Problem A. Aftermath

Input file:	`стандартный ввод`
Output file:	`стандартный вывод`
Time limit:	2 секунды
Memory limit:	512 мегабайт

Однажды, давным-давно, у вас было замечательное целое положительное число n.

Поскольку вам нравится деление, вы быстро нашли все его целые положительные делители.

Как порядочный человек, вы вычислили a — среднее арифметическое делителей n. Удивительно, но это число оказалось целым.

Через некоторое время вы также вычислили h — среднее гармоническое делителей n. Ещё более удивительно, но это число тоже оказалось целым!

К сожалению, память стала вас подводить, и сейчас вы помните a и h, но не помните n. Однако вы помните, что n не превышало 10^{15}.

Ваша муза предложила вернуть старые добрые времена и восстановить любое значение n, удовлетворяющее известной информации.

Input

Первая строка ввода содержит целое положительное число a.

Вторая строка ввода содержит целое положительное число h.

Гарантируется, что существует целое положительное число $n \leq 10^{15}$ такое, что среднее арифметическое делителей n равно a, а среднее гармоническое делителей n равно h.

Output

Выведите любое целое положительное число n, не превышающее 10^{15} и не противоречащее известным данным.

Example

стандартный ввод	стандартный вывод
3 2	6

Note

Среднее арифметическое — это сумма чисел в наборе, делённая на количество чисел в наборе. Например, среднее арифметическое 1, 2, 3 и 6 равно $\frac{1+2+3+6}{4} = 3$.

Среднее гармоническое — это обратная величина к среднему арифметическому обратных величин к числам в наборе. Например, среднее гармоническое 1, 2, 3 и 6 равно $\left(\frac{1^{-1}+2^{-1}+3^{-1}+6^{-1}}{4}\right)^{-1} = 2$.

Таким образом, в первом примере $n = 6$ подходит под все условия, поскольку его делители — это 1, 2, 3 и 6.

Problem B. Believer

Input file:	`стандартный ввод`
Output file:	`стандартный вывод`
Time limit:	2 секунды
Memory limit:	512 мегабайт

Верите ли вы в драконов? Представьте, что один из них будит вас ночью и задаёт следующую задачу:

Будем рассматривать последовательности целых положительных чисел $a = \langle a_1, a_2, \ldots, a_k \rangle$.

Обозначим через $f(a, x)$ количество вхождений числа x в a. Например, $f(\langle 1, 4, 1, 1 \rangle, 1) = 3$.

Обозначим через $c(y)$ количество единиц в двоичной записи числа y. Например, $c(13) = c(1101_2) = 3$.

Пусть также $b(a) = \sum\limits_{i \in a} c(f(a, i))$. Например, $b(\langle 1, 4, 1, 1 \rangle) = c(3) + c(1) = 2 + 1 = 3$.

По заданному значению n найдите максимальное значение $b(a)$ по всем последовательностям целых положительных чисел, для которых $\sum\limits_{i=1}^{k} a_i = n$.

Что вы ответите?

Input

Первая строка ввода содержит целое число t $(1 \le t \le 10^3)$ — количество тестов.

Каждая из следующих t строк содержит целое число n $(1 \le n \le 10^{18})$.

Output

Для каждого теста, в порядке ввода, выведите одно целое число — ответ на задачу.

Example

стандартный ввод	стандартный вывод
2 7 42	3 10

Note

В первом примере одной из возможных последовательностей с $b(a) = 3$ является $a = \langle 1, 4, 1, 1 \rangle$.

Problem C. Chalk Outline (Div. 2 Edition)

Input file:	`стандартный ввод`
Output file:	`стандартный вывод`
Time limit:	2 секунды
Memory limit:	512 мегабайт

Вашей подруге Грейс дали задание. Ей нужно изобразить простой многоугольник. В многоугольнике должно быть n вершин. Он не должен иметь самопересечений или самокасаний. Никакие три последовательные вершины многоугольника не должны лежать на одной прямой. Легко, не правда ли?

