

Задача А. Астронавигация

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	256 мегабайт

В Межгалактическую Империю входят n планет. Каждая планета управляется губернатором, подчиняющимся непосредственно Императору.

Недавно Император объявил об открытии межпланетного сообщения. В рамках этого проекта будут построены межпланетные космодромы и некоторые пары космодромов будут соединены двунаправленными рейсами.

Каждый губернатор хочет, чтобы его планета была соединена ровно одним двунаправленным рейсом с какой-либо другой планетой (при отсутствии рейсов губернатор не получит финансирование из имперской казны на межпланетный космодром, при наличии рейсов с разных планет у губернатора возникнут проблемы с контролем транзитных потоков).

По заданному n определите, какому наибольшему числу губернаторов удастся удовлетворить свои пожелания.

Формат входных данных

Первая строка входных данных содержит одно целое число n ($2 \leq n < 10^{10000}$).

Формат выходных данных

Выведите одно целое число — наибольшее количество губернаторов, которым удастся удовлетворить свои пожелания.

Система оценки

Задача состоит из трёх подзадач.

- В первой подзадаче $n < 10^9$. Эта подзадача оценивается в 33 балла.
- Во второй подзадаче $n < 10^{18}$. Эта подзадача оценивается в 35 баллов.
- В третьей подзадаче дополнительных ограничений нет. Эта подзадача оценивается в 32 балла.

Пример

стандартный ввод	стандартный вывод
4	4

Замечание

В примере к задаче можно соединить первую и вторую планеты, а также третью и четвёртую. Тогда пожелания всех четырёх губернаторов будут выполнены.

Задача В. Беспосадочные перелёты

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	256 мегабайт

Самолёт вылетает из города М в город N в $s1_h$ часов и $s1_m$ минут местного времени, а прибывает в город N в $f1_h$ часов и $f1_m$ минут местного времени. Обратный рейс вылетает из города N в город M в $s2_h$ часов и $s2_m$ минут местного времени, а прибывает в город M в $f2_h$ часов и $f2_m$ минут местного времени.

Известно, что время в городе M и городе N отличается на целое число часов (не более 8), что самолёт летит не быстрее скорости вращения Земли (то есть время прибытия даже с поправкой на таймзону не может быть меньше времени отправления), что время в пути в любом направлении составляет не более 10 часов, и что разница между временем в пути в прямом и обратном направлении строго меньше одного часа.

Найдите точное время в пути для обоих рейсов.

Формат входных данных

Первая строка входных данных содержит четыре целых числа $s1_h$, $s1_m$, $f1_h$, $f1_m$ ($0 \leq s1_h, f1_h \leq 23$, $0 \leq s1_m, f1_m \leq 59$) — местное время отправления рейса из M в N и местное время прибытия рейса из M в N, соответственно (каждое время задано парой «часы, минуты»). Вторая строка содержит четыре целых числа $s1_h$, $s1_m$, $f1_h$, $f1_m$ — местное время отправления и прибытия обратного рейса.

Гарантируется, что входные данные удовлетворяют ограничениям, описанным в условии.

Формат выходных данных

В первой строке выведите два целых числа — часы и минуты, проведённые самолётом в пути из города M в город N. Во второй строке выведите два целых числа — часы и минуты, проведённые самолётом в пути из города N в город M.

Система оценки

Задача состоит из трёх подзадач.

- В первой подзадаче время в пути туда и назад одинаково. Эта подзадача оценивается в 26 баллов.
- Во второй подзадаче гарантируется, что время вылета и время прилёта обоих рейсов принадлежит одним и тем же суткам. Эта подзадача оценивается в 32 балла.
- В третьей подзадаче дополнительных ограничений нет. Эта подзадача оценивается в 42 балла.

Примеры

стандартный ввод	стандартный вывод
23 15 7 30	4 15
20 40 21 22	4 42
16 10 18 26	1 16
23 45 0 0	1 15

Задача С. Выбери звезду

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	256 мегабайт

Назовём *звездой* самопересекающуюся замкнутую ломаную из n звеньев со следующими свойствами:

- Множество вершин ломаной совпадает с множеством вершин правильного n -угольника.
- Длина всех n звеньев одинакова.
- Если стартовать в какой-либо вершине и каждый раз переходить в вершину, соединённую с текущей вершиной звеном, то можно посетить все n вершин.
- Каждое звено пересекает во внутренней точке как минимум одно другое звено.

Вам даны два числа n_1 и n_2 . Ваша задача — найти звезду, количество звеньев которой было бы в интервале от n_1 до n_2 включительно, а длина звена была бы максимальной.

Формат входных данных

Первая строка входных данных содержит два целых числа n_1 и n_2 ($3 \leq n_1 \leq n_2 \leq 10^6$).

Формат выходных данных

Если ни одной звезды в требуемом интервале не существует, выведите единственное целое число -1 .

Иначе выведите два целых числа — количество вершин звезды с максимальной длиной звена и тип звена. Тип звена определяется следующим образом: занумеруем вершины по часовой стрелке, начиная с нуля. Тогда тип звена равен **меньшему** из номеров вершин, с которой соединена звеном нулевая вершина. Если решений несколько, выведите то, в которой первое число меньше.

Система оценки

Задача состоит из трёх подзадач.

- В первой подзадаче $3 \leq n_1 \leq n_2 \leq 7$. Эта подзадача оценивается в 18 баллов.
- Во второй подзадаче $n_1 = n_2$. Эта подзадача оценивается в 34 балла.
- В третьей подзадаче дополнительных ограничений нет. Эта подзадача оценивается в 48 баллов.

