

Задача №1

A+B

(Время: 1 сек. Память: 16 Мб Сложность: 2%)

Требуется сложить два целых числа A и B.

Входные данные

В единственной строке входного файла INPUT.TXT записано два натуральных числа через пробел, не превышающих 10^9 .

Выходные данные

В единственную строку выходного файла OUTPUT.TXT нужно вывести одно целое число — сумму чисел A и B.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	2 3	5

Задача №2

Сумма

(Время: 1 сек. Память: 16 Мб Сложность: 19%)

Требуется посчитать сумму целых чисел от 1 до N.

Входные данные

В единственной строке входного файла INPUT.TXT записано единственное целое число N, не превышающее по абсолютной величине 10^4 .

Выходные данные

В единственную строку выходного файла OUTPUT.TXT нужно вывести одно целое число — сумму чисел от 1 до N.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	5	15

Задача №3

Пятью пять - двадцать пять!

(Время: 1 сек. Память: 16 Мб Сложность: 8%)

Вася и Петя учатся в школе в одном классе. Недавно Петя поведал Васе о хитром способе возведения в квадрат натуральных чисел, оканчивающихся на цифру 5. Теперь Вася может с легкостью возводить в квадрат двузначные (и даже некоторые трехзначные) числа, оканчивающиеся на 5. Способ заключается в следующем: для возведения в квадрат числа, оканчивающегося на 5 достаточно умножить число, полученное из исходного вычеркиванием

последней пятерки на следующее по порядку число, затем остается лишь приписать «25» к получившемуся результату справа. Например, для того, чтобы возвести число 125 в квадрат достаточно 12 умножить на 13 и приписать 25, т.е. приписывая к числу $12 \cdot 13 = 156$ число 25, получаем результат 15625, т.е. $125^2 = 15625$. Напишите программу, возводящую число, оканчивающееся на 5, в квадрат для того, чтобы Вася смог проверить свои навыки.

Входные данные

В единственной строке входного файла INPUT.TXT записано одно натуральное число A , оканчивающееся на цифру 5, не превышающее $4 \cdot 10^5$.

Выходные данные

В выходной файл OUTPUT.TXT выведите одно натуральное число - A^2 без лидирующих нулей.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	5	25
2	75	5625
3	4255	18105025

Задача №4 Игра

(Время: 1 сек. Память: 16 Мб Сложность: 4%)

В свободное время одноклассники Вася и Петя любят играть в различные логические игры: морской бой, крестики-нолики, шахматы, шашки и многое другое. Ребята уже испробовали и поиграли во всевозможные классические игры подобного рода, включая компьютерные. Однажды им захотелось сыграть во что-нибудь новое, но ничего подходящего найти не удалось. Тогда Петя придумал следующую игру «Угадайка»: Играют двое участников. Первый загадывает любое трехзначное число, такое что первая и последняя цифры отличаются друг от друга более чем на единицу. Далее загадавший число игрок переворачивает загаданное число, меняя первую и последнюю цифры местами, таким образом получая еще одно число. Затем из максимального из полученных двух чисел вычитается минимальное. Задача второго игрока – угадать по первой цифре полученного в результате вычитания числа само это число. Например, если Вася загадал число 487, то перестановкой первой и последней цифры он получит число 784. После чего ему придется вычесть из 784 число 487, в результате чего получится число 297, которое и должен отгадать Петя по указанной первой цифре «2», взятой из этого числа. Петя успевает лучше Васи по математике, поэтому практически всегда выигрывает в играх такого типа. Но в данном случае Петя схитрил и специально придумал такую игру, в которой он не проиграет Васе в любом случае. Дело в том, что придуманная Петей игра имеет выигрышную стратегию, которая заключается в следующем: искомое число всегда является трехзначным и вторая его цифра всегда равна девяти, а для получения значения последней достаточно отнять от девяти первую, т.е. в рассмотренном выше случае последняя цифра равна $9 - 2 = 7$. Помогите Пете еще упростить процесс отгадывания числа по заданной его первой цифре, написав соответствующую программу.

Входные данные

В единственной строке входного файла INPUT.TXT задана единственная цифра K , соответствующая первой цифре полученного Васей в результате вычитания наименьшего загаданного Васей значения из наибольшего.

Выходные данные

В выходной файл OUTPUT.TXT нужно вывести значение полученной Васей разности.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	5	594
2	2	297
3	7	792

Задача №5 Статистика

(Время: 1 сек. Память: 16 Мб Сложность: 25%)

Вася не любит английский язык, но каждый раз старается получить хотя бы четверку за четверть, чтобы оставаться ударником. В текущей четверти Вася заметил следующую закономерность: по нечетным дням месяца он получал тройки, а по четным – четверки. Так же он помнит, в какие дни он получал эти оценки. Поэтому он выписал на бумажке все эти дни для того, чтобы оценить, сколько у него троек и сколько четверок. Помогите Васе это сделать, расположив четные и нечетные числа в разных строчках. Вася может рассчитывать на оценку 4, если четверок не меньше, чем троек.

Входные данные

В первой строке входного файла INPUT.TXT записано единственное число N – количество элементов целочисленного массива ($1 \leq N \leq 100$). Вторая строка содержит N чисел, представляющих заданный массив. Каждый элемент массива – натуральное число от 1 до 31. Все элементы массива разделены пробелом.

Выходные данные

В первую строку выходного файла OUTPUT.TXT нужно вывести числа, которые соответствуют дням месяцев, в которые Вася получил тройки, а во второй строке соответственно расположить числа месяца, в которые Вася получил четверки. В третьей строке нужно вывести «YES», если Вася может рассчитывать на четверку и «NO» в противном случае. В каждой строчке числа следует выводить в том же порядке, в котором они идут во входных данных. При выводе, числа отделяются пробелом.

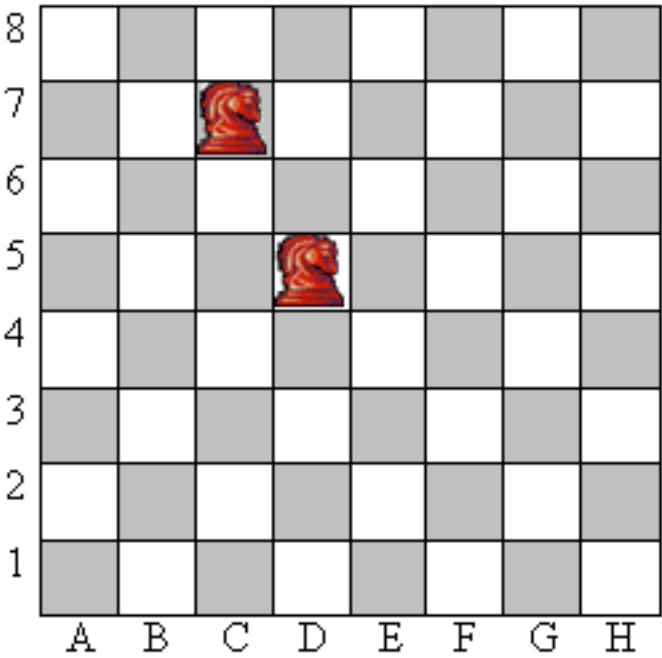
Примеры

№	INPUT.TXT	OUTPUT.TXT
1	5 4 16 19 31 2	19 31 4 16 2 YES
2	8 29 4 7 12 15 17 24 1	29 7 15 17 1 4 12 24 NO

Задача №6 Шахматы

(Время: 1 сек. Память: 16 Мб Сложность: 18%)

Совсем недавно Вася занялся программированием и решил реализовать собственную программу для игры в шахматы. Но у него возникла проблема определения правильности хода конем, который делает пользователь. Т.е. если пользователь вводит значение «C7-D5», то программа должна определить это как правильный ход, если же введено «E2-E4», то ход неверный. Так же нужно проверить корректность записи ввода: если например, введено «D9-N5», то программа должна определить данную запись как ошибочную. Помогите ему осуществить эту проверку!



Входные данные

В единственной строке входного файла INPUT.TXT записан текст хода, который указал пользователь. Пользователь не может ввести строку, длиннее 5 символов.

Выходные данные

В выходной файл OUTPUT.TXT нужно вывести «YES», если указанный ход конем верный, если же запись корректна (в смысле правильности записи координат), но ход невозможен, то нужно вывести «NO». Если же координаты не определены или заданы некорректно, то вывести сообщение «ERROR».

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	C7-D5	YES
2	E2-E4	NO
3	BSN	ERROR

Задача №7
Золото племени АББА

(Время: 1 сек. Память: 16 Мб Сложность: 40%)

Главный вождь племени Абба не умеет считать. В обмен на одну из его земель вождь другого племени предложил ему выбрать одну из трех куч с золотыми монетами. Но вождю племени Абба хочется получить наибольшее количество золотых монет. Помогите вождю сделать правильный выбор!

Входные данные

В первой строке входного файла INPUT.TXT записаны три натуральных числа через пробел. Каждое из чисел не превышает 10^{100} .

Выходные данные

В выходной файл OUTPUT.TXT нужно вывести одно целое число — максимальное количество монет, которые может взять вождь.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	5 7 3	7
2	987531 234 86364	987531
3	189285 283 4958439238923098349024	4958439238923098349024

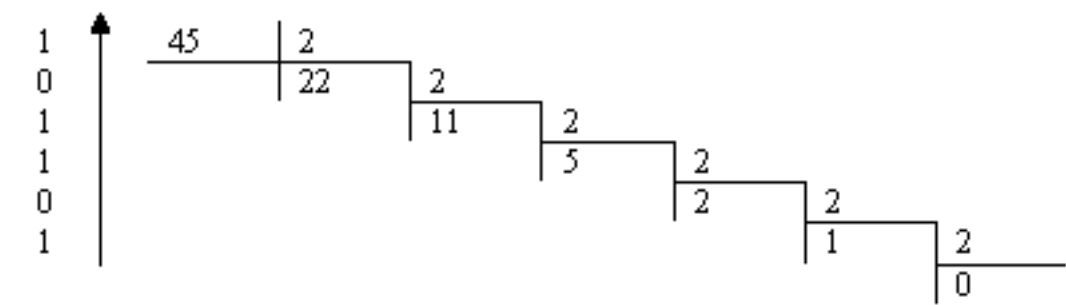
Задача №8 Арифметика

(Время: 1 сек. Память: 16 Мб Сложность: 5%)

В прошлом году Вася пошел в школу и научился считать. В этом году он изучил таблицу умножения и теперь умеет перемножать любые числа от 1 до 10 без ошибок. Друг Петя рассказал ему про системы счисления, отличные от десятичной. В частности, про двоичную, восьмеричную и даже шестнадцатеричную. Теперь Вася без труда (но уже с помощью листка и ручки) может перемножать числа от 1 до 10 и в этих системах, используя перевод из нестандартной системы в десятичную и обратно из десятичной. Например, если Васе нужно перемножить числа 101 и 1001 в двоичной системе, то он сначала эти числа переводит в десятичное представление следующим образом:

$$(101)_2 = 1 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0 = 4 + 0 + 1 = 5$$
$$(1001)_2 = 1 \cdot 2^3 + 0 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0 = 8 + 0 + 0 + 1 = 9$$

После чего перемножение чисел 5 и 9 Вася с легкостью производит в десятичной системе счисления в уме и получает число 45. Далее производится перевод из десятичной системы счисления в двоичную. Для этого Вася делит число 45 на 2 (порядок системы счисления), запоминая остатки от деления, до тех пор пока в результате не останется число 0:



Ответ составляется из полученных остатков от деления путем их записи в обратном порядке. Таким образом Вася получает результат: $(101)_2 * (1001)_2 = (101101)_2$. Но теперь Вася изучает таблицу умножения чисел от 1 до 100 в десятичной системе счисления, а поскольку запомнить такую таблицу очень сложно, то Васе придется очень долго ее зубрить. Составьте для Васи программу, которая поможет ему проверять свои знания.

Входные данные

Во входном файле INPUT.TXT записаны три натуральных числа A, B и C через пробел. Числа A и B $\leq 10^2$, а C $\leq 10^6$.

Выходные данные

В выходной файл нужно вывести YES в том случае, если $A*B=C$ и вывести NO в противном случае.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	8 54 432	YES
2	16 19 777	NO

Задача №9 Домашнее задание

(Время: 1 сек. Память: 16 Мб Сложность: 27%)

Петя успевает по математике лучше всех в классе, поэтому учитель задал ему сложное домашнее задание, в котором нужно в заданном наборе целых чисел найти сумму всех положительных элементов, затем найти где в заданной последовательности находятся максимальный и минимальный элемент и вычислить произведение чисел, расположенных между ними. Так же известно, что минимальный и максимальный элемент встречаются в заданном множестве чисел только один раз. Поскольку задач такого рода учитель дал Пете около ста, то Петя как сильный программист смог написать программу, которая по заданному набору чисел самостоятельно находит решение. А Вам слабо?

Входные данные

В первой строке входного файла INPUT.TXT записано единственное число N – количество элементов массива. Вторая строка содержит N целых чисел, представляющих заданный массив. Все элементы массива разделены пробелом. Каждое из чисел во входном файле не превышает 10^2 по абсолютной величине.

Выходные данные

В единственную строку выходного файла OUTPUT.TXT нужно вывести два числа, разделенных пробелом: сумму положительных элементов и произведение чисел, расположенных между минимальным и максимальным элементами. Значения суммы и произведения не превышают по модулю $3 \cdot 10^4$.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	5 -7 5 -1 3 9	17 -15
2	8 3 14 -9 4 -5 1 -12 4	26 180
3	10 -5 1 2 3 4 5 6 7 8 -3	36 5040

Задача №10 Уравнение

(Время: 1 сек. Память: 16 Мб Сложность: 17%)

Вася в школе изучил квадратные уравнения и понял, как они легко решаются путем вычисления дискриминанта. Но Петя поведал ему о методе решения кубических уравнений вида $A \cdot X^3 + B \cdot X^2 + C \cdot X + D = 0$. На факультативе по математике Васе задали решить около ста

уравнений как раз такого вида. Но, к сожалению, Вася забыл формулы, о которых рассказывал ему Петя. Но Васе было известно, что все корни уравнений – целые числа и находятся на отрезке $[-100, 100]$. Поэтому у Васи есть шанс найти их методом перебора, но для этого ему придется затратить уйму времени, т.к. возможно необходимо будет осуществить перебор нескольких тысяч значений. Помогите Васе написать программу, которая поможет ему найти корни кубических уравнений!

Входные данные

В единственной строке входного файла INPUT.TXT записаны 4 числа: A, B, C и D – целые коэффициенты кубического уравнения. Каждый коэффициент по модулю меньше 32768.

Выходные данные

В единственную строку выходного файла OUTPUT.TXT нужно вывести через пробел в порядке возрастания все корни заданного кубического уравнения. Кратные корни следует выводить только один раз.

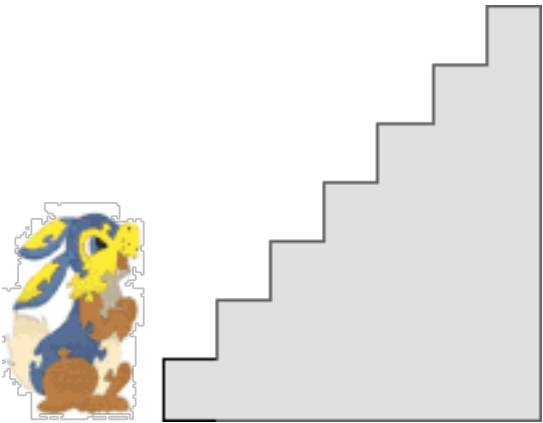
Примеры

№	INPUT.TXT	OUTPUT.TXT
1	1 -3 0 0	0 3
2	3 -15 18 0	0 2 3
3	1 -7 -33 135	-5 3 9

Задача №11 Зайчик

(Время: 1 сек. Память: 16 Мб Сложность: 68%)

В нашем зоопарке появился заяц. Его поместили в клетку, и чтобы ему не было скучно, директор зоопарка распорядился поставить в его клетке лесенку. Теперь наш зайчик может прыгать по лесенке вверх, перепрыгивая через ступеньки. Лестница имеет определенное количество ступенек N. Заяц может одним прыжком преодолеть не более K ступенек. Для разнообразия зайчик пытается каждый раз найти новый путь к вершине лестницы. Директору любопытно, сколько различных способов есть у зайца добраться до вершины лестницы при заданных значениях K и N. Помогите директору написать программу, которая поможет вычислить это количество. Например, если $K=3$ и $N=4$, то существуют следующие маршруты: 1+1+1+1, 1+1+2, 1+2+1, 2+1+1, 2+2, 1+3, 3+1. Т.е. при данных значениях у зайца всего 7 различных маршрутов добраться до вершины лестницы.



Входные данные

В единственной строке входного файла INPUT.TXT записаны два натуральных числа K и N ($1 \leq K \leq N \leq 300$). K - максимальное количество ступенек, которое может преодолеть заяц одним прыжком, N – общее число ступенек лестницы.

Выходные данные

В единственную строку выходного файла OUTPUT.TXT нужно вывести количество

возможных вариантов различных маршрутов зайца на верхнюю ступеньку лестницы без ведущих нулей.

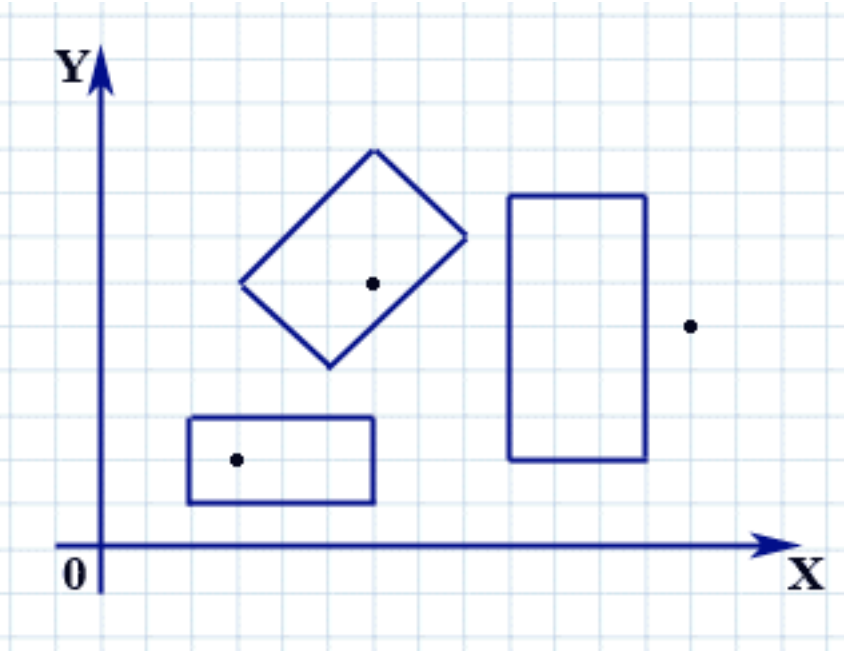
Примеры

№	INPUT.TXT	OUTPUT.TXT
1	1 3	1
2	2 7	21
3	3 10	274

Задача №12 Дачники

(Время: 1 сек. Память: 16 Мб Сложность: 45%)

Всем известно, что дачники – народ странный, почти такой же, как и программисты. Строят они свои дачи непонятно где, да и выращивают там непонятно что и непонятно зачем. А уж как они туда добираются, это другая история: кто на автобусе, кто на электричке, кто на автомобиле, ну а кто-то вовсе пешком ходит от дома и до самого участка. Так что не стоит удивляться, если вдруг Вы узнаете, что некое садоводческое товарищество располагается на острове, а дачники добираются до него самолетом. Да еще и на этом острове может не быть посадочной полосы, так что высадиться на остров можно, только прыгая с парашютом (мы уж не рассматриваем то, как они возвращаются с дач домой). Рассмотрим этот уникальный случай. Пилот всегда старается осуществить высадку парашютистов таким образом, чтобы дачники приземлялись как можно ближе к своим прямоугольным участкам. Пилоту интересно знать: сколько дачников приземлится на свои участки? Помогите ему решить эту задачу!



Входные данные

В первой строке входного файла INPUT.TXT записано натуральное число N (1 ≤ N ≤ 1000) – количество дачников, далее идут N строк, в каждой из которых описаны координаты каждого дачника и его участка:
X Y X1 Y1 X2 Y2 X3 Y3 X4 Y4
где
(X,Y) – координаты приземления парашютиста
(X1, Y1, X2, Y2, X3, Y3, X4,Y4) – координаты прямоугольного участка на плоскости, указанные последовательно.
Все координаты – целые числа, не превышающие 50000 по абсолютной величине

Выходные данные

В выходной файл OUTPUT.TXT нужно вывести количество дачников, приземлившихся на свой участок.

Пример

--	--	--

№	INPUT.TXT	OUTPUT.TXT
1	3 6 6 3 6 6 9 8 7 5 4 13 5 9 2 9 8 12 8 12 2 3 2 2 1 2 3 6 3 6 1	2

Задача №13

Быки и коровы

(Время: 1 сек. Память: 16 Мб Сложность: 26%)

Петя и Вася часто играют в различные логические игры. Недавно Петя поведал Васе о новой игре «Быки и коровы» и теперь они играют в эту игру сутками. Суть игры очень проста: Петя загадывает четырехзначное число, состоящее из различных цифр. Вася отгадывает задуманное Петей число, перебирая возможные варианты. Каждый раз Вася предлагает вариант своего числа, а Петя делает Васе подсказку: сообщает количество быков и коров, после чего Вася с учетом подсказки продолжает отгадывание числа до тех пор, пока не отгадает. Быки – это количество цифр в предложенном Васей числе, совпадающих по значению и стоящих в правильной позиции в задуманном Петей числе. Коровы – количество цифр, совпадающих по значению, но находящихся в неверной позиции. Например, если Петя задумал число 5671, а Вася предложил вариант 7251, то число быков равно 1 (только цифра 1 на своем месте), а число коров равно 2 (только цифры 7 и 5 не на своих местах). Петя силен в математике, но даже он может ошибаться. Помогите Пете написать программу, которая бы по загаданному Петей и предложенному Васей числам сообщала количество быков и коров.

Входные данные

В единственной строке входного файла INPUT.TXT записано два четырехзначных натуральных числа A и B через пробел, где A – загаданное Петей число, а B – предложенный Васей вариант.

Выходные данные

В выходной файл OUTPUT.TXT нужно вывести два целых числа через пробел — количество быков и коров.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	5671 7251	1 2
2	1234 1234	4 0
3	2034 6234	2 1

Задача №14

НОК

(Время: 1 сек. Память: 16 Мб Сложность: 24%)

Требуется написать программу, определяющую наименьшее общее кратное (НОК) чисел a и b.

Входные данные

В единственной строке входного файла INPUT.TXT записаны два натуральных числа A и B через пробел, не превышающих 46340.

Выходные данные

В единственную строку выходного файла OUTPUT.TXT нужно вывести одно целое число — НОК чисел A и B.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	36 27	108
2	39 65	195

Задача №15 Дороги

(Время: 1 сек. Память: 16 Мб Сложность: 18%)

В галактике «Milky Way» на планете «Snowflake» есть N городов, некоторые из которых соединены дорогами. Император галактики «Milky Way» решил провести инвентаризацию дорог на планете «Snowflake». Но, как оказалось, он не силен в математике, поэтому он просит вас сосчитать количество дорог. Требуется написать программу, помогающую императору сосчитать количество дорог на планете «Snowflake».

Входные данные

В первой строке входного файла INPUT.TXT записано число N ($0 \leq N \leq 100$). В следующих N строках записано по N чисел, каждое из которых является единичкой или ноликом. Причем, если в позиции (i, j) квадратной матрицы стоит единичка, то i-ый и j-ый города соединены дорогами, а если нолик, то не соединены.

Выходные данные

В выходной файл OUTPUT.TXT необходимо вывести число, определяющее количество дорог на планете «Snowflake».

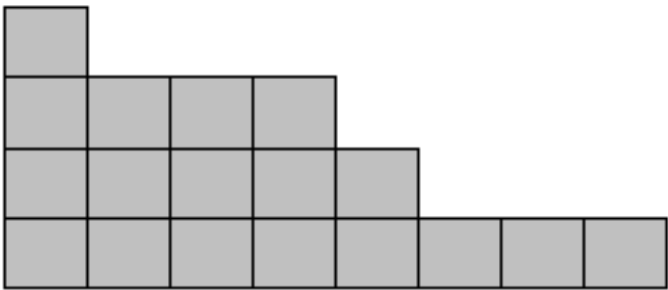
Пример

№	INPUT.TXT	OUTPUT.TXT
1	5 0 1 0 0 0 1 0 1 1 0 0 1 0 0 0 0 1 0 0 0 0 0 0 0 0	3

Задача №16 Лесенка

(Время: 1 сек. Память: 16 Мб Сложность: 55%)

Лесенкой называется набор кубиков, в котором каждый более верхний слой содержит кубиков меньше, чем предыдущий. Требуется написать программу, вычисляющую число лесенок, которое можно построить из N кубиков.



Входные данные

Во входном файле INPUT.TXT записано натуральное число N ($1 \leq N \leq 100$) – количество кубиков в лесенке.