Однако есть ещё одно небольшое ограничение.

Диагональ многоугольника — это отрезок прямой, соединяющий две вершины, не являющиеся соседними в многоугольнике. Будем называть диагональ *внутренней*, если каждая точка, лежащая на диагонали (не считая вершин), лежит **строго внутри** многоугольника.

Количество внутренний диагоналей многоугольника должно быть равно k.

Грейс уже три дня безуспешно пытается решить эту задачу. Вас заинтересовало, выполнимо ли данное задание в принципе.

Input

Единственная строка ввода содержит два целых числа n и k ($4 \le n \le 100$; $0 \le k \le \frac{n(n-3)}{2}$).

Output

Если невозможно изобразить многоугольник, удовлетворяющий условиям, выведите одно слово «`No`». В противном случае выведите «`Yes`».

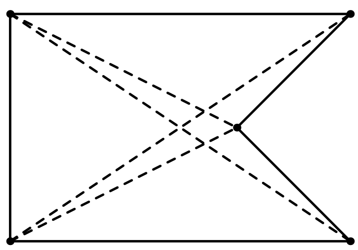
Examples

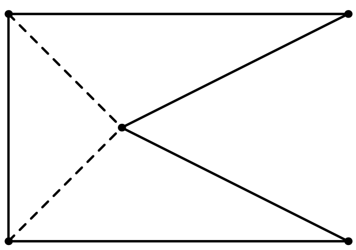
стандартный ввод	стандартный вывод
`5 4`	`Yes`
`5 2`	`Yes`
`4 0`	`No`

Note

Ниже изображены подходящие многоугольники для первого и второго примера, соответственно. Сплошные линии соответствуют рёбрам многоугольника, а пунктирные — внутренним диагоналям.

Любой многоугольник из четырёх вершин имеет хотя бы одну внутреннюю диагональ, поэтому ответ на третий пример — «`No`».





Problem D. Do I Wanna Know? (Div. 2 Edition)

Input file:	`стандартный ввод`
Output file:	`стандартный вывод`
Time limit:	2 секунды
Memory limit:	512 мегабайт

Вы ответственны за проведение ежегодного Арктического Соревнования для Мартышек (ACM). В соревновании участвуют три мартышки с номерами от 1 до 3. Каждые две мартышки участвуют в отдельном контесте с одной задачей друг против друга. Ничьих не бывает.

Вы знаете результаты всех трёх контестов. Эти результаты описаны тремя целыми числами a, b и c. a обозначает результат контеста между мартышками 1 и 2: если $a = 1$, то мартышка 1 обыграла мартышку 2, в противном случае $a = 2$. Аналогично, b равно номеру мартышки, победившей в контесте между мартышками 1 и 3, и c равно номеру мартышки, победившей в контесте между мартышками 2 и 3.

Определите, есть ли мартышка, которая обыграла двух других.

Input

Единственная строка ввода содержит три целых числа a, b и c ($a = 1$ или $a = 2$; $b = 1$ или $b = 3$; $c = 2$ или $c = 3$).

Output

Если есть мартышка, которая обыграла двух других, выведите «`Yes`». В противном случае выведите «`No`».

Examples

стандартный ввод	стандартный вывод
`2 1 2`	`Yes`
`1 3 2`	`No`

Note

В первом примере мартышка 2 обыграла мартышек 1 и 3, поэтому ответ «`Yes`».

Во втором примере мартышка 1 обыграла мартышку 2, мартышка 2 обыграла мартышку 3, а мартышка 3 обыграла мартышку 1. Ни одна из мартышек не обыграла двух других.

Problem E. Exit Song (Div. 2 Edition)

Input file:	`стандартный ввод`
Output file:	`стандартный вывод`
Time limit:	2 секунды
Memory limit:	512 мегабайт

Ваш любимый исполнитель даёт прощальный концерт, и вы просто не можете это пропустить.

Концерт состоится в зале, состоящем из n рядов, пронумерованных от 0 до $n-1$, в каждом из которых есть m мест, последовательно пронумерованных от 0 до $m-1$.

