

Лабораторная работа №6

В рамках данной лабораторной работы необходимо решить предложенные задачи на языке программирования высокого уровня из предложенного перечня. Варианты задач находятся в отдельном файле. В качестве результата выполнения практической работы необходимо приложить **архив с файлами решенных задач, а также отчет о выполнении работы**. Файлы решенных задач представляют собой файлы с исходным кодом с установленным расширением. При решении задач необходимо писать свои собственные структуры данных и методы сортировки, использование встроенных не допускается (кроме массивов).

Отчет о выполнении практической работы должен содержать:

1. Титульный лист

2. Содержание

3. 4 параграфа, в которых раскрыто решение каждой задачи. По каждой задаче необходимо представить следующую информацию:

3.1. *Условие задачи*. Берется из файла.

3.2. *Ход решения задачи*. Студент описывает логику решения данной задачи. Какие алгоритмы и для чего использованы, как построена программа. Данная часть является описательной. Здесь следует говорить именно о построении алгоритма, опуская процессы ввода и вывода данных (если это не является основной сутью алгоритма).

3.3. *Листинг программы с комментариями*. Копируется весь программный код.

3.4. *Тестирование программы*. Если для задачи предусмотрены автотесты, приложить скриншот их прохождения. Если нет: Составляется таблица, содержащая следующие поля: номер теста, входные данные, результат выполнения программы, корректность, время выполнения (мс), затраченная память (Мб). Составляется не менее 10-ти тестовых наборов данных согласно условию задачи. Тестовые наборы входных данных студент составляет самостоятельно. В обязательном порядке программа тестируется на граничных наборах входных данных (например, если N варьируется от 0 до 10^9 , то обязательно рассмотреть решение задачи при $N=0$ и при N близком к 10^9). Если написанная программа не позволяет решить задачу при граничных входных данных, все равно включить в тест и в качестве результата написать "Не решено". В столбце "входные данные" данные впечатываются вручную, в столбце "результат..." представляется скриншот выполнения программы (если не влезает на одну страницу, делать несколько скриншотов).

Каждая лабораторная работа защищается на занятии преподавателю.

Задачи по теме «Жадные алгоритмы»

Задача 1. Сумма элементов подмассива

Ограничение по времени: 1 секунда

Ограничение по памяти: 256 мегабайт

Имеется массив V целых чисел, состоящий из $1 \leq N \leq 10^8$ элементов, $-2 \times 10^9 \leq V_i \leq 2 \times 10^9$.

Подмассивом называют непрерывное подмножество элементов массива, возможно, включающее в себя и полный массив.

Требуется найти наибольшую из возможных сумм всех подмассивов.

Формат входных данных:

N

V_1

V_2

...

V_N

Формат выходных данных:

MaximalSubarraysSum

Примеры:

Стандартный ввод	Стандартный вывод
10 -4 4 3 3 -4 1 2 1 -4 0	10

Задача 2. Маршрутка

Ограничение по времени: 1 секунда

Ограничение по памяти: 64 мегабайта

Вступление

Водитель маршрутки Сергей Александрович Жадных прославился своей феноменальной жадностью повсеместно. Он сам неоднократно заявлял, что за лишнюю копейку готов задушить родного брата и продать всех друзей. К сожалению, проверить эти слова не представлялось возможным, поскольку никакого брата, а также друзей, дома и семьи у Сергея Александровича не было. И что ещё хуже, денег у него тоже не было. Единственным достоянием г-на Жадных являлась старая маршрутка, на которой он и колесил по городу, подвозя редких пассажиров и время от времени осматривая краем глаза тротуары в поисках мелких монет...

В один из дней небеса сжалились на Сергеем Александровичем и решили прекратить его бесполезное существование в этом жестоком мире. С сей благой целью на голову ничего не подозревавшего г-на Жадных, вышедшего из маршрутки на призывный блеск 5-рублёвой монеты, свалился топор. Мечты о выгодной продаже бутылки мигом вылетели у него из головы, поскольку их место занял топор. В переносном смысле этого слова. Орудие небес не смогло пробить окостеневший череп г-на Жадных, однако, как выяснилось позднее, придало ему несколько весьма полезных свойств.