Выходные данные

В выходной файл OUTPUT.TXT необходимо вывести число лесенок, которые можно построить из N кубиков.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	3	2
2	6	4

Задача №17 Поле чудес

(Время: 1 сек. Память: 16 Мб Сложность: 31%)

Для игры в «Поле чудес» используется круглый барабан, разделенный на сектора, и стрелка. В каждом секторе записано некоторое число. В различных секторах может быть записано одно и то же число. Однажды ведущий игры решил изменить правила. Он сам стал вращать барабан и называть игроку (который барабана не видел) все числа подряд в том порядке, в котором на них указывала стрелка в процессе вращения барабана. Получилось так, что барабан сделал целое число оборотов, то есть последний сектор совпал с первым. После этого, ведущий задал участнику вопрос: какое наименьшее число секторов может быть на барабане? Требуется написать программу, отвечающую на этот вопрос ведущего.

Входные данные

В первой строке входного файла INPUT.TXT записано число N – количество чисел, которое назвал ведущий ($2 \leq N \leq 30000$). Во второй строке записано N чисел, на которые указывала стрелка в процессе вращения барабана. Первое число всегда совпадает с последним (в конце стрелка указывает на тот же сектор, что и в начале). Числа, записанные в секторах барабана – натуральные, не превышающие 32000.

Выходные данные

В выходной файл OUTPUT.TXT необходимо вывести одно число – минимальное число секторов, которое может быть на барабане.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	13 5 3 1 3 5 2 5 3 1 3 5 2 5	6

2	4 1 1 1 1	1
3	4 1 2 3 1	3

Задача №18

Факториал

(Время: 1 сек. Память: 16 Мб Сложность: 42%)

Требуется вычислить факториал целого числа N. Факториал обозначают как N! и вычисляют по формуле:

$$N! = 1 * 2 * 3 * \dots * (N-1) * N, \text{ причем } 0! = 1.$$

Так же допустимо рекуррентное соотношение: $N! = (N-1)! * N$

Входные данные

В единственной строке входного файла INPUT.TXT записано одно целое неотрицательное число N (N < 1000).

Выходные данные

В выходной файл OUTPUT.TXT нужно вывести одно целое число — значение N!.

Примеры

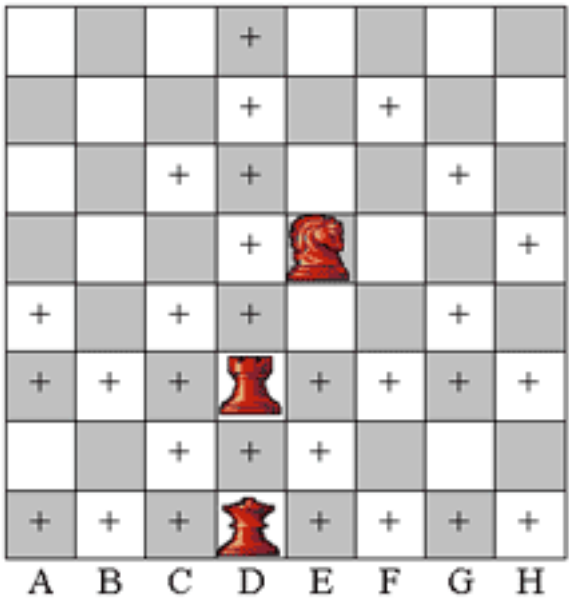
№	INPUT.TXT	OUTPUT.TXT
1	1	1
2	3	6
3	5	120

Задача №19

Ферзь, ладья и конь

(Время: 1 сек. Память: 16 Мб Сложность: 34%)

На шахматной доске 8x8 расположены три фигуры: ферзь, ладья и конь. Требуется определить количество пустых полей доски, которые находятся под боем. Для простоты будем полагать, что фигуры могут «бить» через другие фигуры. Например, в рассмотренной справа ситуации будем считать, что ферзь бьет D5 через ладью.



Входные данные

В единственной строке входного файла INPUT.TXT записаны через пробел координаты расположения трех фигур: ферзя, ладьи и коня соответственно. Каждая координата состоит из одного латинского символа (от A до H) и одной цифры (от 1 до 8).

Выходные данные

В выходной файл OUTPUT.TXT нужно вывести количество пустых полей, которые бьют

указанные во входных данных фигуры.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	D1 D3 E5	29

Задача №20

Пилообразная последовательность

(Время: 1 сек. Память: 16 Мб Сложность: 38%)

Последовательность $a_1, a_2, a_3, \dots, a_{n-1}, a_n$ называется пилообразной, если она удовлетворяет одному из следующих условий:

- 1) $a_1 < a_2 > a_3 < \dots > a_{n-1} < a_n$
- 2) $a_1 > a_2 < a_3 > \dots < a_{n-1} > a_n$

Дана числовая последовательность. Требуется определить длину самой длинной ее пилообразной непрерывной подпоследовательности.

Входные данные

В первой строке входного файла INPUT.TXT записано натуральное число N – количество элементов последовательности. Во второй строке файла через пробел записаны N элементов целочисленной последовательности $\{a_i\}$. Ограничения: $N < 10^6, |a_i| < 32000$.

Выходные данные

В единственную строку выходного файла OUTPUT.TXT нужно вывести одно целое число — длину самой длинной непрерывной пилообразной подпоследовательности.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	3 1 2 3	2
2	12 5 7 6 3 4 2 7 1 8 9 4 5	7
3	5 1 -2 3 -4 5	5

Задача №21

Зарплата

(Время: 1 сек. Память: 16 Мб Сложность: 4%)

В отделе работают 3 сотрудника, которые получают заработную плату в рублях. Требуется определить: на сколько зарплата самого высокооплачиваемого из них отличается от самого низкооплачиваемого.

Входные данные

В единственной строке входного файла INPUT.TXT записаны размеры зарплат всех сотрудников через пробел. Каждая заработная плата – это натуральное число, не превышающее 10^5 .

Выходные данные

В выходной файл OUTPUT.TXT необходимо вывести одно целое число — разницу между максимальной и минимальной зарплатой.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	100 500 1000	900
2	36 11 20	25

Задача №22 Единицы

(Время: 1 сек. Память: 16 Мб Сложность: 16%)

На уроках информатики вас, наверное, учили переводить числа из одних систем счисления в другие и выполнять другие подобные операции. Пришло время продемонстрировать эти знания. Найдите количество единиц в двоичной записи заданного числа.

Входные данные

Во входном файле INPUT.TXT записано целое число n ($0 \leq n \leq 2 \cdot 10^9$).

Выходные данные

В единственную строку выходного файла OUTPUT.TXT нужно вывести одно целое число — количество двоичных единиц в записи числа n .

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	5	2
2	7	3

Задача №23 Гадание

(Время: 1 сек. Память: 16 Мб Сложность: 13%)

Как и многие другие девочки, Маша любит разные гадания. Некоторое время назад Маша узнала новый способ гадать на числах – для какого-нибудь интересующего ее натурального числа n надо посчитать сумму всех чисел, на которые n делится без остатка. Маша не очень любит арифметику, и попросила вас написать программу, которая автоматизирует процесс гадания.

Входные данные

В единственной строке входного файла INPUT.TXT записано натуральное число n ($n \leq 1000$), которое Маша была вынуждена сообщить.

Выходные данные

В выходной файл OUTPUT.TXT выведите сумму всех натуральных делителей числа n .

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	6	12
2	10	18

Задача №24 Вырубка деревьев

(Время: 1 сек. Память: 16 Мб Сложность: 46%)

Король Флатландии решил вырубить некоторые деревья, растущие перед его дворцом. Деревья перед дворцом короля посажены в ряд, всего там растет n деревьев, расстояния между соседними деревьями одинаковы.

После вырубки перед дворцом должно остаться m деревьев, и расстояния между соседними деревьями должны быть одинаковыми. Помогите королю выяснить, сколько существует способов вырубки деревьев.

Требуется написать программу, которая по заданным числам n и m определит, сколько существует способов вырубки некоторых из n деревьев так, чтобы после вырубки осталось m деревьев и соседние деревья находились на равном расстоянии друг от друга.

Входные данные

Входной файл INPUT.TXT содержит два целых числа n и m ($0 \leq m, n \leq 1000$).

Выходные данные

В единственную строку выходного файла OUTPUT.TXT нужно вывести одно целое число — искомое число способов.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	5 3	4

Пояснение к примеру

Если обозначить условно исходное расположение деревьев перед дворцом как «TTTTT», то возможные результаты после вырубки следующие:

«TTT..», «.TTT.», «..TTT», «T.T.T».

Задача №25 Больше-меньше

(Время: 1 сек. Память: 16 Мб Сложность: 3%)

Одна из основных операций с числами – их сравнение. Мы подозреваем, что вы в совершенстве

владеете этой операцией и можете сравнивать любые числа, в том числе и целые. В данной задаче необходимо сравнить два целых числа.

Входные данные

В двух строчках входного файла INPUT.TXT записаны числа A и B, не превосходящие по абсолютной величине $2 \cdot 10^9$.

Выходные данные

Запишите в выходной файл OUTPUT.TXT один символ “<”, если $A < B$, “>”, если $A > B$ и “=”, если $A = B$.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	5 7	<
2	-7 -12	>
3	13 13	=

Задача №26 Две окружности

(Время: 1 сек. Память: 16 Мб Сложность: 17%)

На плоскости даны две окружности. Требуется проверить, пересекаются ли они.

Входные данные

Входной файл INPUT.TXT состоит из двух строк. На каждой строке записана информация об одной окружности – координаты ее центра x и y (целые числа, по модулю не превосходящие 5000) и радиус (целое число $1 \leq r \leq 1000$).

Выходные данные

В выходной файл OUTPUT.TXT выведите «YES», если окружности пересекаются, и «NO» в противном случае.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	0 0 2 0 3 2	YES
2	1 1 1 4 4 1	NO

Задача №27 Художник

(Время: 1 сек. Память: 16 Мб Сложность: 31%)

Известный художник решил написать новый шедевр. После многих дней усердной работы он захотел исследовать свое творение. Художник вспомнил, что картина писалась следующим образом: сначала был взят белый холст, имеющий форму прямоугольника шириной w и высотой h . Затем художник нарисовал на этом холсте n прямоугольников со сторонами, параллельными сторонам холста и вершинами, расположенными в целочисленных координатах. Помогите художнику определить площадь незакрашенной части холста.

Входные данные

Первая строка входного файла INPUT.TXT содержит два натуральных числа w и h ($1 \leq w, h \leq 100$). Во второй строке записано целое число n ($0 \leq n \leq 5000$) – количество прямоугольников. Следующие n строк содержат информацию о всех прямоугольниках. Каждая строка описывает один прямоугольник в виде четырех чисел x_1, y_1, x_2, y_2 , где (x_1, y_1) и (x_2, y_2) – координаты левого верхнего и правого нижнего угла прямоугольника соответственно.

Выходные данные

Выведите в выходной файл OUTPUT.TXT одно целое число – площадь незакрашенной части холста.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	5 5 2 1 1 3 3 2 2 4 4	18
2	6 7 3 0 0 5 5 1 1 4 4 2 2 3 3	17

Задача №28 Симметрия

(Время: 1 сек. Память: 16 Мб Сложность: 19%)

Многие из вас, вероятно, знакомы с понятием симметрии относительно прямой. Пусть на плоскости расположена прямая L и точка A . Точка B называется симметричной точке A относительно прямой L , если отрезок AB перпендикулярен прямой L и делится пополам точкой пересечения с ней. В частности, если точка A лежит на прямой L , то точка B совпадает с точкой A .

Задана прямая L , параллельная одной из осей координат, и точка A . Найдите точку B , симметричную A относительно L .

Входные данные

Первая строка входного файла INPUT.TXT содержит 4 числа: x_1, y_1, x_2, y_2 – координаты двух различных точек, через которые проходит прямая L . Вторая строка входного файла содержит 2 числа x_A и y_A – координаты точки A . Все числа во входном файле целые и не превосходят 10^8 по модулю.

Выходные данные

В выходной файл OUTPUT.TXT выведите числа x_B и y_B – координаты точки В.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	0 0 0 1 10 10	-10 10
2	0 0 1 0 10 10	10 -10

Задача №29 Компьютерная игра

(Время: 1 сек. Память: 16 Мб Сложность: 38%)

Вы можете вспомнить хоть одного своего знакомого до двадцатилетнего возраста, который в детстве не играл в компьютерные игры? Если да, то может быть вы и сами не знакомы с этим развлечением? Впрочем, трудностей при решении этой задачи это создать не должно.

Во многих старых играх с двумерной графикой можно столкнуться с подобной ситуацией. Какой-нибудь герой прыгает по платформам (или островкам), которые висят в воздухе. Он должен перебраться от одного края экрана до другого. При этом при прыжке с одной платформы на соседнюю, у героя уходит $|y_2 - y_1|$ единиц энергии, где y_1 и y_2 – высоты, на которых расположены эти платформы. Кроме того, у героя есть суперприем, который позволяет перескочить через платформу, но на это затрачивается $3 * |y_3 - y_1|$ единиц энергии. Конечно же, энергию следует расходовать максимально экономно.

Предположим, что вам известны координаты всех платформ в порядке от левого края до правого. Сможете ли вы найти, какое минимальное количество энергии потребуется герою, чтобы добраться с первой платформы до последней?

Входные данные

В первой строке входного файла INPUT.TXT записано количество платформ n ($1 \leq n \leq 30000$). Вторая строка содержит n натуральных чисел, не превосходящих 30000 – высоты, на которых располагаются платформы.

Выходные данные

В выходной файл OUTPUT.TXT запишите единственное число – минимальное количество энергии, которую должен потратить игрок на преодоление платформ (конечно же в предположении, что cheat-коды использовать нельзя).

Пример

№	INPUT.TXT	OUTPUT.TXT
1	3 1 5 10	9
2	3 1 5 2	3

Задача №30

Часы

(Время: 1 сек. Память: 16 Мб Сложность: 35%)

Петя очень любит наблюдать за электронными часами. Он целыми днями смотрел на часы и считал, сколько раз встречается каждая цифра. Через несколько месяцев он научился по любому промежутку времени говорить, сколько раз на часах за это время встретится каждая цифра, и очень гордился этим.

Вася решил проверить Петю, но он не знает, как это сделать. Вася попросил Вас помочь ему. Напишите программу, решающую эту задачу.

Входные данные

Первая и вторая строки входного файла INPUT.TXT содержат начало и конец промежутка времени соответственно. Начальное время не превосходит конечное. Время задается в формате hh:mm:ss ($0 \leq hh < 24$, $0 \leq mm < 60$, $0 \leq ss < 60$). hh, mm, ss дополнены ведущими нулями до двух символов. Эти нули также учитываются при подсчете числа цифр.

Выходные данные

Выходной файл OUTPUT.TXT должен содержать 10 строк. В i-ой строке должно быть написано, сколько раз встречается цифра i-1.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	23:59:58 23:59:59	0
		0
		2
		2
		0
		4
		0
		0
		1
		3
2	13:24:09 13:24:40	5
		45
		45
		45
		36
		3
		3
		3
		3
		4

Задача №31

Неподвижные точки

(Время: 1 сек. Память: 16 Мб Сложность: 57%)

Перестановкой $P[1..n]$ размера n называется набор чисел от 1 до n , расположенных в

определенном порядке. При этом в нем должно присутствовать ровно один раз каждое из этих чисел. Примером перестановок являются 1,3,4,5,2 (для n=5) и 3,2,1 (для n=3), а, например, 1,2,3,4,5,1 перестановкой не является, так как число 1 встречается два раза.

Число i называется неподвижной точкой для перестановки P , если $P[i] = i$. Например, в перестановке 1,3,4,2,5 ровно две неподвижных точки: 1 и 5, а перестановка 4,3,2,1 не имеет неподвижных точек.

Даны два числа: n и k . Найдите количество перестановок размера n с ровно k неподвижными точками.

Входные данные

Входной файл INPUT.TXT содержит два целых числа n ($1 \leq n \leq 9$) и k ($0 \leq k \leq n$).

Выходные данные

В выходной файл OUTPUT.TXT выведите ответ на задачу.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	5 2	20
2	9 6	168
3	2 1	0
4	9 0	133496

Задача №32 Годовой баланс

(Время: 1 сек. Память: 16 Мб Сложность: 33%)

В конторе «Рога и Копыта» подходит время подведения годового баланса. В бухгалтерию поступили сведения о том, что, согласно документам, суммарный расход составил a рублей, а суммарный приход – b рублей. Поскольку с реальным положением дел эти цифры все равно не имеют ничего общего, бухгалтер решил реализовать следующую свою идею. Как известно, при наборе чисел на компьютере люди часто вводят цифры в неправильном порядке. Поэтому бухгалтер хочет найти такой способ переставить цифры в числах a и b , чтобы в результате разность $a-b$ (и, соответственно, количество денег, которые он положит к себе в карман), была максимальна, а в случае можно будет сослаться на ошибку секретаря. При этом нельзя забывать о знаке чисел и о том, что ноль не может быть первой цифрой числа. Напишите программу, которая поможет бухгалтеру.

Входные данные

Входной файл INPUT.TXT содержит два целых числа a и b ($-10^9 < a, b < 10^9$).

Выходные данные

В выходной файл OUTPUT.TXT выведите одно целое число – наибольшую разность чисел, первое из которых может быть получено перестановкой цифр a , а второе – перестановкой цифр b .

Примеры

--	--	--

№	INPUT.TXT	OUTPUT.TXT
1	18 10	71
2	1 -23	33

Задача №33
Два бандита

(Время: 1 сек. Память: 16 Мб Сложность: 4%)

Бандиты Гарри и Ларри отдыхали на природе. Решив пострелять, они выставили на бревно несколько банок из-под кока-колы (не больше 10). Гарри начал простреливать банки по порядку, начиная с самой левой, Ларри — с самой правой. В какой-то момент получилось так, что они одновременно прострелили одну и ту же последнюю банку.

Гарри возмутился и сказал, что Ларри должен ему кучу денег за то, что тот лишил его удовольствия прострелить несколько банок. В ответ Ларри сказал, что Гарри должен ему еще больше денег по тем же причинам. Они стали спорить кто кому сколько должен, но никто из них не помнил сколько банок было в начале, а искать простреленные банки по всей округе было неохота. Каждый из них помнили только, сколько банок прострелил он сам.

Определите по этим данным, сколько банок не прострелил Гарри и сколько банок не прострелил Ларри.

Входные данные

В единственной строке входного файла INPUT.TXT записано 2 числа — количество банок, простреленных Гарри и Ларри соответственно.

Выходные данные

В файл OUTPUT.TXT выведите 2 числа — количество банок, не простреленных Гарри и Ларри соответственно.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	4 7	6 3

Задача №34
Секретное сообщение

(Время: 1 сек. Память: 16 Мб Сложность: 46%)

На секретную базу в Арктике поступила шифровка – последовательность из n десятичных цифр. Она содержит номер секретной базы в Антарктиде, который является последовательностью из k десятичных цифр. При этом для того, чтобы отличить его от ненужной Вам информации, он повторен в шифровке хотя бы два раза (возможно, эти два вхождения перекрываются).

Напишите программу, которая по шифровке и длине номера секретной базы определяет, содержит ли шифровка номер базы. Учтите, что у базы может быть несколько номеров, и все они могут быть переданы в шифровке.

Входные данные

Первая строка входного файла INPUT.TXT содержит два целых числа: n ($1 \leq n \leq 10^5$) и k ($1 \leq k \leq 5$) – длину шифровки и длину номера секретной базы соответственно. Вторая строка содержит n цифр – шифровку. Помните, что цифры в шифровке не разделяются пробелами.

Выходные данные

В выходной файл OUTPUT.TXT выведите «YES», если шифровка содержит номер секретной базы, и «NO» в противном случае.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	10 5 0123456789	NO
2	13 2 0123400056789	YES

Задача №35 Конечные автоматы

(Время: 1 сек. Память: 16 Мб Сложность: 11%)

Однажды известный профессор обнаружил описания k конечных автоматов. По его мнению, нетривиальность конечного автомата, имеющего n состояний и m переходов, можно описать целым числом $d = 19m + (n + 239) * (n + 366) / 2$. Чем больше d , тем больший интерес для науки представляет изучение его свойств.

Помогите профессору вычислить нетривиальность имеющихся у него автоматов.

Входные данные

Первая строка входного файла INPUT.TXT содержит целое число k ($1 \leq k \leq 10000$) – количество конечных автоматов. Следующие k строк содержат по два целых числа n_i ($0 \leq n_i \leq 1000$) и m_i ($0 \leq m_i \leq 26n_i^2$) – число состояний и переходов i -го автомата.

Выходные данные

Выходной файл OUTPUT.TXT должен состоять из k строк. На i -й строке выходного файла выведите одно число – нетривиальность i -го автомата.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	4 2 0 13 20 5 23 18 6	44344 48134 45699 49458
2	2 15 20 1000 26000	48767 1340237

Задача №36

Постулат Бертрана

(Время: 1 сек. Память: 16 Мб Сложность: 30%)

Постулат Бертрана (теорема Бертрана-Чебышева, теорема Чебышева) гласит, что для любого $n > 1$ найдется простое число p в интервале $n < p < 2n$. Такая гипотеза была выдвинута в 1845 году французским математиком Джозефом Бертраном (проверившим ее до $n=3000000$) и доказана в 1850 году Пафнутием Чебышевым. Раманужан в 1920 году нашел более простое доказательство, а Эрдеш в 1932 – еще более простое.

Ваша задача состоит в том, чтобы решить несколько более общую задачу – а именно по числу n найти количество простых чисел p из интервала $n < p < 2n$.

Напомним, что число называется простым, если оно делится только само на себя и на единицу.

Входные данные

Входной файл INPUT.TXT содержит целое число n ($2 \leq n \leq 50000$).

Выходные данные

В выходной файл OUTPUT.TXT выведите одно число – ответ на задачу.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	2	1
2	239	39
3	3000	353

Задача №37

Сжимающий оператор

(Время: 1 сек. Память: 16 Мб Сложность: 34%)

Оператором A , действующим из множества X в множество Y (или просто оператором из X в Y) называется правило, согласно которому каждому элементу x множества X сопоставляется элемент $y=Ax$ из множества Y . Пусть X и Y – множества точек на плоскости. Оператор A из X в Y называется сжимающим с коэффициентом q , где q – вещественное число из полуинтервала $[0, 1)$, если для любого x из X выполнено $\|Ax\| \leq q \cdot \|x\|$ (здесь $\|x\|$ - норма точки x – расстояние от x до начала координат). Проще говоря, оператор называется сжимающим с коэффициентом q если он сопоставляет каждой точке точку, которая не менее, чем в q раз ближе к началу координат.

Для заданного оператора A требуется проверить является ли он сжимающим с коэффициентом q .

Входные данные

Первая строка входного файла INPUT.TXT содержит количество точек n ($1 \leq n \leq 100$) и число q ($0 \leq q < 1$), заданное не более чем с 3 знаками после десятичной точки. Следующие n строк содержат по 4 целых числа, по модулю не превосходящих 1000, разделенные пробелами – координаты точки множества X и сопоставленной ей точки из множества Y .

Выходные данные

В выходной файл OUTPUT.TXT выведите одно слово: “Yes” если оператор является сжимающим с коэффициентом q и “No” в противном случае.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	2 0.5 0 10 5 0 10 0 0 1	Yes
2	2 0.1 0 10 5 0 10 0 0 1	No
3	2 0.9 0 0 0 0 10 0 0 1	Yes

Задача №38 Игра - 2

(Время: 1 сек. Память: 16 Мб Сложность: 60%)

Вы любите играть в игры? Конечно, любите! Но про эту игру, возможно, ничего не знаете и не слышали даже. Что ж, расскажем о новой игре. На доске написана последовательность n целых чисел. Играют двое. На очередном ходе игрок выбирает число с правого или с левого края последовательности, затем это число стирается и последовательность становится на одно число меньше, а ход переходит к противнику. Выигрывает тот, кто наберет в сумме больше. Написать программу, определяющую победителя в конкретной игре, при условии, что игроки будут играть оптимально.

Входные данные

В первой строке входного файла INPUT.TXT записано целое число n ($0 < n < 100$). Во второй строке через пробел заданы n натуральных чисел, не превосходящих 1000.

Выходные данные

В единственную строку выходного файла OUTPUT.TXT нужно вывести 1, если победит первый игрок, 2 – если победит второй игрок и 0 – в случае ничьей.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	4 3 2 5 4	1
2	6 5 5 5 5 5 5	0
3	9 2 1 3 2 9 1 2 3 1	2
4	10 2 5 3 12 4 6 13 7 1 3	1

Задача №39

Волосатый бизнес

(Время: 1 сек. Память: 16 Мб Сложность: 32%)

Одного неформала выгнали с работы, и теперь ему надо как-то зарабатывать себе на пиво и сигареты. Поразмыслив, он решил, что сможет иметь очень неплохие деньги на продаже собственных волос. Известно, что пункты приема покупают волосы произвольной длины стоимостью C у.е. за каждый сантиметр. Так как волосяной рынок является очень динамичным, то цена одного сантиметра волос меняется каждый день как и курс валют. Неформал является очень хорошим бизнес-аналитиком. Он смог вычислить, какой будет цена одного сантиметра волос в каждый из ближайших N дней (для удобства пронумеруем дни в хронологическом порядке от 0 до $N-1$). Теперь он хочет определить, в какие из этих дней ему следует продавать волосы, чтобы по истечению всех N дней заработать максимальное количество денег. Заметим, что волосы у неформала растут только ночью и вырастают на 1 сантиметр за ночь. Следует также учесть, что до 0-го дня неформал с горя подстригся наголо и к 0-му дню длина его волос составляла 1 сантиметр.