К сожалению, k мест в зале уже недоступны для бронирования. Эти места заданы парами (r_1, s_1), (r_2, s_2), ..., (r_k, s_k). Для каждого i от 1 до k, билет на место s_i в ряду r_i уже продан.

Вы точно пойдёте на концерт, но вы не знаете, присоединятся ли к вам ваши друзья. Вы рассматриваете все варианты купить билеты на несколько (не менее одного) последовательных мест в одном ряду. Сколько таких вариантов у вас есть?

Input

Первая строка ввода содержит три целых числа n, m и k ($1 \le n, m \le 10^5$; $1 \le k \le \min(n \cdot m, 10^5)$) — размеры концертного зала и количество занятых мест, соответственно.

Вторая строка ввода содержит три целых числа r_1, a_r и b_r ($0 \le r_1, a_r, b_r < n$).

Третья строка ввода содержит три целых числа s_1, a_s и b_s ($0 \le s_1, a_s, b_s < m$).

Так как ввод может быть достаточно большим, он закодирован следующим образом: значения r_1 и s_1 даны, а для каждого i от 2 до k значения r_i и s_i могут быть вычислены по следующим формулам:

$$r_i = (r_{i-1} \cdot a_r + b_r) \bmod n;$$
$$s_i = (s_{i-1} \cdot a_s + b_s) \bmod m.$$

Все пары (r_i, s_i) различны.

Output

Выведите одно целое число — количество вариантов купить билеты на несколько последовательных мест в одном ряду.

Examples

стандартный ввод	стандартный вывод
3 4 3 1 2 0 2 1 1	18
22 13 41 7 12 14 5 8 1	1195

Note

В первом примере заняты места $(1, 2)$, $(2, 3)$ и $(1, 0)$. Есть 10 вариантов купить билеты в ряду 0, 2 варианта в ряду 1 и 6 вариантов в ряду 2. Сумма равна $10 + 2 + 6 = 18$.

Problem F. Forever and Always

Input file:	`стандартный ввод`
Output file:	`стандартный вывод`
Time limit:	2 секунды
Memory limit:	512 мегабайт

Рассмотрим абстрактную процедуру голосования. Например, это могло бы быть голосование за лучшую композицию группы «Bullet for My Valentine» в 2019 году.

В голосовании участвуют n человек, и голосовать можно за m различных вариантов. Каждый человек сформировал свой собственный *список предпочтений*, который включает в себя некоторые из вариантов в порядке от более предпочтительного к менее предпочтительному. Обратите внимание, что некоторые варианты могут отсутствовать в списке предпочтений — такие варианты не просто малопредпочтительны, а вообще неприемлемы.

Голосование состоит из *итераций*.

На первой итерации каждый человек голосует за первый вариант в своём списке предпочтений. Количество голосов за каждый вариант подсчитывается и объявляется всем участникам голосования.

На каждой последующей итерации каждый человек намеревается голосовать за тот вариант из своего списка предпочтений, который получил больше голосов на предыдущей итерации. Если таких вариантов несколько, выбирается тот, который идёт раньше в списке предпочтений.

Перед каждой итерацией всем задаётся вопрос, собирается ли кто-нибудь голосовать не за тот же вариант, что и на предыдущей итерации. Если таких людей нет, итерация не проводится, а результаты последней итерации объявляются окончательными результатами голосования. В противном случае голосование проводится и, аналогично первой итерации, количество голосов за каждый вариант вновь подсчитывается и объявляется всем участникам голосования. Обратите внимание, что голоса, данные на предыдущих итерациях, с этого момента игнорируются.

Процедура голосования такого рода выглядит для вас очень громоздко и, что более важно, кажется, что голосование может занять целую вечность! Чтобы доказать свою точку зрения, предъявите значения n, m и списки предпочтений такие, что будет проведено не менее 100 итераций голосования.

Input

Единственная строка ввода содержит целое число p — требуемое количество итераций.

В задаче два теста. В первом тесте $p = 2$. Во втором тесте $p = 100$.