Вскоре после продажи топора Сергей Александрович обнаружил, что способен предвидеть будущее. Какие возможности, какие перспективы открылись незадачливому водителю маршрутки! Кто мы такие и куда идём? Чего бояться и на что надеяться? Ответы на эти вопросы г-на Жадных совершенно не интересовали. А вот на то, чтобы с помощью новых умений попытаться заработать немного денег, выполняя привычную работу, сообразительности у Сергея Александровича хватило.

Задача

Ежедневно маршрутка совершает один рейс от первой до N-й остановки. В маршрутке M мест для пассажиров. Вечером, просчитав линии вероятностей, г-н Жадных (между прочим, потенциальный Тёмный Иной шестого уровня) выяснил, что завтра на остановках маршрутку будут поджидать K человек. Для каждого человека были определены номер остановки $S[i]$, на которой он желает сесть в маршрутку, и номер остановки $F[i]$, на которой он собирается выйти. В соответствии с ценовой политикой Сергея Александровича, каждый пассажир должен заплатить P рублей за билет независимо от количества остановок. Более того, притормозив на остановке, г-н Жадных может выбирать, кого из желающих посадить в маршрутку, а кого нет. Ставя перед собой задачу максимизации прибыли, Сергей Александрович вполне разумно решил определить, каких именно людей нужно сажать в маршрутку. К сожалению, для этого его сил оказалось недостаточно. А Ваших?

Формат входных данных:

Первая строка содержит целые числа N ($2 \leq N \leq 100000$), M ($1 \leq M \leq 1000$), K ($0 \leq K \leq 50000$) и P ($1 \leq P \leq 10000$). Каждая из следующих K строк содержит целые числа $S[i]$ и $F[i]$ ($1 \leq S[i] < F[i] \leq N$) для соответствующего человека.

Формат выходных данных:

В первую строку вывести максимальную прибыль. Во вторую строку вывести через пробел и в любом порядке номера людей, которых следует сажать в маршрутку для получения этой прибыли. Если задача имеет несколько решений, то вывести любое из них.

Примеры:

Стандартный ввод	Стандартный вывод
6 2 6 9 1 4 2 6 1 5 2 3 4 6 3 6	36 1 5 6 4

Задача 3. Танец точек.

Ограничение по времени: 1 секунда

Ограничение по памяти: 256 мегабайт

На прямой располагается $1 \leq N \leq 10000$ точек с целочисленными координатами $-10^9 \leq V_i \leq 10^9$. Каждой из точек разрешается сделать ровно одно движение (танцевальное па) в любом направлении на расстояние не больше $0 \leq L \leq 10^8$ и остановиться на другой позиции. Какое минимальное количество точек может остаться на прямой после окончания танца (все точки после танца, оказывающиеся на одной позиции, сливаются в одну)?

Формат входных данных:

L N

V1 V2 ... VN

Формат выходных данных:

MinimalNumberOfPoints

Примеры:

Стандартный ввод	Стандартный вывод
10 5 30 3 14 19 21	2

Задача 4. Ровно М простых.

Ограничение по времени: 2 секунды

Ограничение по памяти: 24 мегабайта

Требуется найти такое наименьшее натуральное число $2 \leq K \leq 2 \times 10^7$, что, начиная с этого числа, среди N натуральных чисел имеется ровно M простых.

Если такого числа не существует или оно больше 2×10^7 , вывести -1.

Формат входных данных:

M N

Формат выходных данных:

K или -1

Примеры:

Стандартный ввод	Стандартный вывод
4 10	3
3 15	14

Задача 5. Периодическая дробь.

Ограничение по времени: 1 секунда

Ограничение по памяти: 256 мегабайт

Выведите десятичное представление рациональной правильной дроби. Если в представлении присутствует период, то нужно вывести первое его вхождение в круглых скобках.