Входные данные

В первой строке входного файла INPUT.TXT записано целое число N ($0 < N \leq 100$). Во второй строке через пробел заданы N натуральных чисел, не превосходящих 100, соответствующие стоимости $C[i]$ 1 сантиметра волос за каждый i -й день.

Выходные данные

В единственную строку выходного файла OUTPUT.TXT нужно вывести максимальную денежную сумму, которую может заработать неформал за N дней.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	5 73 31 96 24 46	380
2	10 1 2 3 4 5 6 7 8 9 10	100
3	10 10 9 8 7 6 5 4 3 2 1	55

Задача №40

2^N

(Время: 1 сек. Память: 16 Мб Сложность: 30%)

Необходимо вычислить значение 2^n .

Входные данные

В единственной строке входного файла INPUT.TXT записано натуральное число n ($0 < n < 1000$).

Выходные данные

В единственную строку выходного файла OUTPUT.TXT нужно вывести значение 2^n .

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	3	8
2	10	1024
3	72	4722366482869645213696

Задача №41 Сортировка подсчетом

(Время: 2 сек. Память: 16 Мб Сложность: 29%)

На планете «Аурон» атмосфера практически отсутствует, поэтому она известна своими перепадами температур в различных точках. Известно, что эти перепады колеблются от -100 до 100 градусов. Нашим специалистам удалось выяснить значения температур в N точках этой планеты. К сожалению, эти значения вычислены с большими погрешностями, поэтому их решили округлить до целых чисел. Хотелось бы наглядно видеть участки с повышенной и пониженной температурой. Вам требуется помочь. Вы должны упорядочить температуры участков по неубыванию.

Входные данные

В первой строке входного файла INPUT.TXT задано натуральное число N - количество участков ($N \leq 10^6$). Во второй строке через пробел записаны целые значения температур этих участков, не превосходящие 100 по абсолютной величине.

Выходные данные

В единственную строку выходного файла OUTPUT.TXT нужно вывести разделенные пробелом значения температур всех известных участков, которые должны следовать друг за другом в порядке неубывания.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	3 9 -20 14	-20 9 14
2	10 12 7 92 5 18 4 32 48 11 74	4 5 7 11 12 18 32 48 74 92

Задача №42 Драконы

(Время: 1 сек. Память: 16 Мб Сложность: 40%)

Известно, что у дракона может быть несколько голов и его сила определяется числом голов. Но как определить силу драконьей стаи, в которой несколько драконов и у каждого из них определенное число голов? Вероятно, вы считаете, что это значение вычисляется как сумма всех голов? Это далеко не так, иначе было бы слишком просто вычислить силу драконьей стаи. Оказывается, что искомое значение равно произведению значений числа голов каждого из драконов. Например, если в стае 3 дракона, у которых 3, 4 и 5 голов соответственно, то сила равна $3 \cdot 4 \cdot 5 = 60$. Предположим, что нам известно суммарное значение голов драконьей стаи, как нам вычислить максимально возможное значение силы этого логова драконов? Именно эту задачу Вам и предстоит решить.

Входные данные

В единственной строке входного файла INPUT.TXT записано натуральное число N ($0 < N < 100$) – количество голов драконьей стаи.

Выходные данные

В единственную строку выходного файла OUTPUT.TXT нужно вывести максимально возможное значение силы, которая может быть у стаи драконов из N голов.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	6	9
2	8	18
3	13	108

Задача №43 Нули

(Время: 1 сек. Память: 16 Мб Сложность: 16%)

Требуется найти самую длинную непрерывную цепочку нулей в последовательности нулей и единиц.

Входные данные

В единственной строке входного файла INPUT.TXT записана последовательность нулей и единиц (без пробелов). Суммарное количество цифр не превышает 100.

Выходные данные

В единственную строку выходного файла OUTPUT.TXT нужно вывести искомую длину цепочки нулей.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	00101110000110	4

Задача №44 Стрелки

(Время: 1 сек. Память: 16 Мб Сложность: 20%)

Задана последовательность, состоящая только из символов '>', '<' и '-'. Требуется найти количество стрел, которые спрятаны в этой последовательности. Стрелы – это подстроки вида '>>-->' и '<--<<'.

Входные данные

В первой строке входного файла INPUT.TXT записана строка, состоящая из символов '>', '<' и '-' (без пробелов). Строка состоит не более, чем из 250 символов.

Выходные данные

В единственную строку выходного файла OUTPUT.TXT нужно вывести искомое количество стрелок.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	<<<<>>--><--<<--<<>>>--><<<<<	4

Задача №45 Произведение цифр

(Время: 1 сек. Память: 16 Мб Сложность: 42%)

Требуется найти наименьшее натуральное число Q такое, что произведение его цифр равно заданному числу N.

Входные данные

В единственной строке входного файла INPUT.TXT записано одно целое число N ($0 \leq N \leq 10^9$).

Выходные данные

В выходной файл OUTPUT.TXT нужно вывести искомое число Q. В том случае, если такого числа не существует, следует вывести -1.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	10	25
2	13	-1
3	90	259

Задача №46 Число E

(Время: 1 сек. Память: 16 Мб Сложность: 10%)

Выведите в выходной файл округленное до n знаков после десятичной точки число E. Число E, округленное до 25 знаков после десятичной точки, равно 2.7182818284590452353602875.

Входные данные

Входной файл INPUT.TXT содержит целое число n ($0 \leq n \leq 25$).

Выходные данные

В выходной файл OUTPUT.TXT выведите ответ на задачу.

Пример

№	INPUT.TXT	OUTPUT.TXT

1	0	3
2	25	2.7182818284590452353602875
3	13	2.7182818284590

Задача №47

Наилучший делитель

(Время: 1 сек. Память: 16 Мб Сложность: 26%)

Будем говорить, что число a лучше числа b , если сумма цифр a больше суммы цифр числа b , а в случае равенства сумм их цифр, если число a меньше числа b . Например, число 124 лучше числа 123, так как у первого из них сумма цифр равна семи, а у второго – шести. Также, число 3 лучше числа 111, так как у них равны суммы цифр, но первое из них меньше.

Дано число n . Найдите такой его делитель (само число n и единица считаются делителями числа n), который лучше любого другого делителя числа n .

Входные данные

Первая строка входного файла содержит целое число n ($1 \leq n \leq 10^5$).

Выходные данные

В выходной файл выведите ответ на задачу.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	10	5
2	239	239

Задача №48

Наихудший делитель

(Время: 1 сек. Память: 16 Мб Сложность: 23%)

Будем говорить, что число a лучше числа b , если сумма цифр a больше суммы цифр числа b , а в случае равенства сумм их цифр, если число a меньше числа b . Например, число 124 лучше числа 123, так как у первого из них сумма цифр равна семи, а у второго — шести. Также, число 3 лучше числа 111, так как у них равны суммы цифр, но первое из них меньше.

Дано число n . Найдите такой его делитель d (само число n и единица считаются делителями числа n), что любой другой делитель s числа n лучше, чем d .

Входные данные

Первая строка входного файла INPUT.TXT содержит целое число n ($1 \leq n \leq 10^{5000}$).

Выходные данные

В выходной файл OUTPUT.TXT выведите ответ на задачу.

Примеры

--	--	--

№	INPUT.TXT	OUTPUT.TXT
1	10	10
2	239	1

Задача №49

Шаблоны

(Время: 1 сек. Память: 16 Мб Сложность: 35%)

Шаблоном размера n назовем строку длины n , каждый из символов которой входит в множество $\{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, a, b, c, d, e, f, g, ?\}$. Шаблоны преобразуются в строки из цифр по следующим правилам:

- символы от 0 до 9 могут быть преобразованы только сами в себя;
- символ a может преобразован в любой из символов $0,1, 2, 3$;
- символ b может преобразован в любой из символов $1,2,3,4$;
- символ c может преобразован в любой из символов $2,3,4,5$;
- символ d может преобразован в любой из символов $3,4,5,6$;
- символ e может преобразован в любой из символов $4,5,6,7$;
- символ f может преобразован в любой из символов $5,6,7,8$;
- символ g может преобразован в любой из символов $6,7,8,9$;
- символ $?$ может преобразован в любой из символов от 0 до 9;

Даны два шаблона: p_1 и p_2 . Рассмотрим множество S_1 строк, которые могут быть получены из p_1 по описанным правилам, и множество S_2 строк, которые могут быть получены из p_2 . Необходимо найти количество строк, входящих в оба этих множества.

Входные данные

Первая строка входного файла INPUT.TXT содержит шаблон p_1 , вторая — шаблон p_2 . Шаблоны имеют одинаковый положительный размер, не больше 9.

Выходные данные

В выходной файл OUTPUT.TXT выведите ответ на задачу.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	??? abc	64
2	??? 000	1
3	abc 999	0

Задача №50

Строки

(Время: 1 сек. Память: 16 Мб Сложность: 44%)

Циклическим сдвигом строки s называется строка $s_k s_{k+1} s_{k+2} \dots s_{|s|} s_1 s_2 \dots s_{k-1}$ для некоторого k , здесь $|s|$ - длина строки s . Подстрокой строки s называется строка $s_i s_{i+1} \dots s_{j-1} s_j$ для некоторых i и j . Вам даны две строки a и b . Выведите количество подстрок строки a , являющихся циклическими

сдвигами строки b.

Входные данные

Первая строка входного файла INPUT.TXT содержит строку a ($1 \leq |a| \leq 1000$). Во второй строке входного файла записана строка b ($1 \leq |b| \leq \min(100, |a|)$). Обе строки состоят только из символов латинского алфавита и цифр.

Выходные данные

В выходной файл OUTPUT.TXT выведите целое число – ответ на задачу.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	abcabc abc	4
2	abcabc acb	0
3	aaaaaaaa aa	6
4	aAaa8aaAa aAa	4

Задача №51 Факториалы!!!

(Время: 1 сек. Память: 16 Мб Сложность: 26%)

Определение 1:

$n!!...! = n(n-k)(n-2k) \dots (n \bmod k)$, если n не делится на k,
 $n!!...! = n(n-k)(n-2k) \dots k$, если n делится на k (знаков ! в обоих случаях k штук).

Определение 2:

$X \bmod Y$ — остаток от деления X на Y.
Например, $10 \bmod 3 = 1$; $3! = 3 \cdot 2 \cdot 1$; $10!!! = 10 \cdot 7 \cdot 4 \cdot 1$;

Мы по заданным n и k смогли вычислить значение выражения из определения 1. А вам слабо?

Входные данные

Во входном файле INPUT.TXT содержится ровно одна строка. Сначала – целое число n, ($1 \leq n \leq 10$), затем ровно один пробел, затем k восклицательных знаков ($1 \leq k \leq 20$).

Выходные данные

В выходной файл OUTPUT.TXT выведите одно число – значение $n!!...!$.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	10 !!!	280

2	9 !!	945
3	3 !	6

Задача №52
Счастливым билет

(Время: 1 сек. Память: 16 Мб Сложность: 12%)

Вы пользуетесь общественным транспортом? Вероятно, вы расплачивались за проезд и получали билет с номером. Счастливым билетом называют такой билет с шестизначным номером, где сумма первых трех цифр равна сумме последних трех. Т.е. билет с номером 385916 – счастливый, т.к. 3+8+5=9+1+6. Вам требуется написать программу, которая проверяет счастливость билета.

Входные данные

В единственной строке входного файла INPUT.TXT записано одно целое число N ($0 \leq N < 10^6$).

Выходные данные

В выходной файл OUTPUT.TXT нужно вывести «YES», если билет с номером N счастливый и «NO» в противном случае.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	385916	YES
2	123456	NO

Задача №53
Раскраска таблицы умножения

(Время: 1 сек. Память: 16 Мб Сложность: 32%)

Таблицей умножения назовем таблицу размера n строк на m столбцов, в которой на пересечении i-ой строки и j-ого столбца стоит число $i \cdot j$ (строки и столбцы нумеруются с единицы).

В одной из математических школ было решено провести педагогический эксперимент. Для того, чтобы ученикам было проще запоминать таблицу умножения, некоторые числа в ней будут покрашены в красный, некоторые - в синий, а некоторые - в зеленый цвет (оставшиеся числа будут черными).

Процесс покраски чисел можно условно разбить на четыре этапа. На первом этапе все числа красятся в черный цвет. На втором - все четные числа красятся в красный цвет, на третьем – все числа, делящиеся на 3, красятся в зеленый цвет, на четвертом - все числа, делящиеся на 5, красятся в синий цвет.

Директор школы хочет знать, какое количество картриджей для принтеров необходимо закупить для печати таблиц. Поэтому ему необходима информация о том, сколько чисел какого цвета будет в одной раскрашенной таблице умножения n на m. Напишите программу, решающую задачу подсчета соответствующих количеств.

Входные данные

Входной файл INPUT.TXT содержит два натуральных числа n и m ($1 \leq n, m \leq 1000$).

Выходные данные

В первой строке выходного файла OUTPUT.TXT выведите количество чисел, покрашенных в красный цвет, во второй - в зеленый, в третьей - в синий, в четвертой - в черный. Следуйте формату, приведенному в примерах.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	10 10	RED : 21 GREEN : 39 BLUE : 36 BLACK : 4
2	5 2	RED : 5 GREEN : 2 BLUE : 2 BLACK : 1

Задача №54 Теория игр

(Время: 1 сек. Память: 16 Мб Сложность: 28%)

Одним из интересных объектов, изучаемых в теории игр, являются так называемые антагонистические игры двух лиц. Такие игры характеризуются множеством X стратегий первого игрока, множеством Y стратегий второго игрока и функцией выигрыша $K(x, y)$ (x из X , y из Y). Если множества стратегий X и Y конечны, то такую игру принято называть матричной, так как функцию выигрыша K в этом случае удобно задавать матрицей.

Рассмотрим матричную игру, в которой $X = \{1, \dots, n\}$, $Y = \{1, \dots, m\}$. Матрицу выигрышей обозначим символом K . Нижним значением игры назовем число $\max_{i=1..n} \min_{j=1..m} K_{ij}$. Верхним значением игры назовем число $\min_{j=1..m} \max_{i=1..n} K_{ij}$. Отметим также, что игры, у которых нижнее и верхнее значение совпадают, называются играми с седловой точкой.

Задана матрица выигрышей K для некоторой матричной игры. Найдите ее верхнее и нижнее значение.

Входные данные

Первая строка входного файла INPUT.TXT содержит целые числа n и m ($1 \leq n, m \leq 100$). Далее следуют n строк по m чисел в каждой. j -ое число i -ой строки равно K_{ij} . Все K_{ij} по модулю не превосходят 1000.

Выходные данные

В выходной файл OUTPUT.TXT выведите нижнее и верхнее значение игры через пробел.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	3 3 4 -1 -3 -2 1 3 0 2 -3	-2 2

	3 4	-1 1
2	-1 0 2 1	
	-2 0 1 0	
	2 1 -1 -2	

Задача №55

Фонарики

(Время: 1 сек. Память: 16 Мб Сложность: 31%)

«Одна голова хорошо, а две лучше. Одна лампочка хорошо, а две лучше!» - подумал Миша, и решил собрать фонарик с двумя лампочками. Теперь он хочет узнать, насколько фонарик с двумя лампочками лучше, чем фонарик с одной. Для этого Миша посветил фонариком на стену, и каждая из лампочек осветила на ней круг.

Эффективность фонарика Миша хочет оценить через площадь освещенной части стены. Миша догадался измерить координаты центров освещенных кругов и их радиусы (которые оказались одинаковыми). Причем, площадь, освещаемая фонариком с одной лампочкой известна, т.к. описана в документации, прилагаемой к фонарику. Но что делать дальше он не знает. Напишите программу, которая поможет Мише.

Входные данные

В первых двух строчках входного файла INPUT.TXT содержатся координаты (x_1, y_1) и (x_2, y_2) - центры кругов от лампочек собранного Мишей фонарика. В третьей строке задан радиус r описанных выше кругов, а четвертая строка содержит площадь освещения s фонариком из одной лампочки. Все числа целые и удовлетворяют следующим ограничениям: $1 \leq x_1, y_1, x_2, y_2, r \leq 100, 1 \leq s \leq 10^5$. Так же заметим, что площади, освещаемые разными фонариками, отличаются друг от друга более чем на 10^{-3} .

Выходные данные

В выходной файл OUTPUT.TXT выведите «YES», если Мишин фонарик лучше старого (т.е. освещает большую площадь) и «NO» в противном случае.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	1 2 3 4 2 22	YES
2	1 1 100 100 1 7	NO

Задача №56

Живой Журнал

(Время: 1 сек. Память: 16 Мб Сложность: 29%)

Программист Саша участвует в создании блог-сервиса Живой Журнал. Планируется, что этот сервис будет предоставлять гораздо больше возможностей, чем известный всем LiveJournal. В

настоящее же время проблему составляет реализация всех базовых возможностей LiveJournal'a. Одной из таких возможностей является поддержка списков друзей для пользователей.

Заданы: список пользователей, являющихся друзьями данного пользователя, и список пользователей, у которых данный пользователь содержится в списке друзей.

Необходимо получить список друзей данного пользователя (Friends), список его взаимных друзей (Mutual Friends), и список тех пользователей, у кого данный пользователь содержится в списке друзей, но которые не являются его взаимными друзьями (Also Friend of).

Входные данные

Первая строка входного файла INPUT.TXT содержит число n ($0 \leq n \leq 200$) друзей данного пользователя. Последующие n строк содержат каждая по одному имени пользователя, который является другом данного. $(n + 2)$ -ая строка содержит число m ($0 \leq m \leq 200$) пользователей, у которых данный содержится в списке друзей. Далее заданы имена пользователей, у которых данный находится в списке друзей. Эти пользователи заданы в том же формате, что и друзья данного.

Имена пользователей - строки длиной не более 20 символов, содержащие только строчные буквы латинского алфавита и символы тире ("-"). Каждый пользователь указан не более одного раза в каждом из списков.

Выходные данные

В выходной файл OUTPUT.TXT следует вывести список друзей данного пользователя (Friends), список его взаимных друзей (Mutual Friends), и список тех пользователей, у кого данный пользователь содержится в списке друзей, но которые не являются его взаимными друзьями (Also Friend of). В каждом списке пользователи должны быть отсортированы по алфавиту. Следуйте формату, приведенному в примерах.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	3 vasya-pupkin bill-hates ivan-ivanov 2 vasya-pupkin destroyer	Friends: bill-hates, ivan-ivanov, vasya-pupkin Mutual Friends: vasya-pupkin Also Friend of: destroyer
2	0 0	Friends: Mutual Friends: Also Friend of:

Задача №57 Компьютерная сеть

(Время: 1 сек. Память: 16 Мб Сложность: 33%)

Компания «Маша и медведи» является самым крупным интернет-провайдером во всем лесу. Именно поэтому, с просьбой подключить их к интернету обратились N поросят. Домики поросят расположены в различных точках (x_i, y_i) . Ближайшая точка подключения расположена в точке (x_{net}, y_{net}) .

Для того чтобы подключиться к сети всем N поросятам необходимо:

1. провести провод от точки подключения до домика одного из поросят;
2. от подключенного поросенка провести провода ко всем остальным.
- При этом провода могут при необходимости пересекаться.

Поросята платят деньги в зависимости от длины провода. Количество денег у них ограничено и составляет p тугриков. Они хотят определить: хватит ли им денег на подключение? Так же известно, что единица длины провода стоит s тугриков. Помогите им сделать необходимые расчеты!

Входные данные

В первой строке входного файла INPUT.TXT находится числа N , s и p – целые числа со следующими ограничениями: $1 \leq N \leq 10^3$, $0 \leq s \leq 10^4$, $0 \leq p \leq 10^{15}$. В следующих N строках находятся координаты домов поросят $(x_i; y_i)$. В последней строке записаны координаты точки соединения (x_{net}, y_{net}) . Все координаты целые и не превосходят 1000 по модулю.

Выходные данные

В выходной файл OUTPUT.TXT следует вывести «YES», если у поросят достаточно денег для подключения и «NO» в противном случае.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	3 2 6 0 0 1 0 0 1 -1 0	YES
2	3 1 5 1 1 2 2 3 3 4 4	NO

Задача №58

Проверка на симпатичность

(Время: 1 сек. Память: 16 Мб Сложность: 28%)

Рассмотрим таблицу, содержащую n строк и m столбцов, в каждой клетке которой расположен ноль или единица. Назовем такую таблицу симпатичной, если в ней нет ни одного квадрата 2 на 2, заполненного целиком нулями или целиком единицами.

Так, например, таблица 4 на 4, расположенная слева, является симпатичной, а расположенная справа таблица 3 на 3 - не является.

1	0	1	0
1	1	1	0
0	1	0	1
0	0	0	0

0	0	1
0	0	1
1	1	1

Задано несколько таблиц. Необходимо для каждой из них выяснить, является ли она симпатичной.

Входные данные

Первая строка входного файла INPUT.TXT содержит количество t ($1 \leq t \leq 10$) наборов входных данных. Далее следуют описания этих наборов. Описание каждого набора состоит из строки, содержащей числа n и m ($1 \leq n, m \leq 100$), и n строк, каждая из которых содержит по m чисел, разделенных пробелами. j -ое число в $i+1$ -ой строке описания набора входных данных - элемент a_{ij} соответствующей таблицы. Гарантируется, что все a_{ij} равны либо нулю, либо единице.

Выходные данные

Для каждого набора входных данных выведите в файл OUTPUT.TXT единственную строку, содержащую слово «YES», если соответствующая таблица является симпатичной, и слово «NO» - в противном случае.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	3	YES
	1 1	YES
	0	NO
	4 4	
	1 0 1 0	
	1 1 1 0	
	0 1 0 1	
	0 0 0 0	
	3 3	
	0 0 1	
	0 0 1	
	1 1 1	

Задача №59 Несложное вычисление

(Время: 1 сек. Память: 16 Мб Сложность: 25%)

Задано натуральное число n . Необходимо перевести его в k -ичную систему счисления и найти разность между произведением и суммой его цифр в этой системе счисления.

Например, пусть $n = 239$, $k = 8$. Тогда представление числа n в восьмеричной системе счисления — 357, а ответ на задачу равен $3 \times 5 \times 7 - (3 + 5 + 7) = 90$.

Входные данные

Входной файл INPUT.TXT содержит два натуральных числа: n и k ($1 \leq n \leq 10^9$, $2 \leq k \leq 10$). Оба этих числа заданы в десятичной системе счисления.

Выходные данные

В выходной файл OUTPUT.TXT выведите ответ на задачу (в десятичной системе счисления).

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	239 8	90
2	1000000000 7	-34

Задача №60

Сверхпростые числа

(Время: 1 сек. Память: 16 Мб Сложность: 39%)

Простым числом будем называть натуральное число, большее единицы и делящееся только на единицу и на само себя. Выпишем все простые числа в порядке возрастания и i -ое в этом порядке число обозначим p_i (число 2 при этом будет иметь номер 1). Так, например, $p_1 = 2$, $p_2 = 3$, $p_3 = 5$, $p_{52} = 239$.

Скажем, что число p_i является сверхпростым, если $i = p_k$ для некоторого k . Иными словами, сверхпростое число — это простое число, номер которого в списке простых чисел, упорядоченном по возрастанию, является простым числом.

Дано натуральное число k . Упорядочим все сверхпростые числа по возрастанию. Найдите k -ое сверхпростое число в этом порядке.

Входные данные

Входной файл INPUT.TXT содержит натуральное число k ($1 \leq k \leq 500$).

Выходные данные

В выходной файл OUTPUT.TXT выведите k -ое сверхпростое число.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	1	3
2	2	5
3	3	11
4	100	3911

Задача №61

Баскетбол

(Время: 1 сек. Память: 16 Мб Сложность: 5%)

Известны результаты каждой из 4х четвертей баскетбольной встречи. Нужно определить победителя матча.

Входные данные

Входной файл INPUT.TXT содержит 4 строки, в каждой строке находится два целых числа a и b — итоговый счет в соответствующей четверти. a — количество набранных очков за четверть первой командой, b — количество очков, набранных за четверть второй командой. ($0 \leq a, b \leq 100$).

Выходные данные

В выходной файл OUTPUT.TXT выведите номер выигравшей команды, в случае ничьей следует вывести «DRAW».

