имя входного файла: `cinema.in`
Имя выходного файла: `cinema.out`
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Трое друзей летом посмотрели несколько фильмов. Для каждого школьника известно, какие фильмы и в каком порядке он посмотрел (естественно, если фильм кому-то из ребят сильно понравился, он мог его пересмотреть несколько раз). Так

как друзья любят смотреть фильмы вместе, втроем они посмотрели максимальное возможное количество. Сколько же раз они встречались вместе?

Формат входного файла

В первой строке входного файла заданы три числа: N , M и K — количества фильмов, просмотренных каждым из друзей ($1 \leq N, M, K \leq 300$). В следующих трех строках выписаны номера фильмов, просмотренных ими. У друзей огромные планы на будущее, поэтому фильмы нумеруются числами между 1 и 10^9 .

Формат выходного файла

В первой строке выведите единственное число — максимальное кол-во просмотренных фильмов.

В следующей строке выведите через пробел номера просмотренных фильмов.

Пример

cinema.in	cinema.out
3 3 4	2
1 2 3	1 3
1 3 10	
3 1 10 3	

Задача D. Сумма кубов

Имя входного файла:

Имя выходного файла:

Ограничение по времени: 1 секунда

Ограничение по памяти: 64 мегабайта

Известно, что любое натуральное число можно представить в виде суммы не более чем четырех квадратов натуральных чисел. Вася решил придумать аналогичное утверждение для кубов — он хочет узнать, сколько кубов достаточно для представления любого числа. Его первая рабочая гипотеза — восемь.

Выяснилось, что почти все числа, которые Вася смог придумать, представляются в виде суммы не более чем восьми кубов. Однако число 239, например, не допускает такого представления. Теперь Вася хочет найти какие-либо другие такие числа, а также, возможно, какую-либо закономерность в представлениях всех остальных чисел, чтобы выдвинуть гипотезу относительно вида всех чисел, которые не представляются в виде суммы восьми кубов.

Помогите Васе написать программу, которая проверяла бы, возможно ли представить данное натуральное число в виде суммы не более чем восьми кубов натуральных чисел, и если это возможно, то находила бы какое-либо такое представление.

Формат входного файла

Вводится натуральное число $N \leq 2 * 10^9$.

Формат выходного файла

Требуется вывести не более восьми натуральных чисел, кубы которых в сумме дают N . Если искомого представления не существует, то в выходной файл необходимо вывести слово IMPOSSIBLE.

Пример

17	2 2 1
239	IMPOSSIBLE

Задача E. Обменные башни

Имя входного файла:

Имя выходного файла:

Ограничение по времени: 1 секунда

Ограничение по памяти: 64 мегабайта

Как и в предыдущих задачах, дано три стержня, на первом из которых наде-то n дисков различного размера. Необходимо их переместить на стержень 3 по следующим правилам:

Самый маленький диск (номер 1) можно в любой момент переложить на любой стержень. Перемещение диска номер 1 со стержня a на стержень b будем обозначать 1 a b.

Можно поменять два диска, лежащих на вершине двух стержней, если размеры этих дисков отличаются на 1. Например, если на вершине стержня с номером a лежит диск размером 5, а на вершине стержня с номером b лежит диск размером 4, то эти диски можно поменять местами. Такой обмен двух дисков будем обозначать 0 a b (указываются номера стержней, верхние диски которых обмениваются местами).

Для данного числа дисков n , не превосходящего 10, найдите решение головоломки. Вам не нужно находить минимальное решение, но количество совершенных перемещений не должно быть больше 200000.

Формат входного файла

Формат выходного файла

Пример

1	1 1 3
2	1 1 3 0 1 3 1 1 3

Задача F. Циклические башни

Имя входного файла:

Имя выходного файла:

Ограничение по времени: 1 секунда

Ограничение по памяти: 64 мегабайта

На дорогах Ханоя было введено одностороннее круговое движение, поэтому теперь диск со стержня 1 можно перекладывать только на стержень 2, со стержня 2 на 3, а со стержня 3 на 1.

Решите головоломку с учетом этих ограничений. Вам не нужно находить минимальное решение, но количество совершенных перемещений не должно быть больше 200000, при условии, что количество дисков не превосходит 10.

Формат входного файла

Формат выходного файла

Пример

1	1 1 2 1 2 3
2	1 1 2 1 2 3 2 1 2 1 3 1 2 2 3 1 1 2 1 2 3

Задача G. Сортирующие башни

Имя входного файла:

Имя выходного файла:

Ограничение по времени: 1 секунда

Ограничение по памяти: 64 мегабайта

Первоначально все диски лежат на стержне номер 1. Переместите диски с нечетными номерами на стержень номер 2, а с четными номерами - на стержень номер 3.

Вам не нужно находить минимальное решение, но количество совершенных перемещений не должно быть больше 200000, при условии, что количество дисков не превосходит 10.