Формат входных данных:

Два целых числа, введенных через пробел: $1 \leq N < M \leq 150000000$

Формат выходных данных:

Десятичное представление числа N/M

Примеры:

Стандартный ввод	Стандартный вывод
------------------	-------------------

6 70	0.0(857142)
17 250	0.068

Задача 6. Валютные махинации

Ограничение по времени: 1 секунда

Ограничение по памяти: 16 мегабайт

Петя, изучая, как меняется курс рубля по отношению к доллару и евро, вывел закон, по которому происходят эти изменения (или думает, что вывел :)). По этому закону Петя рассчитал, каков будет курс рубля по отношению к доллару и евро в ближайшие N дней.

У Пети есть 100 рублей. В каждый из дней он может обменивать валюты друг на друга по текущему курсу без ограничения количества (при этом курс доллара по отношению к евро соответствует величине, которую можно получить, обменяв доллар на рубли, а потом эти рубли — на евро). Поскольку Петя будет оперировать не с наличной валютой, а со счетом в банке, то он может совершать операции обмена с любым (в том числе и нецелым) количеством единиц любой валюты.

Напишите программу, которая вычисляет, какое наибольшее количество рублей сможет получить Петя к исходу N -го дня.

Законы изменения курсов устроены так, что в течение указанного периода рублевый эквивалент той суммы, которая может оказаться у Пети, не превысит 10^8 рублей.

Формат входных данных:

Первая строка входного файла INPUT.TXT содержит одно число N ($1 \leq N \leq 5000$). В каждой из следующих N строк записано по 2 числа, вычисленных по Петиним законам для соответствующего дня — сколько рублей будет стоить 1 доллар, и сколько рублей будет стоить 1 евро. Все эти значения не меньше 0.01 и не больше 10000. Значения заданы точно и выражаются вещественными числами не более, чем с двумя знаками после десятичной точки.

Формат выходных данных:

В выходной файл OUTPUT.TXT выведите искомую величину с двумя знаками после десятичной точки.

Примеры:

INPUT.TXT	OUTPUT.TXT
4	4000.00
1 10	
10 5.53	
5.53 1.25	
6 5	

Источник [здесь](#).

Задача 7. Научная конференция

Ограничение по времени: 1 секунда

Ограничение по памяти: 64 мегабайта

Работа научной конференции обычно разделена на несколько одновременно проходящих секций. Например, может быть секция параллельных вычислений, секция визуализации, секция сжатия данных и так далее.

Очевидно, одновременная работа нескольких секций необходима, чтобы уменьшить время научной программы конференции и иметь больше времени на банкет,

чаепитие и неофициальные обсуждения. Однако интересные доклады могут проходить одновременно в разных секциях.

Участник записал расписание всех докладов, интересных ему. Он просит вас определить максимальное количество докладов, которые он сможет посетить.

Формат входных данных

Первая строка содержит количество $1 \leq N \leq 100\,000$ интересных докладов. Каждая из следующих N строк содержит два целых числа T_s и T_e , разделённых пробелом ($1 \leq T_s < T_e \leq 30\,000$). Эти числа — время начала и конца соответствующего доклада. Время задано в минутах от начала конференции.

Формат выходных данных

Выведите максимальное количество докладов, которые участник может посетить. Участник не может посетить два доклада, идущих одновременно, и любые два доклада, которые он посещает, должны быть разделены хотя бы одной минутой. Например, если доклад кончается в 15, следующий доклад, который может быть посещён, должен начинаться в 16 или позже.

Примеры

стандартный ввод	стандартный вывод
5 3 4 1 5 6 7 4 5 1 3	3

Источник: [здесь](#).

Задача 8. Интересная игра с числами

Ограничение по времени: 1 секунда

Ограничение по памяти: 16 мегабайт

Рассмотрим следующую интересную игру для двух игроков. Для этой игры необходима таблица из 2-х строк и N столбцов, в клетках которой записаны натуральные числа, следующего вида:

A_1	A_2	A_3	...	A_N
B_1	B_2	B_3	...	B_N

Игроки делают ходы по очереди. Начинает игру 1-й игрок.

За один ход 1-й игрок выполняет следующие два действия:

- выбирает произвольный столбец (к примеру, j -й), который еще ни разу не был выбран одним из игроков на предыдущих ходах;
- прибавляет к своим очкам число A_j .