Примеры

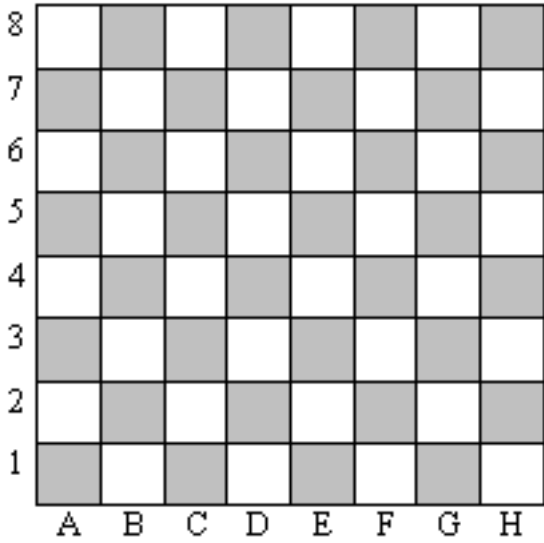
№	INPUT.TXT	OUTPUT.TXT
1	26 17 13 15 19 11 14 16	1
2	14 15 17 18 20 20 15 17	2
3	15 16 18 17 10 12 14 12	DRAW

Задача №62

Клетки

(Время: 1 сек. Память: 16 Мб Сложность: 15%)

Известно, что шахматная доска имеет размерность 8x8 и состоит из клеток 2х цветов, например, черного и белого (см. рисунок). Каждая клетка имеет координату, состоящую из буквы и цифры. Горизонтальное расположение клетки определяется буквой от А до Н, а вертикальное – цифрой от 1 до 8. Заметим, что клетка с координатой А1 имеет черный цвет. Требуется по заданной координате определить цвет клетки.



Входные данные

В единственной строке входного файла INPUT.TXT записана координата клетки на шахматной доске: всего два символа – буква и цифра (без пробелов).

Выходные данные

В выходной файл OUTPUT.TXT нужно вывести «WHITE», если указанная клетка имеет белый цвет и «BLACK», если она черная.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	C3	BLACK
2	G8	WHITE

Задача №63

Загадка

(Время: 1 сек. Память: 16 Мб Сложность: 18%)

Петя и Катя – брат и сестра. Петя – студент, а Катя – школьница. Петя помогает Кате по математике. Он задумывает два натуральных числа X и Y ($X, Y \leq 1000$), а Катя должна их отгадать. Для этого Петя делает две подсказки. Он называет сумму этих чисел S и их произведение P. Помогите Кате отгадать задуманные Петей числа.

Входные данные

Входной файл INPUT.TXT содержит два натуральных числа S и P, разделенных пробелом.

Выходные данные

В выходной файл OUTPUT.TXT выведите два числа X и Y, загаданные Петей. Числа следует вывести в порядке неубывания своих значений, разделенные пробелом.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	4 4	2 2
2	5 6	2 3

Задача №64 Простой ряд

(Время: 1 сек. Память: 16 Мб Сложность: 27%)

Простым числом называется натуральное число (больше 1), которое делится нацело только на 1 и на само себя. Например, числа 2, 3, 5, 7, 11 и 23 - простые . Назовем простым рядом последовательность цифр, полученную следующим образом: подряд идущие по возрастанию простые числа (начиная с 2) записываются друг за другом. Начало данного ряда выглядит так: 23571113171923... Необходимо найти цифру, стоящую в простом ряду на указанном месте. Нумерация позиций начинается с единицы.

Входные данные

В первой строке входного файла INPUT.TXT записано натуральное число M – количество тестов. Во второй строке записано M чисел через пробел, каждое число – номер позиции в простом ряду ($1 \leq M \leq 1000$, $1 \leq \text{номер позиции} \leq 10000$).

Выходные данные

В выходной файл OUTPUT.TXT для каждой позиции выведите цифру из простого ряда, стоящую на этой позиции. Вывести следует M цифр в одной строке, не разделяя цифры пробелами.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	3 1 4 11	271
2	5 2 5 6 8 12	31139

Задача №65 Расстояние Хэмминга

(Время: 1 сек. Память: 16 Мб Сложность: 32%)

В связи с особенностями линии связи, используемой для передачи сообщений из пункта А в пункт В, каждый бит принятого сообщения с вероятностью 0.001 содержит ошибку.

Из пункта А в пункт В было послано одно из n сообщений m_1, m_2, \dots, m_n . В пункте В было принято сообщение s .

Ваша задача заключается в определении наиболее вероятного исходного сообщения. Очевидно, что оно будет одним из тех сообщений, расстояние Хэмминга между которым и строкой s минимально.

Расстоянием Хэмминга двух строк a и b одинаковой длины называется количество позиций, в которых эти строки различаются (количество элементов в множестве $\{i \mid 1 \leq i \leq |a|, a_i \neq b_i\}$).

Входные данные

Первая строка входного файла INPUT.TXT содержит s — принятое сообщение. Вторая строка содержит целое число n — количество сообщений, которые могли быть отправлены. Следующие n строк содержат m_i — эти сообщения. Длины всех сообщений равны ($|s| = |m_1| = |m_2| = \dots = |m_n|$). Сообщения состоят только из символов 0 и 1. Размер входного файла не превосходит 60 Кб.

Выходные данные

В первую строку выходного файла OUTPUT.TXT выведите k — количество сообщений, на которых достигается минимум расстояния Хэмминга. Во вторую строку выведите в порядке возрастания k чисел — номера этих сообщений.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	010101	2
	3	2 3
	110011	
	011001	
	000111	

Задача №66 Клавиатура

(Время: 1 сек. Память: 16 Мб Сложность: 11%)

Для данной буквы латинского алфавита нужно вывести справа стоящую букву на стандартной клавиатуре. При этом клавиатура замкнута, т.е. справа от буквы «р» стоит буква «а», от буквы «l» стоит буква «z», а от буквы «m» — буква «q».

Входные данные

Входной файл INPUT.TXT содержит один символ — маленькую букву латинского алфавита.

Выходные данные

В выходной файл OUTPUT.TXT следует вывести букву стоящую справа от заданной буквы, с учетом замкнутости клавиатуры.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	q	w

2	t	y
3	p	a
4	l	z
5	m	q

Задача №67

Маска подсетей

(Время: 1 сек. Память: 16 Мб Сложность: 33%)

Рассмотрим компьютерную сеть с настроенной TCP/IP маршрутизацией. Будем рассматривать некоторую ее модификацию. А именно в этой сети находить N подсетей. Каждая подсеть характеризуется своей маской. Маска подсети представляет собой 4 однобайтных числа, разделенных точкой. Причем для масок выполнено следующее свойство: если представить маску в двоичном виде, то сначала она будет содержать k единиц, а потом q нулей, причем $k + q = 32$. Например, 255.255.255.0 — маска подсети, а 192.168.0.1 — нет.

Поясним, как получается двоичное представление IP-адреса. Для этого числа, составляющие IP-адрес, представляются в двоичной системе счисления (при этом каждое из них дополняется ведущими нулями до длины в 8 цифр), после чего удаляются точки. Получившееся 32-битное число и есть двоичное представление IP-адреса. Например, для адреса 192.168.0.1 этот процесс выглядит так: 192.168.0.1 → 11000000.10101000.00000000.00000001 → 11000000101010000000000000000001. Таким образом, двоичным представлением IP-адреса 192.168.0.1 является 11000000101010000000000000000001.

Будем говорить, что два компьютера с IP₁ и IP₂ лежат в подсети, если $IP_1 \wedge Mask = IP_2 \wedge Mask$, где Mask — маска этой подсети, а \wedge — операция побитового логического «и». IP компьютера представляет собой так же 4 однобайтных числа, разделенных точкой.

Вам даны M пар IP адресов компьютеров. Для каждой из них Вам надо определить, в скольких подсетях из заданных они лежат.

Входные данные

В первой строке входного файла INPUT.TXT записано число N — количество подсетей. В следующих N строках перечислены маски этих подсетей. В N + 2 строке находится число M ($0 \leq M \leq 10000$). В следующих M строках записаны пары IP адресов, разделенных пробелом.

Выходные данные

Для каждой пары IP адресов в отдельной строке выходного файла OUTPUT.TXT выведите количество подсетей, в которых лежат оба компьютера.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	2	1
	255.255.255.255	1
	255.255.255.0	0
	3	
	192.168.31.1 192.168.31.2	
	192.168.31.3 192.168.31.4	
	192.168.31.1 192.167.31.2	

Задача №68

Дом - Школа - Дом

(Время: 1 сек. Память: 16 Мб Сложность: 21%)

Мальчик Вася каждый день ездит на метро. Утром он едет в школу, а вечером того же дня, обратно из школы, домой. Для того, чтобы немного сэкономить, он покупает электронную смарт-карту на X поездок. Когда он хочет зайти в метро, он прикладывает карту к турникету. Если на карте осталось ненулевое количество поездок, то турникет пропускает Васю и списывает с карты одну поездку. Если же на карте не осталось поездок, то турникет не пропускает Васю, и он (Вася) вынужден купить на этой же станции новую карту на X поездок и вновь пройти через турникет.

Вася заметил, что в связи с тем, что утром метро переполнено, покупать новую карту утром накладно по времени, и он может опоздать в школу. В связи с этим он хочет понять: будет ли такой день, что с утра, поехав в школу, окажется, что у него на карточке ноль поездок.

Вася больше никуда на метро не ездит и поэтому заходит в метро только на станции около дома и на станции около школы.

Входные данные

Во входном файле INPUT.TXT содержится ровно 2 строки. В первой содержится слово «School» или «Home» в зависимости от того, где первый раз Вася купил карточку на X поездок. Во второй строке содержится натуральное число X , $1 \leq X \leq 1000$.

Выходные данные

В выходной файл OUTPUT.TXT следует вывести «Yes», если будет такой день, что утром у Васи на карточке окажется ноль поездок и «No» в противном случае.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	Home 1	Yes
2	School 2	No

Задача №69

N-угольное колесо

(Время: 1 сек. Память: 16 Мб Сложность: 28%)

На одном известном автозаводе страны N-мерики главный инженер-рационализатор внес предложение вместо круглых колес использовать колеса в форме правильных N-угольников. "При этом", — сказал он, "важным показателем качества такого колеса будет разность между радиусом описанной окружности и радиусом вписанной окружности." Причем колесо считается качественным, если его показатель качества меньше единицы.

Задано число N и длина A стороны N-угольного колеса. Необходимо определить: является ли такое колесо качественным.

Входные данные

Входной файл INPUT.TXT содержит два натуральных числа: N и A ($3 \leq N \leq 1000$, $1 \leq A \leq 1000$).

Выходные данные

В выходной файл OUTPUT.TXT выведите «YES», если это качественное колесо и «NO» в противном случае.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	3 1	YES
2	239 566	NO

Задача №70 Степень строки

(Время: 1 сек. Память: 16 Мб Сложность: 30%)

Пусть задана строка $s = s_1s_2...s_n$. Назовем ее k -ой ($k > 0$) степенью s^k строку $s^k = s_1s_2 \dots s_ns_1s_2 \dots s_n \dots s_1s_2...s_n$ (k раз). Например, третьей степенью строки abc является строка abcabcabc.

Корнем k степени из строки s называется такая строка t (если она существует), что $t^k = s$.
Ваша задача состоит в том, чтобы написать программу, находящую степень строки или корень из нее.

Входные данные

Первая строка входного файла INPUT.TXT содержит строку s , она содержит только маленькие буквы латинского алфавита и имеет ненулевую длину, не превосходящую 1000.
Вторая строка входного файла содержит целое число $k \neq 0$, $|k| < 100001$. Если $k > 0$, то необходимо найти k -ую степень строки s , если $k < 0$, то необходимо найти корень степени $|k|$ из s .

Выходные данные

В выходной файл OUTPUT.TXT выведите строку, являющуюся ответом на задачу. Если длина ответа превосходит 1023 символа, выведите только первые 1023 символа. Если искомой строки не существует — выведите NO SOLUTION.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	abc 3	abcabcabc
2	abcdabcd -2	abcd
3	abcd -4	NO SOLUTION

Задача №71 Две кучки камней

(Время: 1 сек. Память: 16 Мб Сложность: 43%)

У Вас есть N камней с массами W_1, W_2, \dots, W_N . Требуется разложить камни на 2 кучки так, чтобы разница масс этих кучек была минимальной.

Входные данные

В первой строке входного файла INPUT.TXT записано число N – количество камней ($1 \leq N \leq 18$). Во второй строке через пробел перечислены массы камней W_1, W_2, \dots, W_N ($1 \leq W_i \leq 10^5$).

Выходные данные

В единственную строку выходного файла OUTPUT.TXT нужно вывести одно неотрицательное целое число – минимально возможную разницу между массами двух кучек.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	5 5 8 13 27 14	3

Задача №72 Анаграмма

(Время: 1 сек. Память: 16 Мб Сложность: 45%)

Пусть задано некоторое слово, состоящее из букв английского алфавита длиной не более 80 символов (например, “WORD”). Рассмотрим набор возможных перестановок, состоящих из букв данного слова (например, “RDOW”, “WODR” и т.д.). Требуется выбрать из этого множества слово, следующее по алфавиту за исходным.

Входные данные

В единственной строке входного файла INPUT.TXT записано слово, не последнее по алфавиту среди возможных его перестановок.

Выходные данные

В единственную строку выходного файла OUTPUT.TXT нужно вывести следующее слово по алфавиту.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	abdc	acbd
2	word	wrdo

Задача №73 Расшифровка

(Время: 1 сек. Память: 16 Мб Сложность: 28%)

Рассмотрим работу простейшего шифра. Шифруемое сообщение состоит из латинских букв, записанных в нижнем регистре и символа пробела. Шифрование происходит посимвольно. Каждой букве ставим в соответствие число: a – 1, b – 2, ... , z – 26, ‘ ‘ – 27. Далее индекс символа

складывается с номером в сообщении по модулю 27, а результат сложения представляется в системе счисления с основанием 27 (0, 1, ..., Q в верхнем регистре).

Необходимо написать дешифратор.

Входные данные

В единственной строке входного файла INPUT.TXT содержится закодированная строка, длиной не более 255 символов. Строка записана в верхнем регистре.

Выходные данные

В единственную строку выходного файла OUTPUT.TXT нужно вывести расшифровку заданной строки, при этом символы латинского алфавита следует выводить в нижнем регистре.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	L7MO	test
2	576J9FLF	decoding

Задача №74 Прыжки с шестом

(Время: 1 сек. Память: 16 Мб Сложность: 38%)

В соревнованиях по прыжкам с шестом было замечено одно интересное явление: на очередном этапе соревнований успешные и неуспешные попытки прыжков чередовались: успешный, неуспешный, успешный, неуспешный и т.д. (первый был успешным). Спортсменам разрешалась только одна попытка. Тот, кто преодолевал планку, переходил в следующий тур (этап), а тот, кто делал неудачную попытку – выбывал из соревнований. Таким образом, первым выбывал всегда спортсмен с номером 2, а последним – победитель с номером 1.

Требуется написать программу, которая по количеству участников и номеру спортсмена вычислит, каким по счету данный спортсмен выбыл из соревнований.

Входные данные

В единственной строке входного файла INPUT.TXT содержатся два натуральных числа: общее число спортсменов N и порядковый номер спортсмена в стартовом списке M. Числа разделены пробелом ($1 \leq M, N \leq 10^9$).

Выходные данные

В единственную строку выходного файла OUTPUT.TXT нужно вывести каким по счету спортсмен M выбыл из соревнований. Если это победитель состязания, то выводится число N.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	4 2	1
2	4 1	4
3	9 5	7

Задача №75

Сумма произведений

(Время: 1 сек. Память: 16 Мб Сложность: 65%)

Требуется вычислить сумму произведений цифр каждого N-значного числа. При этом следует учесть, что если в числе встречается цифра 0, то произведение его цифр равно нулю. Для N=3 искомая сумма представлена следующим рядом:

$$S = 1*0*0 + 1*0*1 + 1*0*2 + \dots + 9*9*8 + 9*9*9 = 91125$$

Входные данные

В единственной строке входного файла INPUT.TXT записано натуральное число N (N < 1000).

Выходные данные

В единственную строку выходного файла OUTPUT.TXT нужно вывести одно целое число — сумму произведений всех N-значных чисел.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	1	45
2	3	91125
3	5	184528125

Задача №76

Музей

(Время: 1 сек. Память: 16 Мб Сложность: 50%)

В музее регистрируется в течение суток время прихода и ухода каждого посетителя. Таким образом, за день получены N пар значений, где первое значение в паре показывает время прихода посетителя и второе значение - время его ухода. Требуется найти максимальное число посетителей, которые находились в музее одновременно.

Входные данные

В первой строке входного файла INPUT.TXT записано натуральное число N (N < 10⁵) – количество зафиксированных посетителей в музее в течении суток. Далее, идут N строк с информацией о времени визитов посетителей: в каждой строке располагается отрезок времени посещения в формате «ЧЧ:ММ ЧЧ:ММ» (00:00 ≤ ЧЧ:ММ ≤ 23:59).

Выходные данные

В единственную строку выходного файла OUTPUT.TXT нужно вывести одно целое число — максимальное количество посетителей, одновременно находящихся в музее.

Пример

№	INPUT.TXT	OUTPUT.TXT
	6	4

1	09:00	10:07
	10:20	11:35
	12:00	17:00
	11:00	11:30
	11:20	12:30
	11:30	18:15

Задача №77

Нолики

(Время: 1 сек. Память: 16 Мб Сложность: 63%)

Для заданных натуральных чисел N и K требуется вычислить количество чисел от 1 до N, имеющих в двоичной записи ровно K нулей.

Например, если N=8 и K=1, то мы можем записать все числа от 1 до 8 в двоичной системе счисления:

1, 10, 11, 100, 101, 110, 111 и 1000.

Откуда видно, что только числа 10, 101 и 110 имеют ровно один ноль в записи, т.е. правильный ответ – 3.

Входные данные

В единственной строке входного файла INPUT.TXT записано два натуральных числа через пробел N и K, не превышающих 10^9 .

Выходные данные

В единственную строку выходного файла OUTPUT.TXT нужно вывести одно целое число — количество чисел от 1 до N с K нулями в двоичном представлении.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	8 1	3
2	13 2	4
3	1000 5	210

Задача №78

Бутылки

(Время: 1 сек. Память: 16 Мб Сложность: 48%)

Группа программистов собралась в понедельник и на все свои деньги купила «Sprite» в бутылках емкостью по 0.25 л., не забыв взять сдачу.

Во вторник они сдали пустую посуду, добавили оставшуюся сдачу и вновь купили столько таких же бутылок «Sprite», сколько могли.

Так они действовали до пятницы. В пятницу, сдав посуду и добавив сдачу с четверга, они смогли купить только одну бутылку напитка. При этом денег у них уже не осталось.

Требуется написать программу, определяющую минимальную сумму, которой располагали программисты в понедельник.

Входные данные

Входной файл INPUT.TXT состоит из единственной строки, содержащей два целых числа F (стоимость одной бутылки «Sprite») и P (стоимость одной пустой бутылки из под «Sprite»), разделенных пробелом.

Ограничения: $1 \leq P < F \leq 10^9$, начальная сумма не превосходит $2 \cdot 10^9$.

Выходные данные

В единственную строку выходного файла OUTPUT.TXT нужно вывести одно целое число – минимальную сумму, которой располагали программисты в понедельник.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	7 3	83

Задача №79 Последняя цифра A^B

(Время: 1 сек. Память: 16 Мб Сложность: 21%)

Требуется написать программу, которая находит цифру, на которую оканчивается число A^B .

Входные данные

Входной файл INPUT.TXT состоит из единственной строки, содержащей два целых числа A и B, разделенных пробелом ($1 \leq A, B \leq 10000$).

Выходные данные

В единственную строку выходного файла OUTPUT.TXT нужно вывести цифру, на которую оканчивается A^B .

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	2 2	4
2	3 7	7
3	24 9	4

Задача №80 Тождество

(Время: 1 сек. Память: 16 Мб Сложность: 26%)

Вам необходимо проверить домашнюю работу Васи Пупкина, в которой он написал равенство. Например, запись вида «2+3=5» является правильной, а «23*7=421» неверная, но корректная. Корректной записью выражения будем называть последовательность: число, операция («+», «-», «*», «/»), число, знак равенства, число. Т.е. если в записи не хватает цифр или же встречается неизвестный символ. Например, записи «2*=3», «173» и «2+2=a» некорректны.

Входные данные

Входной файл INPUT.TXT состоит из единственной строки, содержащей запись арифметического выражения. Все числа в записи целые и не превышают по абсолютной величине 30000. Длина арифметического выражения не превышает 100 символов.

Выходные данные

В выходной файл OUTPUT.TXT нужно вывести «YES», если указанная запись правильна (т.е. равенство представляет собой тождество), «NO» - если корректная, но неверная и «ERROR», если в записи присутствуют ошибки.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	2+3=5	YES
2	3*7=20	NO
3	two plus three is five	ERROR

Задача №81 Арбузы

(Время: 1 сек. Память: 16 Мб Сложность: 14%)

Иван Васильевич пришел на рынок и решил купить два арбуза: один для себя, а другой для тещи. Понятно, что для себя нужно выбрать арбуз потяжелей, а для тещи полегче. Но вот незадача: арбузов слишком много и он не знает как же выбрать самый легкий и самый тяжелый арбуз? Помогите ему!

Входные данные

В первой строке входного файла INPUT.TXT задано одно число N – количество арбузов. Вторая строка содержит N чисел, записанных через пробел. Здесь каждое число – это масса соответствующего арбуза. Все числа натуральные и не превышают 30000.

Выходные данные

В выходной файл OUTPUT.TXT нужно вывести два числа через пробел: массу арбуза, который Иван Васильевич купит теще и массу арбуза, который он купит себе.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	5 5 1 6 5 9	1 9

Задача №82 Пересечение множеств

(Время: 1 сек. Память: 16 Мб Сложность: 34%)

Даны два неупорядоченных набора целых чисел (может быть, с повторениями). Выдать без повторений в порядке возрастания все те числа, которые встречаются в обоих наборах.

Входные данные

В первой строке входного файла INPUT.TXT записано через пробел два целых числа N и M ($1 \leq N, M \leq 10^6$) — количество элементов первого и второго наборов, соответственно. В следующих строках записано сначала N чисел первого набора, а затем M чисел второго набора. Числа разделены пробелами или символами конца строки. Каждое из этих чисел попадает в промежуток от 0 до 10^5 .

Выходные данные

В выходной файл OUTPUT.TXT нужно записать в возрастающем порядке без повторений все числа, которые входят как в первый, так и во второй набор. Числа разделять одним пробелом. Если таких чисел нет, то выходной файл должен оставаться пустым.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	11 6 2 4 6 8 10 12 10 8 6 4 2 3 6 9 12 15 18	6 12

Задача №83 Симпатичные узоры

(Время: 1 сек. Память: 16 Мб Сложность: 66%)

Компания BrokenTiles планирует заняться выкладыванием во дворах у состоятельных клиентов узор из черных и белых плиток, каждая из которых имеет размер 1x1 метр. Известно, что дворы всех состоятельных людей имеют наиболее модную на сегодня форму прямоугольника MxN метров.

Однако при составлении финансового плана у директора этой организации появилось целых две серьезных проблемы: во первых, каждый новый клиент очевидно захочет, чтобы узор, выложенный у него во дворе, отличался от узоров всех остальных клиентов этой фирмы, а во вторых, этот узор должен быть симпатичным. Как показало исследование, узор является симпатичным, если в нем нигде не встречается квадрата 2x2 метра, полностью покрытого плитками одного цвета. На рисунке 1 показаны примеры различных симпатичных узоров, а на рисунке 2 – несимпатичных.

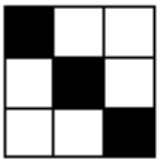


Рисунок 1

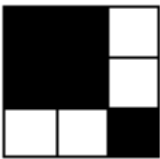
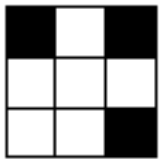
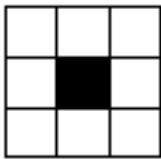
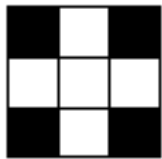
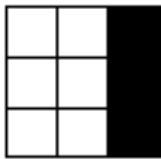


Рисунок 2



Для составления финансового плана директору необходимо узнать, сколько клиентов он сможет обслужить, прежде чем симпатичные узоры данного размера закончатся. Помогите ему!

Входные данные

В первой строке входного файла INPUT.TXT находятся два положительных целых числа, разделенные пробелом – M и N ($1 \leq M \cdot N \leq 30$).

Выходные данные

Выведите в выходной файл OUTPUT.TXT единственное число – количество различных симпатичных узоров, которые можно выложить во дворе размера MxN. Узоры, получающиеся друг из друга сдвигом, поворотом или отражением считаются различными.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	2 2	14
2	3 3	322

Задача №84 Выпуклая оболочка

(Время: 1 сек. Память: 16 Мб Сложность: 27%)

Рассмотрим бесконечный лист клетчатой бумаги. Закрасим некоторое множество клеток в черный цвет. Теперь мы хотим закрасить минимальное количество клеток, так, чтобы множество черных клеток стало выпуклым.

Напомним, что геометрическая фигура Φ называется выпуклой, если для любых точек A из Φ и B из Φ с вещественными координатами отрезок $[AB]$ принадлежит Φ .

Входные данные

В первой строке входного файла INPUT.TXT содержатся два числа N и M ($1 \leq N, M \leq 100$) — размеры куска бумаги, куда попали все черные клетки. В каждой из следующих N строк содержится M символов «*» или «.». Символ «*» обозначает черную клетку, а «.» белую.