Примеры

стандартный ввод	стандартный вывод
3 4	-1
4 5	5 2

Задача D. Гигантский аэробус

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	256 мегабайт

Салон самолёта «BehemothJet» состоит из n рядов кресел. Каждый ряд состоит из g групп кресел, разделённых $g - 1$ проходами. i -я группа кресел состоит из k_i кресел.

Менеджеры дискаунтера «Виктория» называют рассадку пассажиров *выгодной*, если не существует двух свободных соседних кресел внутри какой-либо группы какого-либо ряда. Пассажиры же считают рассадку *приемлемой*, если существует **хотя бы один ряд**, содержащий пару свободных кресел непосредственно справа и непосредственно слева от некоторого прохода.

Например, рассмотрим следующие рассадки на трёх рядах и двух группах (левая — два кресла, правая — три).

1) X. X.X	2) .. XXX	3) .X XXX	4) X. XXX
.X XX.	XX X.X	X. .XX	XX XXX
XX .XX	X. .XX	XX X.X	.X X..

точка — свободное место, X — занятое, пробел — проход.

Рассадка 1 является выгодной, но не является приемлемой (нет двух свободных мест рядом в одном ряду даже через проход). Рассадка 2 не является выгодной (есть два места рядом в левой группе первого ряда) и является приемлемой (два свободных места рядом через проход в третьем ряду). Рассадка 3 является выгодной и приемлемой (два свободных места рядом через проход во втором ряду). Рассадка 4 не является выгодной (есть два места рядом в правой группе третьего ряда) и не является приемлемой (нет двух свободных мест в одном ряду через проход).

По заданным n , g и k_i найдите количество рассадок, которые одновременно являются и выгодными, и приемлемыми. Две рассадки считаются различными, если найдётся такое кресло, которое занято в одной из раскладок и свободно в другой. Так как ответ может быть очень большим, выведите остаток от его деления на 998 244 353.

Формат входных данных

Первая строка входных данных содержит два целых числа n и g ($1 \leq n \leq 100$, $2 \leq g \leq 100$) — количество рядов и групп кресел в BehemothJet, соответственно. Вторая строка содержит g целых чисел k_i ($1 \leq k_i \leq 100$) — количество кресел в каждой группе.

Формат выходных данных

Выведите одно целое число — остаток от деления количества рассадок, которые являются и выгодными, и приемлемыми, на 998 244 353.

Система оценки

Задача состоит из нескольких подзадач

- В первой подзадаче $g = k_1 = k_2 = 2$. Эта подзадача оценивается в 8 баллов.
- Во второй подзадаче все k_i не превосходят 2. Эта подзадача оценивается в 13 баллов.
- В третьей подзадаче $g = 2$. Эта подзадача оценивается в 26 баллов.
- В четвёртой подзадаче сумма всех k_i не превосходит 20. Эта подзадача оценивается в 10 баллов.
- В пятой подзадаче дополнительных ограничений нет. Эта подзадача оценивается в 43 балла.

Пример

стандартный ввод	стандартный вывод
3 2 2 3	1178

Задача Е. Древняя система счисления

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	256 мегабайт

Таблица из Википедии показывает способ записи римских чисел:

Значение разряда	Тысячи	Сотни	Десятки	Единицы
1	M	C	X	I
2	MM	CC	XX	II
3	MMM	CCC	XXX	III
4		CD	XL	IV
5		D	L	V
6		DC	LX	VI
7		DCC	LXX	VII
8		DCCC	LXXX	VIII
9		CM	XC	IX

Заметим, что:

- Числа 4, 9, 40, 90, 400 and 900 записываются в реверсивной нотации, где первый символ вычитается из второго (например, для 40 (XL) 'X' (10) вычитается из 'L' (50)). Это **единственные** места, где реверсивная нотация используется.
- Число, содержащее несколько десятичных цифр, строится дописыванием римского эквивалента каждой цифры от старшего разряда к младшему.
- Если в десятичном разряде стоит 0, никаких цифр в этом разряде в римском представлении не пишется.
- Наибольшее число, которое может быть представлено в римской системе счисления — это число 3,999 (MMMCMXCIX).

Пусть римская запись представляет собой строку S длины k . Определим произведение римских цифр числа следующим образом: берём каждый символ строки, рассматриваем его как римское число и перемножаем k полученных чисел. Например, для числа XXVI произведение равно $10 \cdot 10 \cdot 5 \cdot 1 = 500$.

Вам дано целое число N . Выведите **наименьшее** и **наибольшее** римские числа, произведение римских цифр которых равно N , или -1 , если таких чисел нет.

Формат входных данных

Первая строка содержит одно целое число Q — количество запросов ($1 \leq Q \leq 10^5$). Каждая из последующих Q строк содержит по одному целому числу N_i ($1 \leq N_i < 10^{25}$).

Формат выходных данных

Для каждого запроса выведите в новой строке наименьшее и наибольшее римские числа, произведение цифр которых равно N , или -1 , если таких чисел нет.

Система оценки

Задача состоит из четырёх подзадач.

- В первой подзадаче $Q \leq 1000$, все N_i не превосходят 10^6 . Эта подзадача оценивается в 20 баллов.
- Во второй подзадаче $Q \leq 1000$. Эта подзадача оценивается в 22 балла.
- В третьей подзадаче все N_i не превосходят 10^6 . Эта подзадача оценивается в 25 баллов.
- В четвёртой подзадаче дополнительных ограничений нет. Эта подзадача оценивается в 33 балла.

Пример

стандартный ввод	стандартный вывод
2	-1
2	IV VIII
5	