За один ход 2-й игрок выполняет следующие два действия:

- выбирает произвольный столбец (к примеру, j -й), который еще ни разу не был выбран одним из игроков на предыдущих ходах;
- прибавляет к своим очкам число B_j .

Игра заканчивается, когда какой-либо из игроков не сможет сделать ход (по той причине, что все столбцы уже были выбраны). Изначально, у каждого из игроков есть 0 очков.

После того, как игра закончилась, происходит взаиморасчет между игроками. К примеру, 1-й игрок набрал S_1 очков, а 2-й игрок - S_2 очков. В случае, когда $S_1 > S_2$, 2-й игрок отдает 1-му игроку $S_1 - S_2$ УДЕ (условных денежных единиц). В противном случае, 1-й игрок отдает 2-му игроку $S_2 - S_1$ УДЕ. С этих позиций, целью 1-го игрока является максимизация величины $S_1 - S_2$, а целью 2-го игрока - максимизация $S_2 - S_1$.

Назовем стоимостью игры величину $S_1 - S_2$ при оптимальной игре обоих игроков. Напишите программу, которая определяет стоимость игры.

Формат входных данных:

В первой строке входного файла INPUT.TXT записано натуральное число N - количество столбцов в таблице ($1 \leq N \leq 300000$). Следующие N строк описывают числа в столбцах таблицы. i -я из этих строк содержит два натуральных числа A_i и B_i , разделенные одним пробелом ($1 \leq A_i, B_i \leq 3000$).

Формат выходных данных:

В выходной файл OUTPUT.TXT выведите одно целое число - стоимость игры.

Примеры:

INPUT.TXT	OUTPUT.TXT
1 1 1	1
2 1 1 1 1	0
3 1 2 3 4 5 6	2

Источник: [здесь](#).

Задача 9. Индикатор

Ограничение по времени: 1 секунда

Ограничение по памяти: 16 мегабайт

Недавно Вася приобрел настольный калькулятор с жидкокристаллическим индикатором. Этот индикатор отображает N цифр с помощью N одинаковых элементов.



Отметим, что каждый элемент содержит семь полосок, каждая из которых может быть либо белой, либо черной. В частности, при отображении цифры «1» черными являются две полоски.

Вася — очень любознательный мальчик, поэтому он хочет узнать, какое максимальное и минимальное N -значное число могут быть отображены на индикаторе его нового калькулятора так, чтобы черными были ровно K полосок.

Напишите программу, которая найдет ответ на Васин вопрос. Учитывайте при этом, что числа не могут содержать ведущие нули.

Формат входных данных:

Входной файл INPUT.TXT содержит два натуральных числа N и K ($1 \leq N \leq 100$, $1 \leq K \leq 700$).

Формат выходных данных:

В первой строке выходного файла OUTPUT.TXT выведите минимальное число, во второй строке выходного файла выведите максимальное число. Если указанным образом не может быть представлено ни одно число, выходной файл должен содержать одну строку NO SOLUTION.

Примеры:

INPUT.TXT	OUTPUT.TXT
5 15	10117 97111
10 1	NO SOLUTION

Источник: [здесь](#).

Задача 10. Коррозия металла

Ограничение по времени: 1 секунда

Ограничение по памяти: 16 мегабайт

Для хранения двух агрессивных жидкостей A и B используется емкость с многослойной перегородкой, которая изготавливается из имеющихся N листов. Для каждого листа i ($i = 1, \dots, N$) известно время его растворения жидкостью A — a_i и жидкостью B — b_i . Растворение перегородки каждой из жидкостей происходит последовательно лист за листом, с постоянной скоростью по толщине листа.

Требуется написать программу проектирования такой перегородки, время растворения которой было бы максимальным.

Формат входных данных:

В первой строке входного файла INPUT.TXT записано число N ($1 \leq N \leq 256$). В каждой из последующих N строк содержатся два положительных вещественных числа a_i и b_i , разделенные пробелом (числа не превышают 10^6 и состоят не более чем из 11 значащих цифр).

Формат выходных данных:

В первую строку выходного файла OUTPUT.TXT записать время растворения перегородки с точностью, не меньшей 10^{-3} . В следующую строку файла записать номера листов в порядке их расположения от жидкости A к жидкости B , разделяя числа пробелами.