Output

Выведите два целых числа n и m ($1 \le n \le 10^5$; $1 \le m \le 2 \cdot 10^5$) — количество участников голосования и количество вариантов, соответственно. Далее выведите n списков предпочтений.

Описание каждого списка предпочтений должно начинаться с целого числа k_i ($1 \le k_i \le m$) — количества вариантов в списке, за которым должны следовать k_i различных целых чисел $a_{i,j}$ ($1 \le a_{i,j} \le m$) — номера вариантов в списке в порядке от более предпочтительных к менее предпочтительным.

Сумма всех значений k_i не должна превышать $2 \cdot 10^5$.

Example

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

Note

Рассмотрим пример.

На первой итерации все голосуют за первый вариант в своих списках предпочтений. Таким образом, первый человек голосует за вариант 1, второй и четвёртый человек голосуют за вариант 2, а третий человек голосует за вариант 5.

На второй итерации первый человек, видя, что вариант 2 получил больше голосов на первой итерации, изменит свой голос с варианта 1 на вариант 2. Все остальные не изменят свои голоса. В частности, третий человек вновь проголосует за вариант 5, поскольку варианты 5 и 1 получили по одному голосу на первой итерации, но вариант 5 идёт раньше в его списке предпочтений.

Наконец, третья итерация не состоится, поскольку больше никто не хочет менять свой голос. Состоялось две итерации, что удовлетворяет $p = 2$.

Problem G. Gate 21 (Div. 2 Edition)

Input file:	`стандартный ввод`
Output file:	`стандартный вывод`
Time limit:	2 секунды
Memory limit:	512 мегабайт

Вы участвуете в лыжной гонке. Ходит слух, что главным призом будет автограф Сержа Танкяна.

Каждый лыжник должен проехать через n ворот, пронумерованных от 1 до n. Ворота номер i состоят из нескольких равнозначных контрольных пунктов, которые можно считать точками на плоскости, имеющими координаты (i, j) для всех целых j между l_i и r_i, включительно. Требуется пройти через ровно один контрольный пункт каждых ворот в возрастающем порядке номеров ворот.

К сожалению, вы очень плохо умеете поворачивать на лыжах, поэтому хотели бы подготовить для себя маршрут, являющийся прямой линией, проходящей через один контрольный пункт каждых ворот. Сколько вариантов маршрута у вас есть?

Input

Первая строка ввода содержит целое число n ($2 \le n \le 100$).

Каждая из следующих n строк содержит два целых числа l_i и r_i ($1 \le l_i \le r_i \le 100$).

Output

Выведите одно целое число — количество подходящих для вас прямых маршрутов.

Example

стандартный ввод	стандартный вывод
3 1 3 2 3 1 5	6

Note

В примере все подходящие маршруты выглядят следующим образом:

- $(1, 1) \to (2, 2) \to (3, 3)$;
- $(1, 2) \to (2, 2) \to (3, 2)$;
- $(1, 3) \to (2, 3) \to (3, 3)$;
- $(1, 2) \to (2, 3) \to (3, 4)$;
- $(1, 3) \to (2, 2) \to (3, 1)$;
- $(1, 1) \to (2, 3) \to (3, 5)$.

Problem H. Hamilton (Div. 2 Edition)

Input file:	`стандартный ввод`
Output file:	`стандартный вывод`
Time limit:	2 секунды
Memory limit:	512 мегабайт

Лин-Мануэль перемещается по полоске из n ячеек, последовательно пронумерованных от 1 до n.

Он начинает в ячейке a и хочет финишировать в ячейке b. В процессе он хочет посетить каждую ячейку ровно один раз.

Из каждой ячейки x Лин-Мануэль может перейти в соседнюю ячейку слева — ячейку $x - 1$ (если она существует) или справа — ячейку $x + 1$ (если она существует).

Он также может позвать свою подругу, ведьму Миранду, которая дарует ему магическую способность. С этой способностью он сможет перелететь ровно один раз из своей текущей ячейки x в любую ячейку y такую, что наибольший общий делитель x и y равен 1.