Выходные данные

В выходной файл OUTPUT.TXT выведите выпуклое множество, содержащее минимальное количество дополнительно покрашенных черных клеток, в ровно N строках по M символов «*» или «.» в каждой.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	2 4 ..*. .**.	.*.*. .*.*.
2	4 3 .*. .*. .*. .*. .*.	.*. .*. .*. .*. .*.

Задача №85 Единичный НОД

(Время: 1 сек. Память: 16 Мб Сложность: 23%)

Заданы два натуральных числа в десятичной системе счисления, состоящие из единиц. В первом числе ровно N единиц, а во втором их ровно M . Требуется найти НОД этих чисел.

Напомним, что НОД (наибольший общий делитель) двух чисел a и b — это такое максимальное число c , что b делится на c и a делится на c .

Входные данные

В единственной строке входного файла INPUT.TXT записаны два целых числа N и M ($1 \leq N, M \leq 2000$).

Выходные данные

В выходной файл OUTPUT.TXT выведите ответ без ведущих нулей.

Примеры

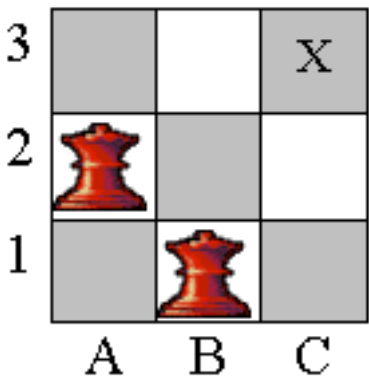
№	INPUT.TXT	OUTPUT.TXT
1	1 1	1
2	1 2	1

Задача №86 Головоломка про ферзей

(Время: 1 сек. Память: 16 Мб Сложность: 15%)

Вероятно, что многие из вас играли в шахматы. Поэтому вы знаете, что ферзь может двигаться как по диагоналям, так и по горизонталям.

Вася недавно начал заниматься шахматами и где-то прочел головоломку, в которой нужно было расставить максимальное количество ферзей на доске 8x8 так, чтобы хотя бы одно поле оказалось небитым. Эта задача легко решается для доски 3x3, т.к. понятно, что более двух ферзей расставить таким образом на ней невозможно.



Помогите Васе решить эту задачу для доски NxN.

Входные данные

В единственной строке входного файла INPUT.TXT записано натуральное число N – размеры шахматной доски NxN ($1 \leq N \leq 100$).

Выходные данные

В единственную строку выходного файла OUTPUT.TXT нужно вывести максимальное количество ферзей, которых можно расставить на шахматной доске NxN так, чтобы одна клетка оставалась небитой.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	3	2

Задача №87 Строки - 2

(Время: 1 сек. Память: 16 Мб Сложность: 32%)

Задан набор строк $S = \{s_1, s_2, s_3, \dots, s_n\}$. Необходимо найти количество строк s_i из S, представимых в виде конкатенации двух строк s_j и s_k из S ($s_i = s_j s_k$, j и k при этом могут совпадать).

Входные данные

Входной файл INPUT.TXT содержит набор строк S – по одному элементу на строке. i -ая строка входного файла содержит s_i . Последняя строка входного файла содержит строку ENDOFINPUT. Она обозначает конец входных данных и не входит в S .

Все s_i состоят только из маленьких букв латинского алфавита и имеют длину не более 100 символов. Во входном файле не более 239 строк.

Выходные данные

В выходной файл OUTPUT.TXT выведите ответ без ведущих нулей.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	aa aaaa ab abaa ENDOFINPUT	2
2	abc bcd def ENDOFINPUT	0

Задача №88 Судоку

(Время: 1 сек. Память: 16 Мб Сложность: 27%)

Судоку размера n называется квадрат со стороной n^2 , разделенный на n^2 средних квадратов со стороной n , каждый из которых разделен на n^2 маленьких квадратов. В каждом маленьком квадрате записано число от 1 до n^2 .

Судоку называется правильным, если в каждом столбце, каждой строке и каждом среднем квадрате встречаются все числа от 1 до n^2 .

Недавно Вася нарисовал Судоку размера n . Ваша задача – помочь ему определить правильный ли он.

Входные данные

В первой строке входного файла INPUT.TXT содержится число n ($1 \leq n \leq 10$). В следующих n^2 строчках содержится по n^2 чисел, задающих нарисованный Васей Судоку.

Все числа во входном файле целые и не превосходят 100 по модулю.

Выходные данные

Если Судоку правильный, то выведите в выходной файл OUTPUT.TXT слово «Correct», иначе выведите «Incorrect».

Примеры

№	INPUT.TXT	OUTPUT.TXT
---	-----------	------------

1	3 1 3 2 5 4 6 9 8 7 4 6 5 8 7 9 3 2 1 7 9 8 2 1 3 6 5 4 9 2 1 4 3 5 8 7 6 3 5 4 7 6 8 2 1 9 6 8 7 1 9 2 5 4 3 5 7 6 9 8 1 4 3 2 2 4 3 6 5 7 1 9 8 8 1 9 3 2 4 7 6 5	Correct
2	1 10	Incorrect

Задача №89

Быстрый поезд

(Время: 1 сек. Память: 16 Мб Сложность: 25%)

Между двумя крупнейшими городами нашей страны Санкт-Петербургом и Москвой ежедневно совершают рейсы n поездов. Для каждого поезда известно его время отправления из Санкт-Петербурга и время прибытия в Москву.

Найдите самый быстрый поезд и его скорость в предположении, что длина железной дороги между Санкт-Петербургом и Москвой равна 650 км.

Входные данные

Первая строка входного файла INPUT.TXT содержит целое число n ($1 \leq n \leq 100$). Каждая из последующих n строк описывает ровно один поезд.

Описание поезда состоит из его названия, времени отправления и времени прибытия. Название поезда – строка длиной не более 50 символов, заключенная в кавычки. Она может содержать буквы латинского алфавита, пробелы, цифры, символы тире («–») и подчеркивания («_»). Время отправления и прибытия заданы в формате ЧЧ:ММ. Строчные и заглавные буквы в названиях поездов различаются.

Время в пути для каждого из поездов составляет хотя бы одну минуту и не превышает 24 часов.

Гарантируется, что самый быстрый поезд определяется единственным образом.

Выходные данные

В выходной файл OUTPUT.TXT выведите название самого быстрого поезда и его скорость. Скорость выводите в километрах в час и округляйте до целых. Следуйте формату вывода, приведенному в примерах.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	3 "ER-200" 06:43 10:40 "Red Arrow" 23:55 07:55 "Express" 23:59 08:00	The fastest train is "ER-200". It's speed is 165 km/h, approximately.
2	3 "Train1" 00:00 00:00	The fastest train is "Train2". It's speed is 39000 km/h, approximately.

	"Train2" 00:00 00:01 "Train3" 00:01 00:01	
3	2 "Slow Train 1" 10:00 09:59 "Slow Train 2" 10:00 10:00	The fastest train is "Slow Train 1". It's speed is 27 km/h, approximately.

Задача №90

Треугольные страны

(Время: 1 сек. Память: 16 Мб Сложность: 37%)

Эта история происходила на одной плоской планете. С незапамятных времен на ней существовал город N, находящийся в точке x_n, y_n . Кроме этого, в разное время на этой же планете существовали страны, каждая из которых имела форму треугольника.

Теперь перед историками встала серьезная задача — по имеющимся у них данным о треугольных странах определить, в какие страны мог входить город N. Город мог входить в страну, если он находится строго внутри нее.

Входные данные

Первая строка входного файла содержит два числа: x_n и y_n — координаты города N. Вторая строка входного файла содержит количество k треугольных стран ($1 \leq k \leq 1000$). Последующие k строк каждая описывают одну треугольную страну. Описание треугольной страны состоит из шести целых чисел $x_1, y_1, x_2, y_2, x_3, y_3$, где $(x_1, y_1), (x_2, y_2), (x_3, y_3)$ — координаты вершин этой страны.

Гарантируется, что все страны имеют ненулевую площадь. Все координаты не превосходят 10000 по абсолютной величине.

Выходные данные

В первой строке выходного файла выведите количество стран, в которые мог входить город N. Во второй строке выведите через пробел номера этих стран в возрастающем порядке. Страны нумеруются с единицы в том порядке, в каком они заданы во входном файле.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	0 1 2 -2 0 2 0 0 2 -3 0 3 0 0 3	2 1 2
2	0 2 2 -2 0 2 0 0 2 -3 0 3 0 0 3	1 2

Задача №91

Две последовательности

(Время: 1 сек. Память: 16 Мб Сложность: 29%)

Определим последовательности a_n и b_n следующим образом: $a_1 = 2, a_2 = 3, a_3 = 4, a_4 = 7, a_n = b_{n-1} + b_{n-3}, n > 4, b_n$ — последовательность чисел, не входящих в a_n , записанных в возрастающем

порядке.

Таким образом, последовательность a_n будет выглядеть следующим образом: 2, 3, 4, 7, 13, 15,..., а последовательность b_n – 1, 5, 6, 8, 9, 10,....

Ваша задача состоит в том, чтобы найти a_n и b_n .

Входные данные

Входной файл содержит целое число n ($1 \leq n \leq 10000$).

Выходные данные

В первой строке выходного файла выведите a_n , во второй – b_n .

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	4	7 8
2	10	25 16
3	6578	19731 9868

Задача №92 Журавлики

(Время: 1 сек. Память: 16 Мб Сложность: 7%)

Петя, Катя и Сережа делают из бумаги журавликов. Вместе они сделали S журавликов. Сколько журавликов сделал каждый ребенок, если известно, что Петя и Сережа сделали одинаковое количество журавликов, а Катя сделала в два раза больше журавликов, чем Петя и Сережа вместе?

Входные данные

В единственной строке входного файла INPUT.TXT записано одно натуральное число S – общее количество сделанных журавликов ($S < 10^6$).

Выходные данные

В единственную строку выходного файла OUTPUT.TXT нужно вывести три числа, разделенных пробелами – количество журавликов, которые сделал каждый ребенок (Петя, Катя и Сережа).

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	6	1 4 1
2	24	4 16 4
3	60	10 40 10

Задача №93

Боги

(Время: 1 сек. Память: 16 Мб Сложность: 26%)

Археологами найден набор древних копий старинных манускриптов с мифами – различными историями о древних богах. К несчастью, переписчики этих манускриптов не отличались особой грамотностью и умудрились в каждом имени сделать ровно по одной орфографической ошибке – т.е. ровно одну из букв божественного имени заменили какой-то другой буквой. Археологи смогли составить список правильных написаний имен богов, так же им удалось выписать из манускриптов все имена собственные. Однако сопоставлять два списка – выше их сил. Помогите им в этом!

Входные данные

Первая строка входного файла INPUT.TXT содержит число N – количество имен богов в списке. Следующие N строк – имена богов. Далее идет строка, содержащая число M – количество «подозрительных» слов, выписанных из манускриптов. Следующие M строк – «подозрительные» слова. Каждое из имен богов и «подозрительных» слов – последовательность из K заглавных букв английского алфавита ($1 \leq N, M, K \leq 30$).

Выходные данные

В выходной файл OUTPUT.TXT выводится N чисел – для каждого божьего имени выводится число “подозрительных” слов, которые являются именем данного бога с одной ошибкой.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	3 ZEUS POSEIDON AFINA 4 ZEVS POSEYDON AVYNA ZERS	2 1 0

Задача №94

Принц и дракон

(Время: 1 сек. Память: 16 Мб Сложность: 22%)

Волшебник Мерлин продает волшебные мечи принцам, желающим убить дракона. Основная характеристика меча – число драконьих голов, которые он срубает за удар. Основная характеристика дракона – число голов, которые он может отрастить за сеанс регенерации. Бои принцев с драконами всегда протекают одинаково – принц атакует, и прячется за щитом; дракон атакует огненным дыханием и регенерирует; так продолжается до тех пор, пока после очередного удара у дракона не кончатся головы. Ясно, впрочем, что не каждым мечом можно победить каждого дракона. Заказ, поступающий Мерлину, всегда содержит число голов дракона и скорость его регенерации. Подсчитайте по известной атакующей силе меча, сможет ли принц убить такого дракона таким мечом и, если да, то сколько ударов потребуется.

Входные данные

Единственная строка входного файла INPUT.TXT содержит число N – число голов, которые меч срубает одним ударом. Далее идет число M – число голов дракона. За ним идет K – число голов, которые дракон регенерирует за раз ($1 \leq N, M, K \leq 10^5$). Все числа разделены пробелом.

Выходные данные

В выходной файл OUTPUT.TXT выведите число ударов, которые необходимо нанести принцу, чтобы убить дракона, если это возможно. Если таким мечом убить дракона нельзя, то следует вывести «NO».

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	3 6 2	4
2	4 4 5	1
3	5 10 6	NO

Задача №95 Нумеролог

(Время: 1 сек. Память: 16 Мб Сложность: 24%)

Чтобы предсказать судьбу человека, нумеролог берет время жизни человека в секундах, затем складывает все цифры этого числа. Если полученное число состоит более чем из одной цифры, операция повторяется, пока в числе не останется одна цифра. Затем по полученной цифре и числу операций, необходимых для преобразования числа в цифру нумеролог предсказывает судьбу человека. Нумеролог плохо умеет считать, а числа, с которыми он работает, могут быть очень большими. Напишите программу, которая бы делала все расчеты за него.

Входные данные

Входной файл INPUT.TXT содержит число N – время жизни человека в секундах ($1 \leq N \leq 10^{1000}$).

Выходные данные

В выходной файл OUTPUT.TXT выведите два числа через пробел: полученную цифру из числа N и число преобразований.

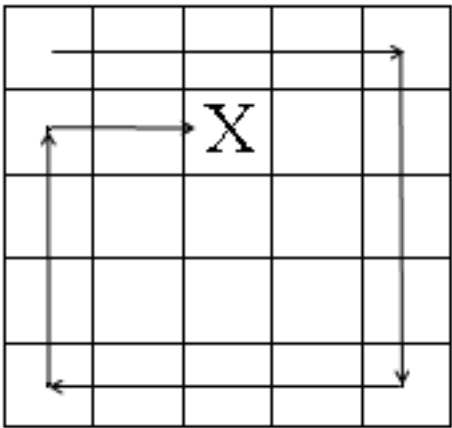
Примеры

№	INPUT.TXT	OUTPUT.TXT
1	1	1 0
2	10	1 1
3	99	9 2

Задача №96 Винни-пух

(Время: 1 сек. Память: 16 Мб Сложность: 41%)

Вини-Пух стоит на прямоугольном поле размером $N \times M$ клеток. В каждой клетке растет по одной ягоде. В начальный момент времени он стоит на левой верхней клетке. Он начинает собирать ягоды по верхнему краю поля. Если он доходит до края поля или до пустой клетки, он поворачивается на 90 градусов вправо и продолжает собирать ягоды. Но дойдя до очередной клетки Вини вспоминает, что его ждет Пятачок, и он уходит с поля.



Входные данные

В первой строке входного файла INPUT.TXT стоят размеры поляны N и M ($0 < N, M \leq 100$) – высота и ширина, во второй числа Y и X ($0 < Y \leq N, 0 < X \leq M$) – номера строки и столбца клетки, дойдя до которой Вини-Пух прекращает собирать ягоды.

Выходные данные

В выходной файл OUTPUT.TXT выведите число ягод, которые соберет Пух.

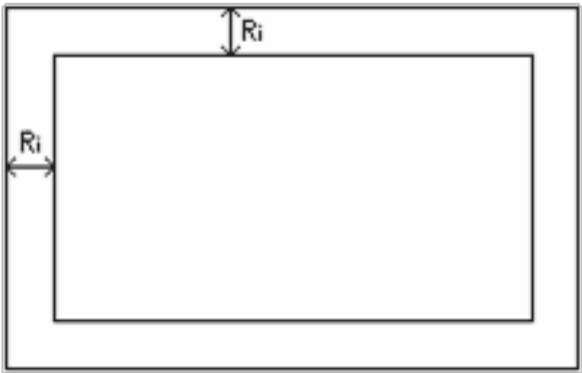
Примеры

№	INPUT.TXT	OUTPUT.TXT
1	1 1 1 1	1
2	3 3 2 3	4
3	5 5 2 3	18

Задача №97 Заповедники

(Время: 1 сек. Память: 16 Мб Сложность: 60%)

В райской долине расположены N заповедников, имеющих форму прямоугольников. Однажды на собрании директоров было принято решение об увеличении площадей заповедников. Для этого директор каждого заповедника выбрал R_i – количество метров, на которое он хочет увеличить зону своего заповедника, смотрите рисунок. Однако после подписания соглашения выяснилось, что некоторые заповедники имеют общие земли. Такие заповедники было решено объединить в один, если объединенный заповедник пересекался с еще каким-нибудь заповедником их опять объединяли и так до тех пор пока не остались заповедник(и) не имеющие общих земель.



Ваша задача посчитать, сколько заповедников стало в долине после объединения.

Входные данные

Первая строка входного файла INPUT.TXT содержит число N ($1 \leq N \leq 100$) – количество заповедников. Далее идет N строк содержащих по пять целых чисел x_1, y_1, x_2, y_2, R . (x_1, y_1) и (x_2, y_2) – координаты противоположных вершин заповедника в метрах ($-10^4 \leq x_1, y_1, x_2, y_2 \leq 10^4$). Стороны заповедников параллельны осям координат. Заповедники, имеющие общую границу,

считаются пересекающимися. R ($0 \leq R \leq 10^4$) – расстояние на которое отодвигается граница заповедника.

Выходные данные

В выходной файл OUTPUT.TXT выведите одно натуральное число – количество оставшихся заповедников после объединения.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	3 3 1 6 4 1 1 -2 2 -3 1 -2 -2 -1 -3 2	2

Задача №98 Игра в числа

(Время: 1 сек. Память: 16 Мб Сложность: 28%)

Игра в числа ведётся на одномерном массиве целых положительных чисел. Перед началом, жеребьёвкой определяется, кто будет ходить первым (первый игрок), а кто – вторым (второй игрок). Процесс игры состоит в том, что игроки по очереди (сначала первый игрок, затем второй, следом опять первый и так далее) вычёркивают числа из массива. Вычеркнуть можно только число, находящееся в конце или начале оставшегося массива. При этом всегда вычёркивается максимальное число из этих двух. Если первое и последнее числа массива равны, то вычёркивается первое. Игра продолжается до того момента, пока не будут вычеркнуты все числа. Каждое вычеркнутое число идёт в актив тому игроку, который его вычеркнул. После окончания игры каждый игрок суммирует вычеркнутые им числа. Победителем объявляется тот, кто наберет больше очков.

Некоторые игроки поняли, что результат не зависит от стратегии игры, и решили попросить Вас написать программу для получения результата.

Входные данные

В первой строке входного файла INPUT.TXT находится одно целое число N – количество чисел в массиве ($1 \leq N \leq 10^4$). Во второй строке находятся N целых положительных чисел из диапазона $[1, 32000]$, разделённых пробелом.

Выходные данные

В выходной файл OUTPUT.TXT выведите два числа, разделенные двоеточием. Первое число – количество очков, набираемых первым игроком при игре на этом массиве, второе число – для второго.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	5 4 4 1 5 4	9 : 9
2	1	1234 : 0

Задача №99

Лабиринт

(Время: 1 сек. Память: 16 Мб Сложность: 57%)

Открыв глаза, Принц Персии обнаружил, что находится на верхнем уровне подземного лабиринта Джаффара. Лабиринт состоит из h уровней, расположенных строго друг под другом. Каждый уровень представляет собой прямоугольную площадку, разбитую на $m \times n$ участков. На некоторых участках стоят колонны, поддерживающие потолок, на такие участки Принц заходить не может.

Принц может перемещаться с одного участка на другой соседний свободный участок того же уровня, так же он может проломить пол под собой и оказаться уровнем ниже (на самом нижнем уровне пол проломить нельзя). Любое перемещение занимает у Принца 5 секунд.

На одном из участков нижнего уровня Принца ждет Принцесса. Помогите Принцу найти Принцессу, потратив на это как можно меньше времени.

Входные данные

В первой строке входного файла INPUT.TXT содержатся натуральные числа h , m и n — высота и горизонтальные размеры лабиринта ($2 \leq h, m, n \leq 50$). Далее во входном файле приведены h блоков, описывающих уровни лабиринта в порядке от верхнего к нижнему. Каждый блок содержит m строк, по n символов в каждой: «.» обозначает свободный участок, «о» обозначает участок с колонной, «1» обозначает свободный участок, в котором оказался Принц в начале своего путешествия, «2» обозначает свободный участок, на котором томится Принцесса. Символы «1» и «2» встречаются во входном файле ровно по одному разу: символ «1» — в описании самого верхнего уровня, а символ «2» — в описании самого нижнего. Соседние блоки разделены одной пустой строкой.

Выходные данные

В выходной файл OUTPUT.TXT выведите минимальное время в секундах, необходимое Принцу, чтобы найти Принцессу. Поскольку добро всегда побеждает Зло, гарантируется, что Принц может это сделать.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	3 3 3 1.. оо. ... ooo ..o .oo ooo o.. o.2	60

Задача №100

Счастливые билеты

(Время: 1 сек. Память: 16 Мб Сложность: 68%)

Требуется вычислить количество N - значных счастливых билетов. Напомним, что билет называется счастливым, если сумма первой половины его цифр равна сумме второй его половины. Например, билет 564159 счастливый, т.к. $5+6+4=1+5+9$.

Входные данные

В единственной строке входного файла INPUT.TXT записано натуральное четное число N ($N \leq 100$) – количество цифр в билете.

Выходные данные

В единственную строку выходного файла OUTPUT.TXT нужно вывести одно целое число – количество N -значных счастливых билетов.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	4	670
2	6	55252
3	12	39581170420

Задача №101

Магараджа

(Время: 1 сек. Память: 16 Мб Сложность: 70%)

Магараджа — это шахматная фигура, сочетающая возможности ферзя и коня. Таким образом, магараджа может ходить и бить на любое количество клеток по диагонали, горизонтали и вертикали (т.е. как ферзь), а также либо на две клетки по горизонтали и на одну по вертикали, либо на одну по горизонтали и на две по вертикали (как конь).

Ваша задача — найти число способов расставить на доске N на N ровно K магараджей так, чтобы они не били друг друга.

Входные данные

Входной файл INPUT.TXT содержит два целых числа: N и K ($1 \leq K \leq N \leq 10$).

Выходные данные

В выходной файл OUTPUT.TXT выведите ответ на задачу.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	3 1	9
2	4 2	20
3	5 3	48

Задача №102

Треугольник

(Время: 1 сек. Память: 16 Мб Сложность: 32%)

В декартовой системе координат на плоскости заданы координаты вершин треугольника и еще одной точки. Требуется написать программу, определяющую, принадлежит ли эта точка треугольнику.

Входные данные

В четырех строках входного файла INPUT.TXT находятся пары целых чисел - координаты точек. Числа в первых трех строках - это координаты вершин треугольника (x_1, y_1) , (x_2, y_2) , (x_3, y_3) , в четвертой строке - координаты тестируемой точки (x_4, y_4) . Все координаты не превышают 10000 по абсолютной величине.

Выходные данные

В выходной файл OUTPUT.TXT необходимо вывести слово «In», если точка находится внутри треугольника и «Out» в противном случае.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	0 0 100 0 0 100 100 100	Out
2	0 0 100 0 0 100 10 10	In
3	0 0 100 0 0 100 50 50	In
4	0 0 100 0 0 100 0 0	In

Задача №103

Снова A+B

(Время: 1 сек. Память: 16 Мб Сложность: 35%)

Требуется сложить два целых числа A и B.

Входные данные

Во входном файле INPUT.TXT записано два неотрицательных целых числа, не превышающих 10^{100} , по одному в каждой строке.

Выходные данные

В единственную строку выходного файла OUTPUT.TXT нужно вывести одно целое число — сумму чисел A и B, без лидирующих нулей.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	2 3	5

Задача №104 Шаблон

(Время: 1 сек. Память: 16 Мб Сложность: 65%)

Будем рассматривать слова из больших латинских букв и шаблоны, состоящие из больших латинских букв и символов «?» и «*». Говорят, что слово подходит под шаблон, если в шаблоне можно заменить каждый символ «?» на большую латинскую букву, а каждый символ «*» - на последовательность (возможно, пустую) больших латинских букв, так, чтобы получилось требуемое слово. Требуется написать программу, определяющую, подходит ли слово под шаблон.

Входные данные

В первых двух строках входного файла INPUT.TXT записаны шаблон и слово: в одной строке записан шаблон - последовательность больших латинских букв, «?» и «*», в другой - слово, состоящее только из больших латинских букв. Обе строки входного файла не превышают 255 символов.

Выходные данные

В выходной файл OUTPUT.TXT необходимо вывести слово «YES», если слово подходит под шаблон и «NO» в противном случае.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	ABBCDA A*CDA	YES

Задача №105 Раскопки

(Время: 1 сек. Память: 16 Мб Сложность: 76%)

Во время недавних раскопок на Марсе были обнаружены листы бумаги с таинственными символами на них. После долгих исследований учёные пришли к выводу, что надписи на них на самом деле могли быть обычными числовыми равенствами. Кроме того, из других источников было получено веское доказательство того, что марсиане знали только три операции - сложение, умножение и вычитание (марсиане никогда не использовали «унарный минус»: вместо «-5» они писали «0-5»). Также ученые доказали, что марсиане не наделяли операции разным приоритетом, а просто вычисляли выражения (если в них не было скобок) слева направо: например, $3+3*5$ у них равнялось 30, а не 18. К сожалению, символы арифметических действий стерлись. Например, если была запись «18=7 (5 3) 2», то возможно восстановить эту запись как «18=7+(5-3)*2». Требуется

написать программу, находящую требуемую расстановку знаков или сообщающую, что таковой не существует.