Примеры:

INPUT.TXT	OUTPUT.TXT
4 1 2 1 2 0.5 1.5 7 3.5	6.000 4 2 1 3

ИСТОЧНИК: [ЗДЕСЬ](#).

Задачи по теме «Динамическое программирование»

Задача 11. Расстояние по Левенштейну

Ограничение по времени: 0.5 секунд (Python – 2.3 с.)

Ограничение по памяти: 64 Мб

Расстоянием Левенштейна между двумя строками s и t называется количества атомарных изменений, с помощью которых можно одну строку превратить в другую. Под атомарными изменениями подразумеваются: удаление одного символа, вставка одного символа, замена одного символа на другой.

Найди расстояние Левенштейна для предложенной пары строк.

Выведите единственное число – расстояние между строками.

Формат ввода данных:

В первой строке дана строка s , во второй – строка t . Длины обеих строк не превосходит 1000. Строки состоят из маленьких латинских букв.

Примеры:

Стандартный ввод	Стандартный вывод
abacaba abaabc	2
innokentiу innokkentia	3
г х	1

Задача 12. Одинаковые суммы

Ограничение по времени: 0.2 с. (Python – 1.5 с.)

Ограничение по памяти: 8 Мб (Java, C# – 64 Мб)

На Алгосах устроили турнир по настольному теннису. Гоша выиграл n партий, получив при этом некоторое количество очков за каждую из них.

Гоша стало интересно, можно ли разбить все заработанные им во время турнира очки на две части так, чтобы сумма в них была одинаковой.

Формат ввода данных:

В первой строке записано целое число n ($0 \leq n \leq 300$) – количество выигранных партий. Во второй строке через пробел записано n целых неотрицательных чисел, каждое из которых не превосходит 300 – заработанные в партиях очки.

Формат вывода данных:

Нужно вывести *True*, если произвести такое разбиение возможно, иначе – *False*

Пример:

Стандартный ввод	Стандартный вывод
4 1 5 7 1	True
3 2 10 9	False

Задача 13. Игра в фишки

Ограничение по времени: 1 секунда

Ограничение по памяти: 16 мегабайта

На столе лежит куча из $1 \leq N \leq 10^6$ фишек. Игроки First и Second ходят строго по очереди, первый ход за игроком First. Каждый ход игрок может взять из кучи любое количество фишек, не превосходящее целой части квадратного корня из оставшегося на столе количества фишек. Например, при 28 фишках на столе он может взять от одной до пяти фишек. Игра заканчивается, когда на столе не остается ни одной фишки и победителем объявляется тот, кто совершил последний ход.

Требуется вывести имя игрока, который побеждает при обоюдной лучшей игре.

Формат входных данных

N

Формат выходных данных

First

или

Second

Пример

Стандартный ввод	Стандартный вывод
5	First
10	Second

Замечание

Если на доске 5 фишек, то первый игрок может взять или одну, или две фишки. Если он возьмет одну фишку, то второй — две и наоборот. Теперь перед первым игроком лежит куча в 2 фишки и на его единственный возможный ход — взятие одной фишки — второй забирает последнюю и выигрывает.

Задача 14. Путешествие продавца

Ограничение по времени: 1 секунда

Ограничение по памяти: 64 мегабайта

Как обычно, кому-то надо продать что-то во многих городах. Имеются города, представленные как M множеств (столбцов) по N городов (строк) в каждом.

Продавец должен посетить ровно по одному городу из каждого множества, затратив на это как можно меньшую сумму денег. Он должен посетить сначала город из первого множества, затем из второго и так далее, строго по порядку. Он может выбирать начало своего путешествия. Число, которое находится в i -й строке и j -м столбце означает стоимость перемещения из предыдущего места в этот город. Однако, имеется ограничение на перемещения: он может перемещаться из города в i -й строке только в города следующего столбца, находящиеся в одной из строк $i-1$, i , $i+1$, если такие строки существуют.

Иногда, чтобы заставить посетить продавца какой-то город, ему доплачивают, то есть, стоимость перемещения может быть отрицательной.

Требуется определить наименьшую стоимость маршрута и сам маршрут.