Лин-Мануэль не хочет слишком сильно напрягать Миранду. Он хотел бы достичь своей цели, используя как можно меньше перелётов.

Помогите ему и найдите наименьшее необходимое количество перелётов, а также оптимальную последовательность посещения ячеек.

Input

Единственная строка ввода содержит три целых числа n, a и b ($2 \le n \le 9$; $1 \le a, b \le n$; $a \ne b$) — количество ячеек в полоске, номер начальной ячейки и номер конечной ячейки, соответственно.

Output

Если невозможно достичь цели с любым количеством перелётов, выведите единственное число -1.

В противном случае выведите наименьшее количество перелётов, необходимое, чтобы переместиться из ячейки a в ячейку b, посещая все ячейки ровно один раз. Далее выведите n различных целых чисел $c_1, c_2, \ldots, c_n$ ($1 \le c_i \le n$) — номера ячеек в порядке посещения, описывающие любой допустимый маршрут, требующий наименьшего возможного количества перелётов. В частности, должно быть верно, что $c_1 = a$ и $c_n = b$.

Examples

стандартный ввод	стандартный вывод
`5 1 5`	`0` `1 2 3 4 5`
`6 4 5`	`1` `4 3 2 1 6 5`
`7 5 3`	`2` `5 4 7 6 1 2 3`
`4 1 3`	`-1`

Problem I. I've Got Friends (Div. 2 Edition)

Input file:	`стандартный ввод`
Output file:	`стандартный вывод`
Time limit:	2 секунды
Memory limit:	512 мегабайт

Британские учёные выяснили, что дружба очень предсказуема. Они утверждают, что два человека могут стать друзьями тогда и только тогда, когда у них есть хотя бы один общий любимый вид еды среди двух любимых видов еды каждого из них.

Известный оркестр из Манчестера был выбран для проведения научного эксперимента. Каждого музыканта попросили выбрать ровно два различных вида еды, которые ему нравятся больше всего.

В начальном отчёте учёные опубликовали список двух любимых видов еды каждого из музыкантов.

Вам интересно, сколько пар музыкантов могут стать друзьями в соответствии с научной теорией.

Input

Первая строка ввода содержит целое число n ($2 \le n \le 100$) — количество музыкантов в экспериментальной группе. Музыканты пронумерованы от 1 до n.

Каждая из следующих n строк содержит два целых числа $f_{i,0}$ и $f_{i,1}$ ($1 \le f_{i,j} \le 10^4$; $f_{i,0} \ne f_{i,1}$) — идентификаторы любимых видов еды i-го музыканта. Различные числа соответствуют различным видам еды.

Output

Выведите одно целое число — количество пар музыкантов, которые могли бы стать друзьями в соответствии с научной теорией.

Example

стандартный ввод	стандартный вывод
7 58 42 101 202 42 58 303 202 78 7788 202 404 404 101	6

Note

В первом примере пары музыкантов, которые могли бы стать друзьями — $(1,3)$, $(2,4)$, $(2,6)$, $(2,7)$, $(4,6)$ и $(6,7)$.

Problem J. Joke

Input file:	`стандартный ввод`
Output file:	`стандартный вывод`
Time limit:	2 секунды
Memory limit:	512 мегабайт

Карточная игра, которую часто называют «Дурак», довольно популярна в России. Мы опишем модификацию этой игры для двух игроков и колоды из 54 карт (52 обычных карт и двух джокеров, чёрного и красного).

Будем считать, что все пиковые и трефовые карты — чёрного цвета, а все бубновые и червовые карты — красного цвета. У джокеров нет ни достоинства, ни масти, только цвет. Одна из мастей объявляется *козырной*, карты этой масти называются *козырями*.

В начале игры у обоих игроков по шесть карт. Оставшиеся 42 карты в некотором порядке составляют *колоду*.