Формат входного файла

Формат выходного файла

Пример

2	1 1 2 2 1 3
3	1 1 2 2 1 3 1 2 3 3 1 2 1 3 2

Задача H. Несправедливые башни

Имя входного файла:

Имя выходного файла:

Ограничение по времени: 1 секунда

Ограничение по памяти: 64 мегабайта

В Ханое несправедливо запретили класть самый маленький диск (номер 1) на средний колышек (номер 2).

Решите головоломку с учетом этих ограничений. Вам не нужно находить минимальное решение, но количество совершенных перемещений не должно быть больше 200000, при условии, что количество дисков не превосходит 10.

Формат входного файла

Формат выходного файла

Пример

2	1 1 3 2 1 2 1 3 1 2 2 3 1 1 3

Задачи на 5 Задача I. Количество ПСП

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

Посчитайте количество правильных скобочных последовательностей длины $2n$ (n открывающихся скобок и n закрывающихся), составленных из круглых и квадратных скобок так, что внутри любой пары круглых скобок нет квадратных скобок.

Формат входного файла

В единственной строке через пробел записано целое неотрицательное число n , не превосходящее 1000.

Формат выходного файла

Выведите остаток от деления количества искомых правильных скобочных последовательностей на $10^9 + 7$.

Примеры

<code>quant.in</code>	<code>quant.out</code>
1	2
2	7

Задача J. Покупка билетов

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

За билетами на премьеру нового мюзикла выстроилась очередь из N человек, каждый из которых хочет купить 1 билет. На всю очередь работала только одна касса, поэтому продажа билетов шла очень медленно, приводя «постояльцев» очереди в отчаяние. Самые сообразительные быстро заметили, что, как правило, несколько билетов в одни руки кассир продаёт быстрее, чем когда эти же билеты продаются по одному. Поэтому они предложили нескольким подряд стоящим людям отдавать деньги первому из них, чтобы он купил билеты на всех.

Однако для борьбы со спекулянтами кассир продавала не более 3-х билетов в одни руки, поэтому договориться таким образом между собой могли лишь 2 или 3 подряд стоящих человека.

Известно, что на продажу i -му человеку из очереди одного билета кассир тратит A_i секунд, на продажу двух билетов — B_i секунд, трех билетов — C_i секунд. Напишите программу, которая подсчитает минимальное время, за которое могли быть обслужены все покупатели.

Обратите внимание, что билеты на группу объединившихся людей всегда покупает первый из них. Также никто в целях ускорения не покупает лишних билетов (то есть билетов, которые никому не нужны).

Формат входного файла

Во входном файле записано сначала число N — количество покупателей в очереди ($1 \leq N \leq 5000$). Далее идет N троек натуральных чисел A_i, B_i, C_i . Каждое из этих чисел не превышает 3600. Люди в очереди нумеруются начиная от кассы.

Формат выходного файла

В выходной файл выведите одно число — минимальное время в секундах, за которое могли быть обслужены все покупатели.

Пример

tickets.in	tickets.out
5	12
5 10 15	
2 10 15	
5 5 5	
20 20 1	
20 1 1	

Задача К. Ханойская сортировка

Имя входного файла:

Имя выходного файла:

Ограничение по времени: 1 секунда

Ограничение по памяти: 64 мегабайта

Кто-то перепутал Ханойскую головоломку и разместил диски на первом стержне не соблюдая правила игры. Переложите эти диски так, чтобы они оказались на одном из стержней строго в порядке возрастания номеров.

Программа получает на вход число дисков $n \leq 10$. Во второй строке записаны n чисел — номера дисков на первом стержне сверху вниз.

Перемещать диск можно только в том случае, если он кладется на диск большего номера или на пустой стержень. Выведите последовательность перекладываний, размещающая диски на любом стержне в порядке возрастания номеров. Формат вывода одного перекладывания: $A B C$, где A номер перемещаемого диска ($1 \leq A \leq n$), B — номер стержня с которого снимается диск, C — номер стержня на который кладется диск. Количество перекладываний не должно быть больше 200000.

Формат входного файла

Формат выходного файла

Пример

3	2 1 2
2 1 3	1 1 3
	2 2 1
	1 3 1

Задачи на 5+ Задача L. Восстановление

Имя входного файла: `recover.in`

Имя выходного файла: `recover.out`

Ограничение по времени: 1 секунда

Ограничение по памяти: 64 мегабайта

Вася обнаружил ошибку в своей программе, которая удаляет все символы из строки кроме "(" и ")". Оказывается, некоторые символы заменяются на что-то нечитаемое.

Теперь его заинтересовал вопрос, сколько различных правильных скобочных последовательностей длины $2n$ могут являться результатом исправленного алгоритма, то есть не будут противоречить данным, которые он таки не потерял.

Формат входного файла

Единственная строка входного файла содержит строку из круглых скобок и знаков вопроса, где вопросами обозначены утраченные символы. Длина строки не превосходит 1000.

Формат выходного файла

Выведите одно число — количество различных скобочных последовательностей, удовлетворяющих Васиному шаблону, по модулю $10^9 + 7$.

Пример

recover.in	recover.out
(??()?)	2