Формат входных данных

N M

C11 C12 ... C1M

C21 C22 ... C2M

... ..

CN1 CN2 ... CNM

$3 \leq N \leq 150$

$$3 \leq M \leq 1000$$

$$-1000 \leq C_{ij} \leq 1000$$

Формат выходных данных

Первая строка — список через пробел номеров строк (начиная с 1) из M посещенных городов.

Вторая строка — общая стоимость поездки.

Если имеется несколько маршрутов с одной стоимостью, требуется вывести маршрут, наименьший в лексикографическом порядке.

Начинать и заканчивать маршрут можно в любой строке.

Примеры

Стандартный ввод	Стандартный вывод
5 4 1 7 4 3 5 1 6 7 4 1 9 2 7 3 7 5 8 2 4 1	1 2 1 1 9
5 6 3 4 6 2 8 6 6 1 8 2 7 4 5 9 3 9 9 5 8 4 1 3 9 6 3 7 2 8 6 4	1 2 3 2 2 2 20

Задача 15. Наибольшая общая подстрока

Ограничение по времени: 2 секунды

Ограничение по памяти: 64 мегабайта

Во входном файле находятся две строки, длиной до 30000 символов, состоящих из цифр и прописных и строчных букв латинского алфавита, каждая в отдельной строке файла.

Необходимо найти общую подстроку наибольшей длины. Если таких подстрок несколько, то следует вывести ту из них, которая лексикографически меньше.

Обратите внимание, что в приведенном примере имеется две подстроки длины 4 — rash и abra. Несмотря на то, что первая встречается раньше, ответом будет вторая, так как она лексикографически меньше.

Пример

стандартный ввод	стандартный вывод
ubrashvabracadabra calamburashabratha	abra

Задача 16. Счастливые билеты

Ограничение по времени: 2 секунды

Ограничение по памяти: 64 мегабайта

Билет состоит из четного числа N цифр в M -ричной системе счисления. Счастливым билетом называется билет, сумма первой половины цифр которого равна сумме второй половины цифр.

Найти количество счастливых билетов. Учтите: их число может быть велико.

Формат входных данных

N M

$2 \leq N \leq 150$

$N \bmod 2 = 0$

$2 \leq M \leq 26$

Формат выходных данных

Количество счастливых билетов

Примеры

стандартный ввод	стандартный вывод
6 10	55252
28 12	35806106077437501422929813320

Задача 17. Самая тяжелая подтаблица

Ограничение по времени: 3 секунды

Ограничение по памяти: 256 мегабайта

В каждой ячейке прямоугольной таблица размером $N \times M$ состоит из чисел от -10^9 до 10^9 .

Назовем подтаблицей любую часть таблицы, включая целую, образующую прямоугольник, а ее весом — сумму всех ее чисел.

Найти вес самой тяжелой из всех возможных подтаблиц, которые можно построить на основе оригинальной.

Формат входных данных

N M

C_{11} C_{12} ... C_{1M}

C_{21} C_{22} ... C_{2M}

...

C_{N1} C_{N2} ... C_{NM}

$5 \leq N, M \leq 500$

Формат выходных данных

MaximalPossibleWeight

Примеры

стандартный ввод	стандартный вывод
5 5 1 0 7 -8 2 2 7 -5 3 1 6 -8 4 2 1 -7 3 1 -2 1 2 7 4 0 -50	24
5 5	27

10 -1 -1 7 -3 -6 -6 5 7 -6 8 -2 1 5 6 -1 -2 -3 -8 1 -9 -9 5 6 -1	
--	--

Задача 18. К-ичные числа

Ограничение по времени: 1 секунда

Ограничение по памяти: 64 мегабайта

Рассмотрим N -значные числа в системе счисления с основанием K . Будем считать число *правильным*, если его K -ичная запись не содержит двух подряд идущих нулей. Например:

- 1010230 — правильное 7-значное число;
- 1000198 не является правильным числом;
- 0001235 — не 7-значное, а 4-значное число.

Даны числа N и K , вычислите количество правильных K -ичных чисел, состоящих из N цифр.