Игра состоит из раундов. Перед раундом каждый игрок имеет одну или несколько карт, один из игроков *подкидывает*, другой — *отбивается*. Подкидывающий игрок начинает с того, что выкладывает одну или несколько карт одинакового достоинства на стол. Подкидывающий игрок не имеет права выкладывать джокеров. Количество выложенных карт не должно превышать количества карт у отбивающегося игрока. Отбивающийся игрок, в свою очередь, *покрывает* все карты некоторыми из своих карт, выкладывая их на стол поверх непокрытых карт. Карта может покрыть другую карту, если верно хотя бы одно из следующих утверждений:

- покрывающая карта — той же масти и выше достоинством (порядок достоинств карт: 2, 3, 4, 5, 6, 7, 8, 9, 10, J, Q, K, A, где 2 — самое низкое достоинство, A — самое высокое);
- покрывающая карта — козырь, а покрываемая — нет;
- покрывающая карта — джокер, цвет которого совпадает с цветом покрываемой карты;
- покрывающая карта — джокер, цвет которого совпадает с цветом козырной масти.

После того, как все карты на столе покрыты, подкидывающий игрок может *подкинуть* ещё одну или несколько карт. Аналогично, подкидывающий игрок не имеет права подкидывать джокеров. Достоинства подкидываемых карт должны присутствовать среди достоинств карт, уже лежащих на столе в этот момент. Далее подкинутые карты должны быть покрыты отбивающимся игроком, после чего подкидывающий игрок может подкинуть ещё карт, и так далее. Подкидывающий игрок не может подкинуть больше карт, чем есть у отбивающегося игрока на данный момент.

Раунд заканчивается, когда либо отбивающийся игрок не может или не хочет покрывать все непокрытые карты на столе, либо подкидывающий игрок не может или не хочет подкинуть дополнительных карт. В первом случае, когда отбивающийся игрок объявляет, что не хочет покрывать все непокрытые карты на столе, подкидывающий игрок имеет возможность подкинуть ещё карт, не являющихся джокерами. Достоинства подкидываемых карт должны присутствовать среди достоинств карт, уже лежащих на столе. Количество непокрытых карт на столе не может превышать количества карт у отбивающегося игрока на данный момент. После этого отбивающийся игрок проигрывает раунд и забирает все карты со стола, добавляя эти карты к своим. Подкидывающий игрок остаётся в роли подкидывающего на следующий раунд.

Во втором случае, когда все карты на столе покрыты, а подкидывающий игрок не может или не хочет подкинуть ещё карт, отбивающийся игрок выигрывает раунд, а карты на столе уходят в отбой (убираются из игры). Игроки меняются ролями на следующий раунд: отбивающийся игрок становится подкидывающим, и наоборот.

Между раундами, если подкидывающий игрок предыдущего раунда имеет менее шести карт, он добирает дополнительные карты с верха колоды по одной до тех пор, пока у него не окажется ровно шесть карт или не кончится колода. После этого, аналогично, если отбивающийся игрок предыдущего раунда имеет менее шести карт, он добирает дополнительные карты с верха колоды по одной до тех пор, пока у него не окажется ровно шесть карт или не кончится колода.

Если перед стартом раунда у одного из игроков нет карт, а у другого есть хотя бы одна, то игрок без карт объявляется победителем. Если у обоих игроков нет карт, игра заканчивается ничьей. Если у обоих игроков есть хотя бы одна карта, но все карты подкидывающего игрока предстоящего раунда — джокеры, то этот игрок не может совершить ход и выложить хотя бы одну карту, игра заканчивается, а его соперник — отбивающийся игрок предстоящего раунда — объявляется победителем.

Два игрока, Иоганн и Себастьян, собираются сыграть игру по описанным выше правилам. Иоганн будет подкидывающим игроком в первом раунде.

По заданным козырной масти, картам у игроков в начале игры и порядку оставшихся карт в колоде определите, кто выиграет игру, если оба игрока будут действовать оптимально. У обоих игроков есть полная информация о картах в игре и порядке карт в колоде.