Входные данные

Первая строка входного файла INPUT.TXT состоит из натурального числа, не превосходящего 2^{30} , знака равенства, и последовательности натуральных чисел (не более десяти), произведение которых также не превосходит 2^{30} . Некоторые группы чисел (одно или более) могут быть окружены скобками. Длина входной строки не будет превосходить 80 символов, и других ограничений на количество и вложенность скобок нет. Между двумя соседними числами, не разделенными скобками, всегда будет хотя бы один пробел, во всех остальных местах может быть любое (в том числе и 0) число пробелов (естественно, внутри числа пробелов нет).

Выходные данные

В выходной файл OUTPUT.TXT необходимо вывести одну строку, содержащую полученное равенство (т.е., исходное равенство со вставленными знаками арифметических действий без лишних пробелов). В случае если требуемая расстановка знаков невозможна, вывести строку, состоящую из единственного числа «-1». Выходная строка не должна содержать пробелов.

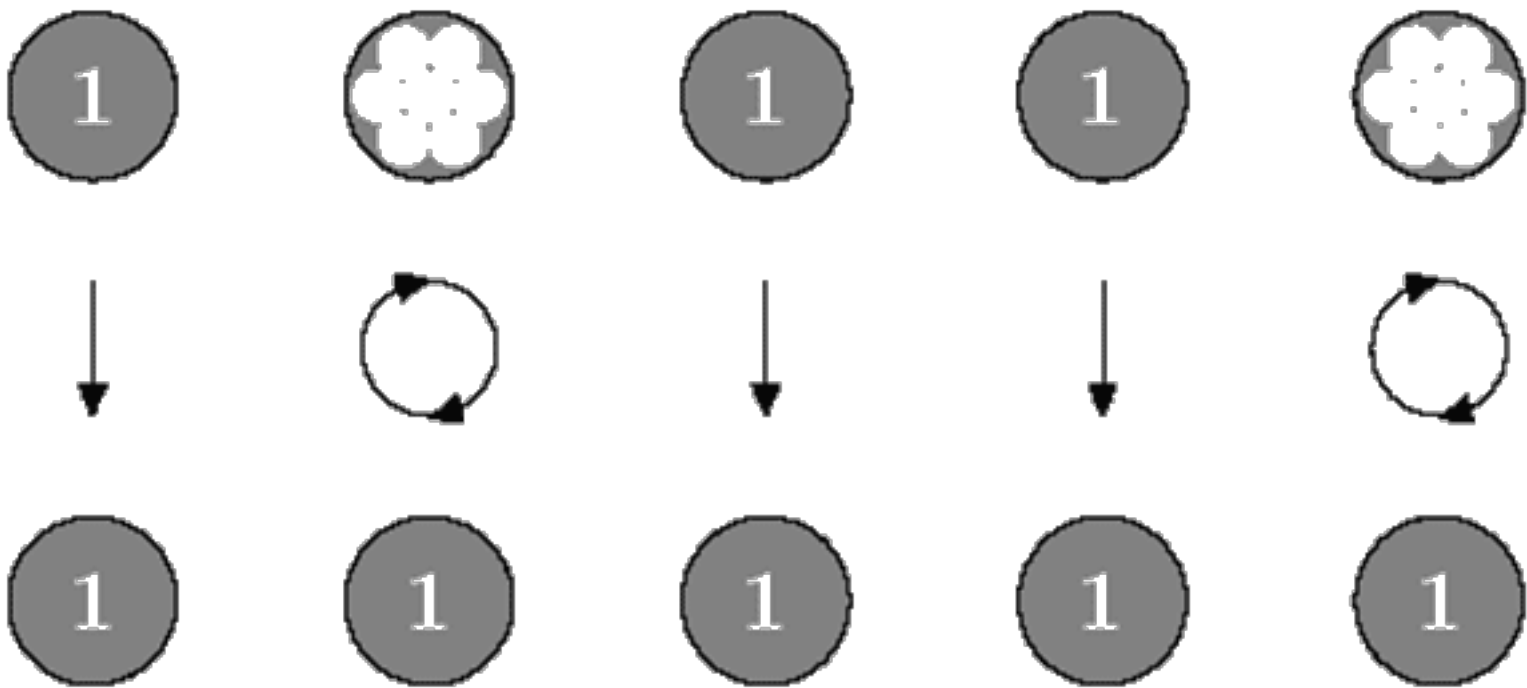
Примеры

№	INPUT.TXT	OUTPUT.TXT
1	18=7 (5 3) 2	18=7+(5-3) *2
2	5= 3 3	-1

Задача №106 Монетки

(Время: 1 сек. Память: 16 Мб Сложность: 8%)

На столе лежат n монеток. Некоторые из них лежат вверх решкой, а некоторые – гербом. Определите минимальное число монеток, которые нужно перевернуть, чтобы все монетки были повернуты вверх одной и той же стороной.



Входные данные

В первой строке входного файла INPUT.TXT записано натуральное число N ($1 \leq N \leq 100$) –

число монеток. В каждой из последующих N строк содержится одно целое число – 1 если монетка лежит решкой вверх и 0 если вверх гербом.

Выходные данные

В выходной файл OUTPUT.TXT выведите минимальное количество монет, которые нужно перевернуть.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	5 1 0 1 1 0	2

Задача №107 Красивые номера

(Время: 1 сек. Память: 16 Мб Сложность: 62%)

Вы, наверное, замечали, что многие компании используют для рекламы «красивые» номера телефонов, которые удобны для запоминания потенциальными клиентами. Но что делать, если номер вашей компании ничем не примечателен? Можно присмотреться к нему повнимательнее, а вдруг, если перегруппировать цифры номера некоторым образом, номер станет намного красивее? Например, если у вашей компании номер 872-73-33, то его можно сделать красивее, если перегруппировать цифры так: 8727-333.

Введем следующую оценку красоты разбиения номера. Будем разбивать номер дефисами на группы размером от 2 до 4 цифр. Теперь красотой разбиения назовем сумму баллов, которые приносит каждая группа. Эти баллы будем считать, пользуясь приведенной справа таблицей.

Шаблон группы	Баллы
aa	2
aba	2
aab, abb	2
aaa	3
abac, baca	2
abab	3
aabb	3
abba	4
baaa, abaa, aaba, aaab	3
aaaa	5

В этой таблице символами «a», «b», «c» обозначены различные цифры. Например, под шаблон «aab» подходят группы «223», «667», но не подходят «123» и «888». Пользуясь предложенной оценкой, найдите наиболее красивое разбиение заданного номера.

Входные данные

Входной файл INPUT.TXT содержит одну строку из 7 цифр – заданный телефонный номер.

Выходные данные

Выведите в первой строке выходного файла OUTPUT.TXT наиболее красивое разбиение номера, а во второй – величину его красоты. Если разбиений с максимальной величиной красоты несколько, выведите в выходной файл любое из этих разбиений.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	8727333	8727-333 5
2	8827291	88-272-91 4

Задача №108

Неглухой телефон

(Время: 1 сек. Память: 16 Мб Сложность: 1%)

Возможно, что Вы когда то играли в игру «Глухой телефон», либо слышали о ней. В этой игре участникам приходится передавать информацию друг другу различными способами: словесно, образно, бывает даже приходится писать левой рукой текст, который другой участник команды должен будет прочитать. Так же известно, что практически никогда передаваемая информация не доходит до конечного адресата. Обозначим за $F_i(x)$ функцию, которая преобразует текст передаваемой информации x в ту, которую получит участник $i+1$ от участника i . Тогда последний n -й участник получит данные y , которые будут выражаться следующей формулой:

$$y = F_{n-1}(F_{n-2}(\dots F_2(F_1(x))))$$

Но Вам необходимо исключить какие-либо внешние факторы, которые могут исказить исходную информацию и Вы должны реализовать программу «неглухой телефон», которая сможет безошибочно доставлять исходные данные, т.е. в нашем случае функция $F_i(x) = x$ для всех i от 1 до $n-1$.

Входные данные

В единственной строке входного файла INPUT.TXT записано натуральное число от 1 до 100.

Выходные данные

В выходной файл OUTPUT.TXT нужно вывести в точности то же число, которое задано во входном файле.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	5	5

Задача №109

A / B

(Время: 1 сек. Память: 16 Мб Сложность: 43%)

Требуется получить точное значение частного A/B для двух натуральных чисел A и B.

Входные данные

В единственной строке входного файла INPUT.TXT записано частное двух натуральных чисел, не превышающих 1000. Числа разделены символом «/» без лишних пробелов.

Выходные данные

В выходной файл OUTPUT.TXT нужно вывести точное значение A/B без лишних точек, нулей и пробелов. В случае присутствия бесконечной записи числа следует вывести период в скобках.

Например, неправильно выведены числа: 08.92, 3.20, 120.6(6), 0.(33), 5.(0), 2. , .3, 0.33(03) . Их следует выводить как 8.92, 3.2, 120.(6), 0.(3), 5, 2, 0.3, 0.3(30) .

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	10 / 7	1 . (4 2 8 5 7 1)
2	1 / 3	0 . (3)
3	100 / 25	4

Задача №110 Красивые числа

(Время: 1 сек. Память: 16 Мб Сложность: 78%)

Саша считает красивыми числа, десятичная запись которых не содержит других цифр, кроме 0 и k ($1 \leq k \leq 9$). Например, если $k = 2$, то такими числами будут 2, 20, 22, 2002 и т.п. Остальные числа Саше не нравятся, поэтому он представляет их в виде суммы красивых чисел. Например, если $k = 3$, то число 69 можно представить так: $69 = 33 + 30 + 3 + 3$.

Однако, не любое натуральное число можно разложить в сумму красивых целых чисел. Например, при $k = 5$ число 6 нельзя представить в таком виде. Но если использовать красивые десятичные дроби, то это можно сделать: $6 = 5.5 + 0.5$.

Недавно Саша изучил периодические десятичные дроби и начал использовать и их в качестве слагаемых. Например, если $k = 3$, то число 43 можно разложить так: $43 = 33.(3) + 3.(3) + 3 + 3.(3)$.

Оказывается, любое натуральное число можно представить в виде суммы положительных красивых чисел. Но такое разложение не единственно — например, число 69 можно также представить и как $69 = 33 + 33 + 3$. Сашу заинтересовало, какое минимальное количество слагаемых требуется для представления числа n в виде суммы красивых чисел.

Требуется написать программу, которая для заданных чисел n и k находит разложение числа n в сумму положительных красивых чисел с минимальным количеством слагаемых.

Входные данные

Во входном файле INPUT.TXT записаны два натуральных числа n и k ($1 \leq n \leq 10^9$; $1 \leq k \leq 9$).

Выходные данные

В выходной файл OUTPUT.TXT выведите разложение числа n в сумму положительных чисел, содержащих только цифры 0 и k, количество слагаемых в котором минимально. Разложение должно быть представлено в виде: $n = a_1 + a_2 + \dots + a_m$. Слагаемые a_1, a_2, \dots, a_m должны быть выведены без ведущих нулей, без лишних нулей в конце дробной части. Запись каждого слагаемого должна быть такой, что длины периода и предпериода дробной части имеют минимально возможную длину. Если решений несколько, то следует вывести то, где меньше слагаемых и по возможности с наименьшим количеством чисел с периодической дробью.

Например, неправильно выведены числа: 07.7; 2.20; 55.5(5); 0.(66); 7.(0); 7. ; .5; 0.33(03). Их следует выводить так: 7.7; 2.2; 55.(5); 0.(6); 7; 7; 0.5; 0.3(30). Предпериод и период каждого из выведенных чисел должны состоять не более чем из 100 цифр. Гарантируется, что хотя бы одно такое решение существует. Числа ряда следует выводит в порядке неубывания. Выходной файл не

должен содержать пробелов.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	69 3	69=3+33+33
2	6 5	6=0.5+5.5
3	10 9	10=9.(9)

Задача №111 Игра «Пуговицы»

(Время: 0,5 сек. Память: 16 Мб Сложность: 48%)

Правила игры очень просты. Перед двумя играющими находится кучка из K пуговиц. Играющие по очереди берут пуговицы из кучки, причем за один ход каждый из них может взять от 1 до L пуговиц. Выигрывает тот из спортсменов, которому удастся взять последнюю пуговицу.

Тот из игроков, которому по жребию выпадает делать первый ход, получает возможность собственноручно назначить число K . Тот из игроков, который будет ходить вторым, выбирает, в свою очередь, число L .

Вам необходимо определить наилучшую стратегию для участника, который ходит вторым.

Входные данные

Во входном файле INPUT.TXT записано одно натуральное число K ($1 \leq K \leq 10^8$) – общее количество пуговиц.

Выходные данные

В выходной файл OUTPUT.TXT необходимо вывести целое число L ($2 \leq L < K$) – максимальное количество пуговиц, которое можно взять за один ход, обеспечивающее победу второму игроку. Если таких чисел несколько, то следует вывести наименьшее из них. Если таких чисел нет, то следует вывести число 0.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	3	2
2	26	12
3	31	30

Задача №112 Армия

(Время: 1 сек. Память: 16 Мб Сложность: 58%)

Всем известно, что в армии без строевой подготовки и порядка дело не обходится и за этим там строго следят. Однажды утром сержант построил всех своих подчиненных в K рядов по N человек в каждом, но оказалось, что солдаты выстроились не по росту, и поэтому сержант решил их наказать. Солдаты должны были выстроиться по росту в каждом отдельном ряде так, что слева должны были стоять самые низкие, а справа самые высокие. Ну а поскольку в армии виноваты всегда слабые (низкие), то наказание было следующим: каждый солдат должен был отжаться столько раз, сколько

солдат стоит от него слева выше его ростом.

Оказалось, что все солдаты были разного роста, и многим пришлось отжиматься достаточно много раз. Сержанту стало интересно: сколько же раз в общей сложности пришлось отжаться солдатам?

Помогите ему решить эту задачу!

Входные данные

В первой строке входного файла INPUT.TXT записаны два натуральных числа N и K ($2 \leq N \leq 10^4$, $1 \leq K \leq 20$) – число солдат в каждом ряду и число рядов. Следующие K строк файла содержат ровно N разных натуральных чисел от 1 до N – рост солдат. Первое число ряда – рост первого солдата (самого левого в ряду), второе – рост второго, и т.д.

Выходные данные

В выходной файл OUTPUT.TXT необходимо вывести общее количество отжиманий, которые должны были выполнить солдаты.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	3 3 1 2 3 2 1 3 3 2 1	4
2	5 2 1 5 2 4 3 2 3 1 5 4	7

Задача №113 Фермер

(Время: 1 сек. Память: 16 Мб Сложность: 46%)

Фермер решил на своем квадратном участке земли вспахать пашню квадратной формы максимальной площади, т.к. он посчитал, что именно квадратная форма пашни наиболее удобна для обработки. Но на его участке присутствуют деревья и хозяйственные постройки, которые он никуда не хочет переносить, а так же иные места, не пригодные для пашни. Для удобства он составил квадратную карту местности $N \times N$ в форме матрицы и пометил нулями непригодные для пашни зоны, в остальные зоны он поставил единицу.

Необходимо помочь фермеру определить максимальную площадь пашни.

Входные данные

В первой строке входного файла INPUT.TXT записано единственное натуральное число N ($1 \leq N \leq 1000$) – длина стороны квадратного участка фермы. Далее, следует N строк, в каждой из которых находится последовательность (без пробелов) нулей и единиц, описывающих ферму.

Выходные данные

В выходной файл OUTPUT.TXT необходимо вывести максимально возможную площадь пашни.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	7 1101101 1111110 1011100 0011100 1000010 1100111 1001110	9

Задача №114 Без двух нулей подряд

(Время: 1 сек. Память: 16 Мб Сложность: 37%)

Требуется вычислить количество N-значных чисел в системе счисления с основанием K, таких что их запись не содержит двух подряд идущих нулей.

Входные данные

Во входном файле INPUT.TXT записаны два натуральных числа N и K в десятичной системе счисления ($2 \leq K \leq 10$; $2 \leq N$; $4 \leq N+K \leq 18$).

Выходные данные

В выходной файл OUTPUT.TXT необходимо вывести целое число в десятичной записи – ответ на задачу.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	2 10	90
2	4 2	5
3	6 3	328

Задача №115 Прямоугольник

(Время: 1 сек. Память: 16 Мб Сложность: 42%)

Задан целочисленный прямоугольный массив MxN. Необходимо определить прямоугольную область данного массива, сумма элементов которого максимальна.

Входные данные

В первой строке входного файла INPUT.TXT записаны два натуральных числа N и M ($1 \leq N, M \leq 100$) – количество строк и столбцов прямоугольной матрицы. Далее идут N строк по M чисел, записанных через пробел – элементы массива, целые числа, не превосходящие 100 по абсолютной величине.

Выходные данные

В выходной файл OUTPUT.TXT необходимо вывести целое число – сумму элементов найденного прямоугольного подмассива.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	2 3 5 0 9 1 2 7	24
2	4 5 -7 8 -1 0 -2 2 -9 2 4 -6 -7 0 6 8 1 4 -8 -1 0 -6	20

Задача №116 Фермер - 2

(Время: 1 сек. Память: 16 Мб Сложность: 60%)

После решения задачи с пашней земли, фермер хочет построить на этой земле как можно больший по площади сарай прямоугольной формы. Но на его участке есть деревья и хозяйственные постройки, которые он не хочет никуда переносить. Для простоты представим ферму прямоугольной сеткой размера $M \times N$. Каждое из деревьев и построек размещается в одном или нескольких узлах сетки. Сарай должен быть построен на свободных узлах сетки.

Помогите фермеру определить максимально возможную площадь сарая.

Входные данные

В первой строке входного файла INPUT.TXT записаны два натуральных числа N и M ($1 \leq N, M \leq 1000$) – размеры фермы. Далее, следует N строк, в каждой из которых находится последовательность (без пробелов) из M нулей и единиц, описывающих ферму. Единицы соответствуют свободным для постройки участкам.

Выходные данные

В выходной файл OUTPUT.TXT необходимо вывести максимально возможную площадь сарая, который может построить фермер на своем участке.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	5 10 1011011111 0111111110 1111111111 1011111111 1101110111	21

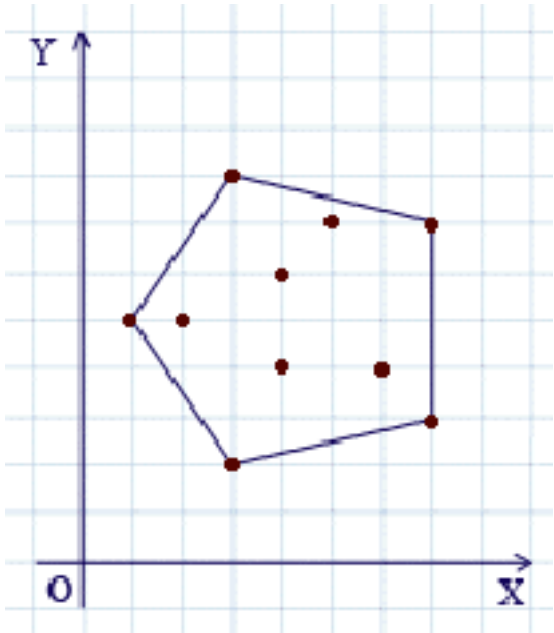
Задача №117 Опасная зона

(Время: 1 сек. Память: 16 Мб Сложность: 57%)

Группа экспертов обнаружила на территории нежилго массива множество опасных участков, соприкосновение с которыми небезопасно для жизни человека.

В целях безопасности требуется создать защитный периметр в форме выпуклого многоугольника, который бы смог обезопасить проникновение человека в эту зону.

По заданным координатам опасных участков требуется вычислить минимально возможную площадь опасной зоны, которая попадет в защитный периметр.



Входные данные

В первой строке входного файла INPUT.TXT записано натуральное число N – количество опасных участков. В каждой из N последующих строк находятся два числа X_i и Y_i - координаты участков, размерами которых можно пренебречь. При этом участки могут повторяться.

Все числа целые, не превосходящие 1000 по абсолютной величине.

Выходные данные

В выходной файл OUTPUT.TXT необходимо вывести одно число — площадь опасной зоны, округленной до целого значения.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	10 4 6 2 5 6 4 7 7 4 4 1 5 3 8 3 2 5 7 7 3	26

Задача №118 Задача Иосифа Флавия

(Время: 1 сек. Память: 16 Мб Сложность: 19%)

Существует легенда, что Иосиф Флавий - известный историк первого века - выжил и стал известным благодаря математической одаренности. В ходе иудейской войны он в составе отряда из 41 иудейского воина был загнан римлянами в пещеру. Предпочитая самоубийство плену, воины решили выстроиться в круг и последовательно убивать каждого третьего из живых до тех пор, пока не останется ни одного человека. Однако Иосиф наряду с одним из своих единомышленников счел подобный конец бессмысленным - он быстро вычислил спасительные места в порочном круге, на

которые поставил себя и своего товарища. И лишь поэтому мы знаем его историю...

В нашем варианте мы начнем с того, что выстроим в круг N человек, пронумерованных числами от 1 до N, и будем исключать каждого k-ого до тех пор, пока не уцелеет только один человек.

Например, если N=10, K=3, то сначала умрет 3-й, потом 6-й, затем 9-й, затем 2-й, затем 7-й, потом 1-й, потом 8-й, за ним - 5-й, и потом 10-й. Таким образом, уцелеет 4-й.

Требуется написать программу, которая по заданным N и K будет определять номер уцелевшего человека.

Входные данные

Входной файл INPUT.TXT содержит два натуральных числа N и K. Ограничения: N<=500, K<=100.

Выходные данные

В выходной файл OUTPUT.TXT нужно вывести номер уцелевшего человека.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	10 3	4

Задача №119 Сортировка времени

(Время: 1 сек. Память: 16 Мб Сложность: 13%)

Требуется выполнить сортировку временных моментов, заданных в часах, минутах и секундах.

Входные данные

Во входном файле INPUT.TXT в первой строке записано число N (1<=N<=100), а в последующих N строках N моментов времени. Каждый момент времени задается 3 целыми числами - часы (от 0 до 23), минуты (от 0 до 59) и секунды (от 0 до 59).

Выходные данные

В выходной файл OUTPUT.TXT выведите моменты времени, упорядоченные в порядке неубывания без ведущих нулей.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	4	7 30 0
	10 20 30	10 20 30
	7 30 00	13 30 30
	23 59 59	23 59 59
	13 30 30	

Задача №120

Минимальный путь в таблице

(Время: 1 сек. Память: 16 Мб Сложность: 32%)

В прямоугольной таблице NxM (в каждой клетке которой записано некоторое число) в начале игрок находится в левой верхней клетке. За один ход ему разрешается перемещаться в соседнюю клетку либо вправо, либо вниз (влево и вверх перемещаться запрещено). При проходе через клетку с игрока берут столько у.е., какое число записано в этой клетке (деньги берут также за первую и последнюю клетки его пути).

1	1	1	1
5	2	2	100
9	4	2	1

Требуется найти минимальную сумму у.е., заплатив которую игрок может попасть в правый нижний угол.

Входные данные

Во входном файле INPUT.TXT задано два числа N и M - размеры таблицы (1<=N<=20, 1<=M<=20). Затем идет N строк по M чисел в каждой - размеры штрафов в у.е. за прохождение через соответствующие клетки (числа от 0 до 100).

Выходные данные

В выходной файл OUTPUT.TXT выведите минимальную сумму, потратив которую можно попасть в правый нижний угол.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	3 4 1 1 1 1 5 2 2 100 9 4 2 1	8
2	5 5 1 1 1 1 1 3 100 100 100 100 1 1 1 1 1 2 2 2 2 1 1 1 1 1 1	11

Задача №121 Гвоздики

(Время: 1 сек. Память: 16 Мб Сложность: 34%)

На прямой доске вбиты гвоздики. Любые два гвоздика можно соединить ниточкой. Требуется соединить некоторые пары гвоздиков ниточками так, чтобы к каждому гвоздику была привязана хотя бы одна ниточка, а суммарная длина всех ниточек была минимальна.

Входные данные

В первой строке входного файла INPUT.TXT записано число N - количество гвоздиков (2 <= N <= 100). В следующей строке записано N чисел - координаты всех гвоздиков (неотрицательные целые числа, не превосходящие 10000).

Выходные данные

В выходной файл OUTPUT.TXT нужно вывести единственное число - минимальную суммарную длину всех ниточек.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	6 3 4 12 6 14 13	5

Задача №122

Максимальная подпоследовательность

(Время: 1 сек. Память: 16 Мб Сложность: 38%)

Дана числовая последовательность, требуется найти длину наибольшей возрастающей подпоследовательности.

Входные данные

В первой строке входного файла INPUT.TXT записано число N - длина последовательности ($1 \leq N \leq 1000$). Во второй строке записана сама последовательность (через пробел). Числа последовательности - целые числа, не превосходящие 10000 по модулю.