Ограничения: $2 \leq K \leq 10$; $N \geq 2$; $N + K \leq 18$.

Формат входных данных

Числа N и K в десятичной записи, разделенные переводом строки.

Формат выходных данных

Искомое количество в десятичной записи.

Примеры

стандартный ввод	стандартный вывод
2 10	90

Источник: [здесь](#).

Задача 19. Дебютный альбом

Ограничение по времени: 2 секунды

Ограничение по памяти: 64 мегабайта

Поп-группа «Розовый слон» приступила к записи своего дебютного альбома. Правда, у группы пока всего две песни: «Моя любовь» и «Я скучаю по тебе», но зато на каждую сделано огромное количество ремиксов.

Продюсер группы сказал, что в альбом должно войти ровно n треков, каждый из которых — это ремикс на одну из двух песен группы. Поразмыслив, музыканты решили, что диск будет интересно слушать только в том случае, если на нём не более a треков подряд будут ремиксами на песню «Моя любовь» и не более b треков подряд будут ремиксами на песню «Я скучаю по тебе». Иначе есть риск, что даже самые преданные фанаты не дослушают альбом до конца.

Сколько существует различных вариантов записи альбома из n композиций, который будет интересно слушать? Вариант записи — это последовательность чисел 1 и 2, где единицы обозначают ремиксы на песню «Моя любовь», а двойки — ремиксы на песню «Я

скупаю по тебе». Два варианта считаются различными, если для некоторого i в одном варианте на i -м месте стоит единица, а в другом — двойка.

Формат входных данных

В единственной строке записаны целые числа n, a, b ($1 \leq a, b \leq 300; \max(a, b) + 1 \leq n \leq 50\,000$).

Формат выходных данных

Выведите количество различных вариантов записи альбома, взятое по модулю $10^9 + 7$.

Примеры

стандартный ввод	стандартный вывод
3 2 1	4

Замечания

В примере существуют следующие варианты записи: 112, 121, 211, 212.

Источник: [здесь](#).

Задача 20. Миллиардная Функция Васи

Ограничение по времени: 1 секунда

Ограничение по памяти: 64 мегабайта

Вася — начинающий математик — решил сделать вклад в развитие этой науки и прославиться на весь мир. Но как это сделать, когда самые интересные факты, типа теоремы Пифагора, давно уже доказаны? Правильно! Придумать что-то свое, оригинальное. Вот юный математик и придумал Теорию Функций Васи, посвященную изучению поведения этих самых функций. Функции Васи (ФВ) устроены довольно просто: значением N -й ФВ в точке S будет количество чисел от 1 до N , имеющих сумму цифр S . Вам, как крутым программистам, Вася поручил найти значения миллиардной ФВ (то есть ФВ с $N = 10^9$), так как сам он с такой задачей не справится. А Вам слабо?

Формат входных данных

Целое число S ($1 \leq S \leq 81$).

Формат выходных данных

Значение миллиардной Функции Васи в точке S .

Примеры

стандартный ввод	стандартный вывод
1	10

Источник: [здесь](#).

Задача 21. Трипростые числа

Ограничение по времени: 1 секунда

Ограничение по памяти: 64 мегабайта

Отдых на море — это замечательно! Но вот программисту Паше ужасно скучно лежать на пляже в Турции. Настолько скучно, что Паша решил посчитать количество трехзначных простых чисел. Он так увлекся этим занятием, что начал изучать 3-простые числа. Так Паша называет числа, у которых любые 3 подряд идущие цифры образуют трехзначное простое число. Паша уже начал работу над теорией божественного происхождения таких чисел, когда какие-то хулиганы окатили Пашу холодной водой и

стали кричать какие-то загадочные слова вроде «Sunstroke!», «Sonnenstich!» и «Colpo di sole!»

Вам предстоит продолжить дело Паши и выяснить насколько часты (или редки) 3-простые числа.

Формат входных данных

Ввод содержит целое число n ($3 \leq n \leq 10000$).

Формат выходных данных

Выведите количество n -значных 3-простых чисел, вычисленное по модулю $10^9 + 9$.

Примеры

стандартный ввод	стандартный вывод
4	204

Источник: [здесь](#).