Input

Первая строка ввода содержит целое число t ($1 \le t \le 10^4$) — количество тестов.

Описание каждого теста состоит из четырёх строк. Первая строка содержит шесть описаний карт — карты Иоганна. Вторая строка содержит шесть описаний карт — карты Себастьяна. Третья строка содержит 42 описания карт — карты в колоде в порядке от верхней к нижней. Четвёртая строка содержит один символ — козырную масть.

Каждая карта, кроме джокеров, описывается своим достоинством (‘`2`’…‘`9`’, ‘`T`’ — десятка, ‘`J`’ — валет, ‘`Q`’ — дама, ‘`K`’ — король, ‘`A`’ — туз), за которым следует масть (‘`S`’ — пики, ‘`C`’ — трефы, ‘`D`’ — бубны, ‘`H`’ — червы). Красный джокер описывается двумя символами “`RJ`”. Чёрный джокер описывается двумя символами “`BJ`”.

Все 54 карты в каждом тесте различны.

Output

Для каждого теста выведите одну строку, содержащую имя победителя игры («`Johann`» или «`Sebastian`») или «`Bach`», если игра закончится вничью.

Example

стандартный ввод	стандартный вывод
2 TC QD 2S TH 4S 3C AS RJ AC 7D 6C BJ 3D 4C 8C AD TD TS 7H JS KD 4H QC 6H 9D 7C 9H JC AH 5H 6S QH KS 5S 5D 3H JD JH 8H QS 2H 4D 5C 9S KH 6D 9C 8D 8S KC 7S 3S 2D 2C S TC 8S JS JD 5C 9C QS 8C 3H 4D 4H 2D QH 7S 7H 3C 2H 7C TD 9H 8D AH 7D QC JH 5D AS 5H 3D JC 2S 6D AC 9D 4C 6S KD 8H 6C 4S RJ KH 3S TS KC KS 5S QD 9S BJ 6H TH AD 2C D	Johann Sebastian

Note

Третья строка обоих тестовых примеров отображается в несколько строк. В официальных тестах все 42 карты колоды описаны в одной строке.

Problem K. Kids Aren't Alright (Div. 2 Edition)

Input file:	`стандартный ввод`
Output file:	`стандартный вывод`
Time limit:	2 секунды
Memory limit:	512 мегабайт

Как ни странно, сумасшедший парень на другом конце телефона утверждает, что украл вашего драгоценного ребёнка. Вы не слишком-то ему верите, так как все ваши дети (возможно, ни одного) играют перед вами прямо сейчас, живы и здоровы. Тем не менее, ситуация кажется вам любопытной, поэтому вы спрашиваете преступника, чего он хочет в качестве выкупа.

Как ни скучно, но похититель просит денег. Всего лишь денег. Вы уже собирались было разочарованно бросить трубку, как нечто странное привлекло ваше внимание. Ваш собеседник не называет вам точную сумму, которую он хочет. Вместо этого он предлагает вам загадку.

Как ни глупо, но загадка звучит так:

«*Сколько непустых множеств целых положительных чисел существует, таких, что наибольший общий делитель чисел в множестве равен 1, а наименьшее общее кратное равно m?*».

Похититель сообщает вам, что ответ на эту загадку, взятый по модулю 998244353, и есть точная сумма выкупа, которую он хочет за возврат вашего воображаемого отпрыска.

Теперь вас заинтересовали тарифы на рынке похищения детей, поскольку вы уже довольно давно не занимались подобными делами. Не то чтобы вы собираетесь платить вору хотя бы копейку.

Input

Единственная строка ввода содержит целое число m ($1 \le m \le 10^5$).

Output

Выведите одно целое число — сумму денег, которую от вас требуют.

Examples

стандартный ввод	стандартный вывод
6	7
100	322

Note

В первом примере все подходящие множества — $\{1, 6\}$, $\{2, 3\}$, $\{1, 2, 3\}$, $\{1, 2, 6\}$, $\{1, 3, 6\}$, $\{2, 3, 6\}$ и $\{1, 2, 3, 6\}$.