Выходные данные

В выходной файл OUTPUT.TXT требуется вывести наибольшую длину возрастающей подпоследовательности.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	6 3 29 5 5 28 6	3

Задача №123

Восстановление скобок

(Время: 1 сек. Память: 16 Мб Сложность: 54%)

Задан шаблон, состоящий из круглых скобок и знаков вопроса. Требуется определить, сколькими способами можно заменить знаки вопроса круглыми скобками так, чтобы получилось правильное скобочное выражение.

Входные данные

Единственная строка входного файла INPUT.TXT содержит заданный шаблон длиной не более 80 символов.

Выходные данные

Выведите в выходной файл OUTPUT.TXT искомое количество способов. Исходные данные будут таковы, что это количество не превзойдет $2 \cdot 10^9$.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	???? (?	2

Задача №124
Светофорчики

(Время: 1 сек. Память: 16 Мб Сложность: 25%)

В подземелье M тоннелей и N перекрестков, каждый тоннель соединяет какие-то два перекрестка. Мышиный король решил поставить по светофору в каждом тоннеле перед каждым перекрестком. Напишите программу, которая посчитает, сколько светофоров должно быть установлено на каждом из перекрестков. Перекрестки пронумерованы числами от 1 до N .

Входные данные

Во входном файле INPUT.TXT записано два числа N и M ($0 < N \leq 100, 0 \leq M \leq N \cdot (N-1)/2$). В следующих M строках записаны по два числа i и j ($1 \leq i, j \leq N$), которые означают, что перекрестки i и j соединены тоннелем. Можно считать, что любые два перекрестка соединены не более, чем одним тоннелем. Нет тоннелей от перекрестка i до него самого.

Выходные данные

В выходной файл OUTPUT.TXT вывести N чисел: k -ое число означает количество светофоров на k -ом перекрестке.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	7 10 5 1 3 2 7 1 5 2 7 4 6 5 6 4 7 5 2 1 5 3	3 3 2 2 5 2 3

Задача №125
Цветной дождь

(Время: 1 сек. Память: 16 Мб Сложность: 26%)

В Банановой республике очень много холмов, соединенных мостами. На химическом заводе произошла авария, в результате чего испарилось экспериментальное удобрение "зован". На следующий день выпал цветной дождь, причем он прошел только над холмами. В некоторых местах падали красные капли, в некоторых - синие, а в остальных - зеленые, в результате чего холмы стали соответствующего цвета. Президенту Банановой республики это понравилось, но ему захотелось покрасить мосты между вершинами холмов так, чтобы мосты были покрашены в цвет холмов, которые они соединяют. К сожалению, если холмы разного цвета, то покрасить мост таким

образом не удастся. Посчитайте количество таких "плохих" мостов.

Входные данные

В файле INPUT.TXT в первой строке записано N ($0 < N \leq 100$) - число холмов. Далее идет матрица смежности, описывающая наличие мостов между холмами (1-мост есть, 0-нет). Предпоследняя строка пустая, а в последней строке записано N чисел, обозначающих цвет холмов: 1 - красный; 2 - синий; 3 - зеленый.

Выходные данные

В файл OUTPUT.TXT вывести количество "плохих" мостов.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	7 0 1 0 0 0 1 1 1 0 1 0 0 0 0 0 1 0 0 1 1 0 0 0 0 0 0 0 0 0 0 1 0 0 1 0 1 0 1 0 1 0 0 1 0 0 0 0 0 0 1 1 1 1 1 3 3	4

Задача №126 Издевательство

(Время: 1 сек. Память: 16 Мб Сложность: 28%)

Штирлиц ехал на машине, увидел голосующего Бормана, и проехал мимо. Через некоторое время он снова увидел голосующего Бормана, и снова проехал мимо. Вскоре он опять увидел голосующего Бормана.

- Издевается! - подумал Борман.
- Кольцевая! - догадался Штирлиц.

В городе N площадей. Любые две площади соединены между собой ровно одной дорогой с двусторонним движением. В этом городе живет Штирлиц. У Штирлица есть хобби - он любит воскресным утром выйти из дома, сесть в машину, выбрать какой-нибудь кольцевой маршрут, проходящий ровно по трем площадям (то есть сначала он едет с какой-то площади на какую-то другую, потом - на третью, затем возвращается на начальную, и опять едет по этому маршруту). Он воображает, что где-то на этом пути стоит Борман. И так вот ездит Штирлиц все воскресенье, пока голова не закружится, и радуется...

Естественно, что Штирлицу хочется проезжать мимо точки, в которой, как он воображает, стоит Борман, как можно чаще. Для этого, естественно, выбранный Штирлицем маршрут должен быть как можно короче. Напишите программу, которая выберет оптимальный для Штирлица маршрут.

Входные данные

Во входном файле INPUT.TXT записано сначала число N ($3 \leq N \leq 100$), а затем матрица NxN расстояний между площадями (число в позиции i,j обозначает длину дороги, соединяющей i-ую и j-

ую площади). Все числа в матрице (кроме стоящих на главной диагонали) - натуральные, не превышающие 1000. Матрица симметрична относительно главной диагонали, на главной диагонали стоят 0.

Выходные данные

В выходной файл OUTPUT.TXT выведите длину оптимального маршрута.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	5 0 20 10 30 40 20 0 30 1 2 10 30 0 40 1000 30 1 40 0 21 40 2 1000 21 0	24

Задача №127 Путь

(Время: 1 сек. Память: 16 Мб Сложность: 40%)

В неориентированном графе требуется найти длину кратчайшего пути между двумя вершинами.

Входные данные

Во входном файле INPUT.TXT записано сначала число N - количество вершин в графе (1<=N<=100). Затем записана матрица смежности (0 обозначает отсутствие ребра, 1 - наличие ребра). Затем записаны номера двух вершин - начальной и конечной.

Выходные данные

В выходной файл OUTPUT.TXT выведите длину кратчайшего пути. Если пути не существует, выведите одно число -1.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	5 0 1 0 0 1 1 0 1 0 0 0 1 0 0 0 0 0 0 0 0 1 0 0 0 0 3 5	3

Задача №128 Один конь

(Время: 1 сек. Память: 16 Мб Сложность: 43%)

На шахматной доске NxN в клетке (x1,y1) стоит голодный шахматный конь. Он хочет попасть

в клетку (x2,y2), где растет вкусная шахматная трава. Какое наименьшее количество ходов он должен для этого сделать?

Входные данные

Входной файл INPUT.TXT содержит пять чисел: N, x1, y1, x2, y2 (5 <= N <= 20, 1 <= x1, y1, x2, y2 <= N). Левая верхняя клетка доски имеет координаты (1,1), правая нижняя - (N,N).

Выходные данные

В выходной файл OUTPUT.TXT необходимо вывести наименьшее число ходов коня.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	5 1 1 3 1	2

Задача №129 Табличка

(Время: 1 сек. Память: 16 Мб Сложность: 40%)

Вам дана табличка, состоящая из N строк и M столбцов. В каждой клетке таблицы стоит либо 0, либо 1. Расстоянием между клетками (x1,y1) и (x2,y2) называется |x1-x2|+|y1-y2|. Вам нужно построить другую таблицу, в которой в каждой клетке стоит расстояние от данной до ближайшей клетки, содержащей 1 (в начальной таблице). Гарантируется, что хотя бы одна 1 в таблице есть.

Входные данные

В первой строке входного файла INPUT.TXT содержатся два натуральных числа, не превосходящих 100 - N и M. Далее идут N строк по M чисел - элементы таблицы.

Выходные данные

Выходной файл OUTPUT.TXT должен содержать N строк по M чисел - элементы искомой таблицы.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	2 3 0 0 1 1 0 0	1 1 0 0 1 1

Задача №130 Два коня

(Время: 1 сек. Память: 16 Мб Сложность: 55%)

На стандартной шахматной доске (8x8) живут 2 шахматных коня: красный и зеленый. Обычно они беззаботно скачут по просторам доски, пощипывая шахматную травку, но сегодня особенный день: у зеленого коня день рождения. зеленый конь решил отпраздновать это событие вместе с

красным. Но для осуществления этого прекрасного плана им нужно оказаться на одной клетке. Заметим, что красный и зеленый шахматные кони сильно отличаются от черного с белым: они ходят не по очереди, а одновременно, и если оказываются на одной клетке, никто никого не съедает. Сколько ходов им потребуется, чтобы насладиться праздником?

Входные данные

Во входном файле INPUT.TXT содержатся координаты коней, записанные по стандартным шахматным правилам (т.е. двумя символами - маленькая латинская буква (от а до h) и цифра (от 1 до 8), задающие столбец и строку соответственно).

Выходные данные

Выходной файл OUTPUT.TXT должен содержать наименьшее необходимое количество ходов, либо -1, если кони не могут встретиться.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	a1 a3	1

Задача №131 Перепись

(Время: 1 сек. Память: 16 Мб Сложность: 15%)

В доме живет N жильцов. Однажды решили провести перепись всех жильцов данного дома и составили список, в котором указали возраст и пол каждого жильца. Требуется найти номер самого старшего жителя мужского пола.

Входные данные

Во входном файле INPUT.TXT в первой строке задано натуральное число N – количество жильцов ($N \leq 100$). В последующих N строках располагается информация о всех жильцах: каждая строка содержит два целых числа: V и S – возраст и пол человека ($1 \leq V \leq 100$, S – 0 или 1). Мужскому полу соответствует значение $S=1$, а женскому – $S=0$.

Выходные данные

Выходной файл OUTPUT.TXT должен содержать номер самого старшего мужчины в списке. Если таких жильцов несколько, то следует вывести наименьший номер. Если жильцов мужского пола нет, то выведите -1.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	4 25 1 70 1 100 0 3 1	2

Задача №132

Алгоритм Дейкстры

(Время: 1 сек. Память: 16 Мб Сложность: 47%)

Дан ориентированный взвешенный граф. Для него вам необходимо найти кратчайшее расстояние от вершины S до вершины F.

Входные данные

В первой строке входного файла INPUT.TXT записаны три числа: N, S и F ($1 \leq N \leq 100$; $1 \leq S, F \leq N$), где N - количество вершин графа. В следующих N строках записаны по N чисел - матрица смежности графа, где число в i-ой строке j-ом столбце соответствует ребру из i в j: -1 означает отсутствие ребра между вершинами, а любое неотрицательное целое число (от 0 до 100) - наличие ребра данного веса . На главной диагонали матрицы всегда записаны нули.

Выходные данные

В выходной файл OUTPUT.TXT необходимо вывести искомое расстояние или -1, если пути между указанными вершинами не существует.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	3 1 2 0 -1 2 3 0 -1 -1 4 0	6

Задача №133 Заправки

(Время: 1 сек. Память: 16 Мб Сложность: 49%)

В стране N городов, некоторые из которых соединены между собой дорогами. Для того, чтобы проехать по одной дороге требуется один бак бензина. В каждом городе бак бензина имеет разную стоимость. Вам требуется добраться из первого города в N-ый, потратив как можно меньшее количество денег.

Входные данные

Во входном файле INPUT.TXT записано сначала число N ($1 \leq N \leq 100$), затем идет N чисел, i-ое из которых задает стоимость бензина в i-ом городе (все числа целые из диапазона от 0 до 100). Далее идет число M - количество дорог в стране, далее идет описание самих дорог. Каждая дорога задается двумя числами - номерами городов, которые она соединяет. Все дороги двухсторонние (то есть по ним можно ездить как в одну, так и в другую сторону); между двумя городами всегда существует не более одной дороги; не существует дорог, ведущих из города в себя.

Выходные данные

В выходной файл OUTPUT.TXT выведите одно число - суммарную стоимость маршрута или -1, если добраться невозможно.

Пример

№	INPUT.TXT	OUTPUT.TXT

1	4	3
	1 10 2 15	
	4	
	1 2	
	1 3	
	4 2	
	4 3	

Пояснение к примеру

Оптимальное решение в примере: из 1-го города поехать в 3-й, а затем в 4-й. Горючее придется покупать в 1 и 3 городах.

Задача №134 Автобусы

(Время: 1 сек. Память: 16 Мб Сложность: 50%)

Между некоторыми деревнями края Власюки ходят автобусы. Поскольку пассажиропотоки здесь не очень большие, то автобусы ходят всего несколько раз в день.

Марии Ивановне требуется добраться из деревни d в деревню v как можно быстрее (считается, что в момент времени 0 она находится в деревне d).

Входные данные

Во входном файле INPUT.TXT записано число N - общее число деревень (1 <= N <= 100), номера деревень d и v, затем количество автобусных рейсов R (0 <= R <= 10000). Затем идут описания автобусных рейсов. Каждый рейс задается номером деревни отправления, временем отправления, деревней назначения и временем прибытия (все времена - целые от 0 до 10000). Если в момент t пассажир приезжает в деревню, то уехать из нее он может в любой момент времени, начиная с t.

Выходные данные

В выходной файл OUTPUT.TXT вывести минимальное время, когда Мария Ивановна может оказаться в деревне v. Если она не сможет с помощью указанных автобусных рейсов добраться из d в v, вывести -1.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	3 1 3 4 1 0 2 5 1 1 2 3 2 3 3 5 1 1 3 10	5

Задача №135 Алгоритм Флойда

(Время: 1 сек. Память: 16 Мб Сложность: 36%)

Полный ориентированный взвешенный граф задан матрицей смежности. Постройте матрицу кратчайших путей между его вершинами. Гарантируется, что в графе нет циклов отрицательного веса.

Входные данные

В первой строке входного файла INPUT.TXT записано единственное число N ($1 \leq N \leq 100$) - количество вершин графа. В следующих N строках по N чисел - матрица смежности графа (j -ое число в i -ой строке соответствует весу ребра из вершины i в вершину j). Все числа по модулю не превышают 100. На главной диагонали матрицы - всегда нули.

Выходные данные

В выходной файл OUTPUT.TXT выведите N строк по N чисел - матрицу кратчайших расстояний между парами вершин. j -ое число в i -ой строке должно быть равно весу кратчайшего пути из вершины i в вершину j .

Пример

№	INPUT.TXT	OUTPUT.TXT
1	4 0 5 9 100 100 0 2 8 100 100 0 7 4 100 100 0	0 5 7 13 12 0 2 8 11 16 0 7 4 9 11 0

Задача №136 Алгоритм Флойда - 2

(Время: 1 сек. Память: 16 Мб Сложность: 39%)

Дан ориентированный взвешенный граф. Вам необходимо найти пару вершин, кратчайшее расстояние от одной из которых до другой максимально среди всех пар вершин.

Входные данные

В первой строке входного файла INPUT.TXT записано единственное число N ($1 \leq N \leq 100$) - количество вершин графа. В следующих N строках по N чисел - матрица смежности графа, где -1 означает отсутствие ребра между вершинами, а любое неотрицательное число - присутствие ребра данного веса. Элементы матрицы - целые числа от -1 до 100. На главной диагонали матрицы - всегда нули.

Выходные данные

В выходной файл OUTPUT.TXT требуется вывести искомое максимальное кратчайшее расстояние.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	4 0 5 9 -1 -1 0 2 8 -1 -1 0 7	16

4	-1	-1	0
---	----	----	---

Задача №137

Существование пути

(Время: 1 сек. Память: 16 Мб Сложность: 65%)

Дан ориентированный взвешенный граф. По его матрице смежности нужно для каждой пары вершин определить: существует кратчайший путь между ними или нет.

Кратчайший путь может не существовать по двум причинам: либо нет ни одного пути, либо есть путь сколь угодно маленького веса.

Входные данные

В первой строке входного файла INPUT.TXT записано единственное число N ($1 \leq N \leq 100$) - количество вершин графа. В следующих N строках по N чисел - матрица смежности графа (j-ое число в i-ой строке соответствует весу ребра из вершины i в вершину j), в которой число 0 обозначает отсутствие ребра, а любое другое число - наличие ребра соответствующего веса. Все числа по модулю не превышают 100.

Выходные данные

В выходной файл OUTPUT.TXT выведите N строк по N чисел: j-ое число в i-ой строке должно быть равно 0, если путь из i в j не существует, 1 - если существует кратчайший путь, и 2 - если существует путь сколь угодно маленького веса.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	5	1 1 1 0 0
	0 1 2 0 0	1 1 1 0 0
	1 0 3 0 0	1 1 1 0 0
	2 3 0 0 0	0 0 0 2 2
	0 0 0 0 -1	0 0 0 2 2
	0 0 0 -1 0	

Задача №138

Алгоритм Форда-Беллмана

(Время: 1 сек. Память: 16 Мб Сложность: 38%)

Дан ориентированный граф, в котором могут быть кратные ребра и петли. Каждое ребро имеет вес, выражающийся целым числом (возможно, отрицательным). Гарантируется, что циклы отрицательного веса отсутствуют.

Требуется посчитать длины кратчайших путей от вершины номер 1 до всех остальных вершин.

Входные данные

Во входном файле INPUT.TXT записано сначала число N ($1 \leq N \leq 100$) - количество вершин графа, далее идет число M ($0 \leq M \leq 10000$) - количество ребер. Далее идет M троек чисел, описывающих ребра: начало ребра, конец ребра и вес (вес - целое число от -100 до 100).

Выходные данные

В выходной файл OUTPUT.TXT выведите N чисел - расстояния от вершины номер 1 до всех вершин графа. Если пути до соответствующей вершины не существует, вместо длины пути выведите число 30000.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	4 5 1 2 10 2 3 10 1 3 100 3 1 -10 2 3 1	0 10 11 30000

Задача №139 Лабиринт знаний

(Время: 1 сек. Память: 16 Мб Сложность: 53%)

В стране Умландии построили аттракцион "Лабиринт знаний". Лабиринт представляет собой N комнат, занумерованных от 1 до N, между некоторыми из которых есть двери. Когда человек проходит через дверь, показатель его знаний изменяется на определенную величину, фиксированную для данной двери. Вход в лабиринт находится в комнате 1, выход - в комнате N. Каждый ученик проходит лабиринт ровно один раз и попадает в ту или иную учебную группу в зависимости от количества набранных знаний (при входе в лабиринт этот показатель равен нулю). Ваша задача показать наилучший результат.

Входные данные

Первая строка входного файла INPUT.TXT содержит целые числа N ($1 \leq N \leq 2000$) - количество комнат и M ($1 \leq M \leq 10000$) - количество дверей. В каждой из следующих M строк содержится описание двери - номера комнат, из которой она ведет и в которую она ведет (через дверь можно ходить только в одном направлении), а также целое число, которое прибавляется к количеству знаний при прохождении через дверь (это число по модулю не превышает 10000). Двери могут вести из комнаты в нее саму, между двумя комнатами может быть более одной двери.

Выходные данные

В выходной файл OUTPUT.TXT выведите ":" - если можно получить неограниченно большой запас знаний, ":(" - если лабиринт пройти нельзя, и максимальное количество набранных знаний в противном случае.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	2 2 1 2 5 1 2 -5	5

Задача №140 Цикл отрицательного веса

(Время: 1 сек. Память: 16 Мб Сложность: 46%)

Дан взвешенный граф. Определить, есть ли в нем цикл отрицательного веса.

Входные данные

Во входном файле INPUT.TXT в первой строке записано число N ($1 \leq N \leq 100$) - количество вершин графа. В следующих N строках находится по N чисел - матрица смежности графа. Веса ребер не превышают по модулю 10000. Если ребра нет, соответствующее значение равно 100000.

Выходные данные

В выходной файл OUTPUT.TXT выведите "YES", если цикл существует, или "NO" в противном случае.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	2 0 -1 -1 0	YES

Задача №141 Дерево

(Время: 1 сек. Память: 16 Мб Сложность: 42%)

Неориентированный граф без петель и кратных ребер задан матрицей смежности. Требуется определить, является ли этот граф деревом.

Входные данные

Во входном файле INPUT.TXT записано сначала число N - количество вершин графа (от 1 до 100). Далее записана матрица смежности размером $N \times N$, в которой 1 обозначает наличие ребра, 0 - его отсутствие. Матрица симметрична относительно главной диагонали.

Выходные данные

В выходной файл OUTPUT.TXT выведите сообщение YES, если граф является деревом, и NO в противном случае.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	3 0 1 0 1 0 1 0 1 0	YES

Задача №142 Минимальный каркас

(Время: 1 сек. Память: 16 Мб Сложность: 53%)

От вас требуется определить вес минимального остовного дерева для неориентированного взвешенного связного графа.

Входные данные

В первой строке входного файла INPUT.TXT находятся числа N и M ($1 \leq N \leq 100$; $1 \leq M \leq 6000$), где N - количество вершин в графе, а M - количество рёбер. В каждой из последующих M строк записано по тройке чисел A, B, C, где A и B - номера вершин, соединённых ребром, а C - вес ребра (натуральное число, не превышающее 30000).

Выходные данные

В выходной файл OUTPUT.TXT выведите одно число - искомый вес.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	3 3 1 2 1 2 3 2 3 1 3	3

Задача №143 A-B

(Время: 1 сек. Память: 16 Мб Сложность: 44%)

Требуется найти разность между неотрицательными числами A и B.

Входные данные

Во входном файле INPUT.TXT в двух строках записаны два неотрицательных целых числа A и B, не превышающие 10^{1000} .

Выходные данные

В выходной файл OUTPUT.TXT выведите значение A-B.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	7 5	2
2	5 17	-12

Задача №144 A*B

(Время: 1 сек. Память: 16 Мб Сложность: 37%)

Даны два целых неотрицательных числа A и B. Требуется найти их произведение.

Входные данные

Во входном файле INPUT.TXT записаны целые неотрицательные числа A и B по одному в

строке ($A < 10^{100}$, $B \leq 10000$).

Выходные данные

В выходной файл OUTPUT.TXT выведите единственное число без лидирующих нулей: $A \cdot B$.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	5 7	35

Задача №145 $A \div B$

(Время: 1 сек. Память: 16 Мб Сложность: 40%)

Даны два целых числа A и B . Требуется найти их целую часть от их частного.

Входные данные

Во входном файле INPUT.TXT записаны целые числа A и B по одному в строке ($0 \leq A \leq 10^{100}$, $0 < B \leq 10000$).

Выходные данные

В выходной файл OUTPUT.TXT выведите единственное число без лидирующих нулей: $A \div B$.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	7 3	2

Задача №146 Длинный корень

(Время: 1 сек. Память: 16 Мб Сложность: 67%)

По заданному натуральному числу A требуется найти наибольшее число B такое, что $B^2 \leq A$.

Входные данные

Во входном файле INPUT.TXT записано натуральное число A ($A \leq 10^{3000}$).

Выходные данные

В выходной файл OUTPUT.TXT выведите максимальное натуральное число B , квадрат которого не превосходит A . Число B следует выводить без лидирующих нулей.

Пример

№	INPUT.TXT	OUTPUT.TXT

1	17	4
---	----	---

Задача №147
Числа Фибоначчи

(Время: 1 сек. Память: 16 Мб Сложность: 16%)

Последовательностью Фибоначчи называется последовательность чисел $a_0, a_1, \dots, a_n, \dots$, где $a_0 = 0, a_1 = 1, a_k = a_{k-1} + a_{k-2} (k > 1)$.

Требуется найти N-е число Фибоначчи.

Входные данные

Во входном файле INPUT.TXT записано целое число N ($0 \leq N \leq 30$).

Выходные данные

В выходной файл OUTPUT.TXT выведите N-е число Фибоначчи.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	7	13

Задача №148
НОД

(Время: 1 сек. Память: 16 Мб Сложность: 15%)

Даны два натуральных числа A и B. Требуется найти их наибольший общий делитель (НОД).

Входные данные

Во входном файле INPUT.TXT в единственной строке записаны натуральные числа A и B через пробел ($A, B \leq 10^9$).

Выходные данные

В выходной файл OUTPUT.TXT выведите НОД чисел A и B.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	12 42	6

Задача №149
Разворот

(Время: 1 сек. Память: 16 Мб Сложность: 9%)

Дано натуральное число N и последовательность из N элементов. Требуется вывести эту последовательность в обратном порядке.

Входные данные

В первой строке входного файла INPUT.TXT записано натуральное число N ($N \leq 10^3$). Во второй строке через пробел идут N целых чисел, по модулю не превосходящих 10^3 - элементы последовательности.

Выходные данные

В выходной файл OUTPUT.TXT выведите заданную последовательность в обратном порядке.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	3 1 2 3	3 2 1

Задача №150 Друзья

(Время: 1 сек. Память: 16 Мб Сложность: 41%)

В клубе N человек. Многие из них - друзья. Так же известно, что друзья друзей так же являются друзьями. Требуется выяснить, сколько всего друзей у конкретного человека в клубе.

Входные данные

В первой строке входного файла INPUT.TXT заданы два числа: N и S ($1 \leq N \leq 100$; $1 \leq S \leq N$), где N - количество человек в клубе, а S – номер конкретного человека. В следующих N строках записано по N чисел - матрица смежности, состоящая из единиц и нулей. Причем единица, стоящая в i -й строке и j -м столбце гарантирует, что люди с номерами i и j – друзья, а 0 – выражает неопределенность.

Выходные данные

В выходной файл OUTPUT.TXT выведите количество гарантированных друзей у человека с номером S , помня о транзитивности дружбы.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	3 1 0 1 0 1 0 1 0 1 0	2

Задача №151 Банкет

(Время: 1 сек. Память: 16 Мб Сложность: 54%)

На банкет были приглашены N Очень Важных Персон (ОВП). Были поставлены 2 стола. Столы достаточно большие, чтобы все посетители банкета могли сесть за любой из них. Проблема заключается в том, что некоторые ОВП не ладят друг с другом и не могут сидеть за одним столом. Вас попросили определить, возможно ли всех ОВП рассадить за двумя столами.

Входные данные

В первой строке входного файла INPUT.TXT дано два целых числа: N и M ($0 \leq N, M \leq 100$), где N - количество ОВП, а M - количество пар ОВП, которые не могут сидеть за одним столом. В следующих M строках записано по 2 числа - пары ОВП, которые не могут сидеть за одним столом.

Выходные данные

Если способ рассадить ОВП существует, то в выходной файл OUTPUT.TXT выведите YES и NO в противном случае.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	3 2 1 2 1 3	YES

Задача №152 Построение

(Время: 1 сек. Память: 16 Мб Сложность: 62%)

Группа солдат-новобранцев прибыла в армейскую часть N666. После знакомства с прапорщиком стало очевидно, что от работ на кухне по очистке картофеля спасти солдат может только чудо.

Прапорщик, будучи не в состоянии запомнить фамилии, пронумеровал новобранцев от 1 до N. После этого он велел им построиться по росту (начиная с самого высокого). С этой несложной задачей могут справиться даже совсем необученные новобранцы, да вот беда, прапорщик уверил себя, что знает про некоторых солдат, кто из них кого выше, и это далеко не всегда соответствует истине.

После трех дней обучения новобранцам удалось выяснить, что знает (а точнее, думает, что знает) прапорщик. Помогите им, используя эти знания, построиться так, чтобы товарищ прапорщик остался доволен.

Входные данные

Во входном файле INPUT.TXT сначала идут числа N и M ($1 \leq N \leq 100$, $1 \leq M \leq 5000$) - количество солдат в роте и количество пар солдат, про которых прапорщик знает, кто из них выше. Далее идут эти пары чисел A и B по одной на строке ($1 \leq A, B \leq N$), что означает, что, по мнению прапорщика, солдат A выше, чем B.

Выходные данные

В выходной файл OUTPUT.TXT выведите "Yes" если можно построиться так, чтобы прапорщик остался доволен и "No" если нельзя.

Пример

№	INPUT.TXT	OUTPUT.TXT
	5 4 1 3	Yes

1	1 4
	4 3
	5 2

Задача №153

Монетки - 2

(Время: 1 сек. Память: 16 Мб Сложность: 51%)

В волшебной стране используются монетки достоинством A_1, A_2, \dots, A_M . волшебный человечек пришел в магазин и обнаружил, что у него есть ровно по две монетки каждого достоинства. Ему нужно заплатить сумму N . Напишите программу, определяющую, сможет ли он расплатиться без сдачи.

Входные данные

Во входном файле INPUT.TXT записано сначала число N ($1 \leq N \leq 10^9$), затем - число M ($1 \leq M \leq 15$) и далее M попарно различных чисел A_1, A_2, \dots, A_M ($1 \leq A_i \leq 10^9$).

Выходные данные

В выходной файл OUTPUT.TXT выведите количество монет, которое придется отдать волшебному человечку, если он сможет заплатить указанную сумму без сдачи. Если решений несколько, выведите вариант, в котором волшебный человек отдаст наименьшее возможное количество монет. Если без сдачи не обойтись, то выведите одно число 0. Если же у волшебного человечка не хватит денег, чтобы заплатить указанную сумму, выведите одно число -1 (минус один).

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	5 2 1 2	3
2	7 2 1 2	-1
3	5 2 3 4	0

Задача №154

Сумма кубов

(Время: 1 сек. Память: 16 Мб Сложность: 52%)

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.

Входные данные

Во входном файле INPUT.TXT записано натуральное число N ($1 \leq N \leq 2 \cdot 10^9$).

Выходные данные

В выходной файл OUTPUT.TXT выведите не более восьми натуральных чисел в порядке невозрастания, кубы которых в сумме дают N . Если вариантов несколько, то выведите любой. Если искомого представления не существует, то в выходной файл необходимо вывести слово IMPOSSIBLE.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	17	2 2 1
2	239	IMPOSSIBLE

Задача №155 Резисторы

(Время: 1 сек. Память: 16 Мб Сложность: 59%)

Радиолобитель Петя решил собрать детекторный приемник. Для этого ему понадобился конденсатор емкостью C мкФ. В распоряжении Пети есть набор из n конденсаторов, емкости которых равны C_1, C_2, \dots, C_n соответственно. Петя помнит, как вычисляется емкость параллельного соединений двух конденсаторов ($C_{new} = C_1 + C_2$) и последовательного соединения двух конденсаторов ($C_{new} = (C_1 \cdot C_2) / (C_1 + C_2)$). Петя хочет спаять некоторую последовательно-параллельную схему из имеющегося набора конденсаторов, такую, что ее емкость ближе всего к искомой (то есть абсолютная величина разности значений минимальна). Разумеется, Петя не обязан использовать для изготовления схемы все конденсаторы.

Напомним определение последовательно-параллельной схемы. Схема, составленная из одного конденсатора, - последовательно-параллельная схема. Любая схема, полученная последовательным соединением двух последовательно-параллельных схем, - последовательно-параллельная, а также любая схема, полученная параллельным соединением двух последовательно-параллельных схем, - последовательно-параллельная.

Входные данные

В первой строке каждого входного файла INPUT.TXT заданы числа n и C . Во второй строке содержится последовательность емкостей имеющихся в наличии конденсаторов C_1, C_2, \dots, C_n . Значения всех емкостей - вещественные числа. Для всех входных файлов $n < 7$.

Выходные данные

В выходной файл OUTPUT.TXT необходимо вывести YES, если Пете удастся собрать схему, емкость которой отличается не более чем на 0.01 от требуемого значения C . В противном случае следует вывести NO.

Пример

--	--	--

№	INPUT.TXT	OUTPUT.TXT
1	3 1.66 1 2 1	YES

Пояснения к примеру

Последовательно соединим первый и второй конденсаторы, а затем полученную схему соединим параллельно с третьим. Полученная схема будет иметь емкость 1.(6)

Задача №156 Шахматы - 2

(Время: 1 сек. Память: 16 Мб Сложность: 46%)

Требуется найти число способов расставить на шахматной доске N×N K ладей так, чтобы они не били друг друга. Все ладьи считаются одинаковыми.

Входные данные

Во входном файле INPUT.TXT записаны натуральные числа N и K (N, K ≤ 8).

Выходные данные

В выходной файл OUTPUT.TXT выведите одно целое число – ответ на задачу.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	8 8	40320

Задача №157 Карточки

(Время: 1 сек. Память: 16 Мб Сложность: 48%)

На день рождения Пете подарили набор карточек с буквами. Теперь Петя с большим интересом составляет из них разные слова. И вот, однажды, составив очередное слово, Петя заинтересовался вопросом: "А сколько различных слов можно составить из тех же карточек, что и данное?".

Помогите ему ответить на этот вопрос.

Входные данные

Во входном файле INPUT.TXT задано слово, составленное Петей - строка из маленьких латинских букв не длиннее 15 символов.

Выходные данные

В выходной файл OUTPUT.TXT выведите одно целое число - ответ на поставленную задачу.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	solo	12

Задача №158

Великий комбинатор

(Время: 1 сек. Память: 16 Мб Сложность: 60%)

В результате очередной хитроумной комбинации у Остапа Бендера и его компаньонов - К детей лейтенанта Шмидта оказалось X рублей пятирублевыми банкнотами. И вот дело, как водится, дошло до дележа...

Шура Балаганов предложил "по справедливости", т.е. всем поровну. Паниковский порешил себе отдать половину, а остальным "по заслугам". Каждый из K детей лейтенанта предложил что-нибудь интересное. Однако, у Великого Комбинатора имелось свое мнение на этот счет...

Ваша же задача состоит в нахождении количества способов разделить имеющиеся деньги между всеми участниками этих славных событий: K детьми лейтенанта Шмидта и Остапом Бендером.

Входные данные

Во входном файле INPUT.TXT записаны целые числа X ($0 \leq X \leq 500$) и K ($0 \leq K \leq 100$). Естественно, что число X делится на 5. Да и при дележе рвать пятирублевые банкноты не разрешается.

Выходные данные

В выходной файл OUTPUT.TXT выведите одно целое число - количество способов дележа.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	15 2	10

Задача №159

Обратная перестановка

(Время: 1 сек. Память: 16 Мб Сложность: 25%)

По заданной перестановке требуется определить обратную.

Перестановкой из N элементов называется упорядоченный набор из N различных чисел от 1 до N . Количество различных перестановок порядка N равно $P_N = N!$

Пусть у нас есть упорядоченное множество из N элементов. Перестановка задает преобразование этого множества. А именно, она говорит, что на i место нужно поставить a_i элемент множества, где a_i - i -тый элемент перестановки.

Обратной перестановкой к перестановке π называется такая перестановка π^{-1} , что $\pi\pi^{-1} = \pi^{-1}\pi = \varepsilon$, где ε - тождественная перестановка.

Входные данные

В первой строке входного файла INPUT.TXT записано число $0 < N \leq 20000$ - порядок перестановки. Во второй строке записана сама перестановка.

Выходные данные

В выходной файл OUTPUT.TXT выведите обратную перестановку.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	3 2 3 1	3 1 2

Задача №160

Степень перестановки

(Время: 1 сек. Память: 16 Мб Сложность: 46%)

Требуется вычислить степень заданной перестановки.

Перестановкой из N элементов называется упорядоченный набор из N различных чисел от 1 до N . Количество различных перестановок порядка N равно $P_N = N!$

Пусть у нас есть упорядоченное множество из N элементов. Перестановка задает преобразование этого множества. А именно, она говорит, что на i место нужно поставить a_i элемент множества, где a_i - i -тый элемент перестановки.

Обратной перестановкой к перестановке π называется такая перестановка π^{-1} , что $\pi\pi^{-1} = \pi^{-1}\pi = \varepsilon$, где ε – тождественная перестановка.

Степенью перестановки называется минимальное натуральное число k такое, что $\pi^k = \varepsilon$

Входные данные

В первой строке входного файла INPUT.TXT записано число $0 < N \leq 100$ - порядок перестановки. Во второй строке записана сама перестановка.

Выходные данные

В выходной файл OUTPUT.TXT выведите степень данной перестановки. Гарантируется, что ответ не превышает 10^9 .

Пример

№	INPUT.TXT	OUTPUT.TXT
1	3 2 3 1	3

Задача №161

Восстановление перестановки

(Время: 1 сек. Память: 16 Мб Сложность: 44%)

Перестановкой из N элементов называется упорядоченный набор из N различных чисел от 1 до N .

Пусть у нас есть упорядоченное множество из N элементов. Перестановка задает преобразование этого множества. А именно, она говорит, что на i место нужно поставить a_i элемент множества, где a_i - i -тый элемент перестановки.

Пусть дана перестановка π . Обозначим $\phi[i]$ - количество таких j , что $\pi[j] > \pi[i]$, а $j < i$. ϕ

называется *таблицей инверсий* перестановки π .

Требуется по данной таблице инверсий восстановить перестановку.

Входные данные

В первой строке входного файла INPUT.TXT записано число $0 < N \leq 2000$ - порядок перестановки. Во второй строке записана таблица инверсий.

Выходные данные

В выходной файл OUTPUT.TXT выведите искомую перестановку.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	3 0 0 2	2 3 1

Задача №162 Манхэттенский полицейский (Время: 1 сек. Память: 16 Мб Сложность: 56%)

Недавно Билл устроился на работу полицейским. Теперь ему предстоит каждый вечер обходить свой участок, который представляет собой прямоугольник, состоящий из $N \times M$ кварталов. Каждый квартал имеет вид квадрата размером 100×100 метров, кварталы отделены друг от друга прямыми улицами.

Таким образом, через участок Билла проходит $N+1$ улица, идущая с запада на восток, и $M+1$ улица, идущая с севера на юг. Перекрестки разбивают улицы на $(N+1) \cdot M + (M+1) \cdot N$ отрезков, каждый из которых имеет длину 100 метров.

Совершая обход, Билл выходит из полицейского управления, расположенного около юго-западного угла его участка, обходит свой участок и возвращается в управление. Во время обхода Билл должен пройти по каждому отрезку улицы на территории своего участка как минимум один раз. Известно, что во время обхода Билл проходит отрезок длиной 100 метров за одну минуту. Выясните, какое минимальное число минут потребуется Биллу, чтобы совершить обход.

Входные данные

Входной файл INPUT.TXT содержит натуральные числа N и M , не превышающие 10 000.

Выходные данные

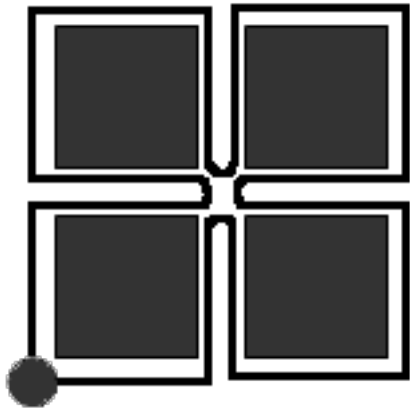
В выходной файл OUTPUT.TXT выведите минимальное время, за которое Билл может совершить обход.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	1 1	4
2	2 2	16
3	4 3	38

Пояснение

Один из возможных оптимальных путей для Билла во втором примере показан на рисунке:



Задача №163 Уравнение для 5 класса!

(Время: 1 сек. Память: 16 Мб Сложность: 25%)

Уравнение для пятиклассников представляет собой строку длиной 5 символов. Вторым символом строки является либо знаком '+' (плюс) либо '-' (минус), четвёртый символ — знак '=' (равно). Из первого, третьего и пятого символов ровно два являются цифрами из диапазона от 0 до 9, и один — буквой x, обозначающей неизвестное.

Требуется написать программу, которая позволит решить данное уравнение относительно x.

Входные данные

Входной файл INPUT.TXT состоит из одной строки, в которой записано уравнение.

Выходные данные

В выходной файл OUTPUT.TXT выведите целое число — значение x.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	x+5=7	2
2	3-x=9	-6

Задача №164 Счастливым билет - 2

(Время: 1 сек. Память: 16 Мб Сложность: 26%)

Билет называется счастливым, если его можно разрезать прямой линией на две части таким образом, что оказавшиеся на них числа имеют одинаковые цифровые корни. Чтобы вычислить цифровой корень числа, его цифры складывают, если в результате получится число большее или равное 10, то цифры складывают снова и так далее, пока не получится число от 0 до 9 – это и есть цифровой корень. Например, билет с номером 0015420 является счастливым, так как разрезав его на части с числами 0015 и 420 имеем у этих чисел одинаковые цифровые корни.

Требуется написать программу, которая определит, является ли счастливым билет с заданным номером.

Входные данные

Входной файл INPUT.TXT содержит номер счастливого билета. Номер может начинаться с нулей и содержит не более 100 цифр.

Выходные данные

В выходной текстовый файл OUTPUT.TXT выведите «YES», если билет счастливый и «NO» иначе.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	0015420	YES
2	00100	NO

Задача №165 Только вправо или вниз

(Время: 1 сек. Память: 16 Мб Сложность: 32%)

Игровое поле $N \times M$ заполняется целыми числами, одно неотрицательное целое число в каждой клетке. Цель игры состоит в том, чтобы пройти по любому разрешенному пути от верхнего левого угла до правого нижнего. Целое число в каждой клетке указывает, какой длины шаг должен быть из текущей клетки. Все шаги могут быть или направо или вниз. Если в результате какого-либо шага игрок покидает пределы поля, такой шаг запрещается.

На рис. 1 приведен пример игрового поля 3×4 , где сплошная окружность показывает положение начала, а пунктирная окружность – цель. Рис. 2 показывает три возможных пути от начала до цели для рассматриваемого примера игрового поля, с удаленными промежуточными числами.

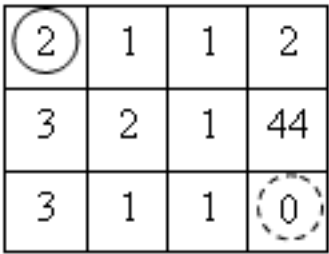


Рис. 1.

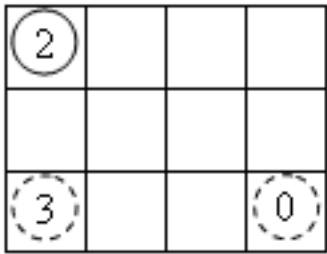
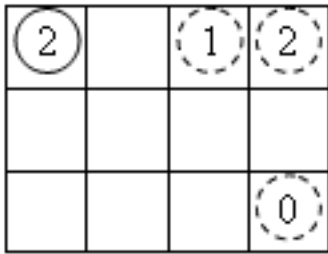
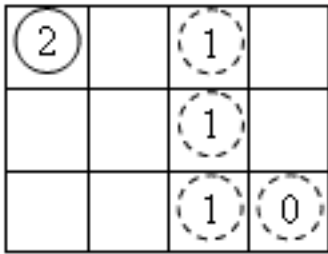


Рис. 2.



Требуется написать программу, которая определит число различных вариантов путей от верхнего левого угла до правого нижнего.

Входные данные

Входной файл INPUT.TXT содержит в первой строке размеры поля N ($1 \leq N \leq 70$) и M ($1 \leq M \leq 70$). В последующих N строках входного файла, каждая из которых описывает отдельную строку игрового поля, записаны через пробел по M целых чисел – длины шагов из клеток данной строки.

Выходные данные

Выходной файл OUTPUT.TXT должен содержать одно число - число различных вариантов путей от верхнего левого угла до правого нижнего. Для каждого поля будет менее чем 2^{31} различных путей.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	3 4 2 1 1 2 3 2 1 44 3 1 1 0	3

Задача №166 Сообщество роботов

(Время: 1 сек. Память: 16 Мб Сложность: 30%)

Сообщество роботов живет по следующим законам:

- один раз в начале года они объединяются в группы по три или пять роботов;
- за год группа из трех роботов собирает 5 новых, а группа из 5 роботов – 9 новых;
- роботы объединяются так, чтобы собрать за год наибольшее количество новых роботов;
- каждый робот живет ровно три года после сборки.

В начале первого года было K роботов и все они были только что собраны.

Требуется написать программу, которая найдет количество роботов в начале N-го года.

Входные данные

Входной файл INPUT.TXT содержит записанные через пробел числа K ($1 \leq K \leq 500$) и N ($1 \leq N \leq 100$).

Выходные данные

Выходной файл OUTPUT.TXT должен содержать одно число - количество роботов в начале N-го года. Количество роботов меньше, чем 2^{31} .

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	3 2	8
2	8 2	22

Задача №167 Количество треугольников

(Время: 1 сек. Память: 16 Мб Сложность: 51%)

Рассмотрим фигуру, аналогичную показанной на рисунке (большой равносторонний треугольник, составленный из маленьких равносторонних треугольников). На рисунке приведена фигура, состоящая из 4-х уровней треугольников.

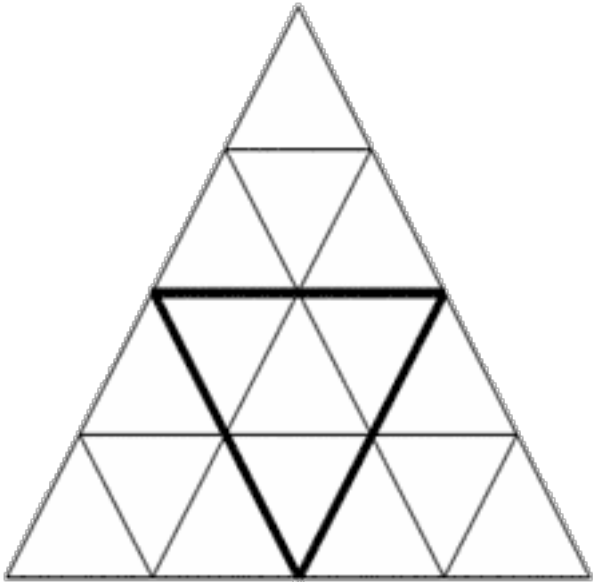
Требуется написать программу, которая будет определять, сколько всего в ней треугольников (необходимо учитывать не только "маленькие" треугольники, а вообще все треугольники — в частности, треугольник, выделенный жирным, а также вся фигура, являются интересующими нас треугольниками).

Входные данные

Входной файл INPUT.TXT содержит одно число N — количество уровней в фигуре ($1 \leq N \leq 10^5$).

Выходные данные

Выходной файл OUTPUT.TXT должен содержать одно число - количество треугольников в такой фигуре.



Примеры

№	INPUT.TXT	OUTPUT.TXT
1	1	1
2	2	5
3	4	27

Задача №168 Натуральный ряд чисел

(Время: 1 сек. Память: 16 Мб Сложность: 28%)

Натуральные числа записаны в строку без каких-либо разделителей. Начало этой строки имеет вид 123456789101112131415161718192021... .

Требуется написать программу, которая определит первое вхождение десятичной записи заданного числа N в этой строке.

Входные данные

Входной файл INPUT.TXT содержит заданное число N ($1 \leq N \leq 10^4$).

Выходные данные

Выходной файл OUTPUT.TXT должен содержать одно число — позицию, начиная с которой в строке записано первое вхождение заданного числа. Нумерация позиций начинается с единицы.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	45	4
2	101	10
3	142	73

Задача №169 Магазин

(Время: 1 сек. Память: 16 Мб Сложность: 34%)

На расстоянии N шагов от магазина стоит человек. Каждую минуту он выбирает, куда сделать шаг: к магазину или в противоположном направлении.

Требуется написать программу, которая определит, сколькими способами он может попасть в магазин, пройдя ровно K шагов и оказавшись в магазине только после выполнения последнего шага.

Входные данные

Входной файл INPUT.TXT содержит 2 числа n и k , записанные через пробел. Известно, что $1 \leq n \leq k \leq 37$.

Выходные данные

Выходной файл OUTPUT.TXT должен содержать одно число – количество способов попадания в магазин.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	2 4	2
2	5 5	1

Задача №170 Разложение числа

(Время: 1 сек. Память: 16 Мб Сложность: 35%)

Любое натуральное число можно представить в виде суммы нескольких последовательных натуральных чисел. Например, число 25 можно представить в виде суммы из одного (25), двух (12+13) или пяти (3+4+5+6+7) чисел.

Требуется написать программу, которая определит максимальное количество чисел в таком разложении.

Входные данные

Входной файл INPUT.TXT содержит одно натуральное число N ($1 \leq N \leq 10^9$).

Выходные данные

Выходной файл OUTPUT.TXT должен содержать одно натуральное число – максимальное количество чисел в разложении числа N на сумму последовательных натуральных чисел.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	2	1
2	5	2
3	25	5

Задача №171 Количество делителей

(Время: 1 сек. Память: 16 Мб Сложность: 50%)

Пусть x — натуральное число. Назовем y его делителем, если $1 \leq y \leq x$ и остаток от деления x на y равен нулю.

Задано число x . Найдите количество его делителей.

Входные данные

Входной файл INPUT.TXT содержит заданное число x ($1 \leq x \leq 10^{18}$). Все простые делители числа x не превосходят 1000.

Выходные данные

В выходной файл OUTPUT.TXT выведите ответ на задачу.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	12	6
2	239	2

Задача №172 Деление с остатком

(Время: 1 сек. Память: 16 Мб Сложность: 39%)

Заданы два числа: N и K . Необходимо найти остаток от деления N на K .

Входные данные

Входной файл INPUT.TXT содержит два целых числа: N и K ($1 \leq N \leq 10^{100}$, $1 \leq K \leq 10^9$).

Выходные данные

В выходной файл OUTPUT.TXT выведите остаток от деления N на K .

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	239 16	15
2	4638746747645731289347483927 6784789	1001783

Задача №173 Число - палиндром

(Время: 1 сек. Память: 16 Мб Сложность: 29%)

Напомним, что палиндромом называется строка, одинаково читающаяся с обеих сторон. Например, строка «АВВА» является палиндромом, а строка «АВС» - нет.

Необходимо определить, в каких системах счисления с основанием от 2 до 36 представление заданного числа N является палиндромом.

В системах счисления с основанием большим 10 в качестве цифр используются буквы латинского алфавита: А, В, ..., Z. Например, $A_{11} = 10_{10}$, $Z_{36} = 35_{10}$.

Входные данные

Входной файл INPUT.TXT содержит заданное число N в десятичной системе счисления ($1 \leq N \leq 10^9$).

Выходные данные

Если соответствующее основание системы счисления определяется единственным образом, то выведите в первой строке выходного файла OUTPUT.TXT слово «unique», если оно не единственно — выведите в первой строке выходного файла слово «multiple». Если же такого основания системы счисления не существует — выведите в первой строке выходного файла слово «none».

В случае существования хотя бы одного требуемого основания системы счисления выведите через пробел в возрастающем порядке во второй строке выходного файла все основания системы счисления, удовлетворяющие требованиям.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	123	unique 6
2	111	multiple 6 10 36
3	102892748	none

Задача №174 Свадьба

(Время: 1 сек. Память: 16 Мб Сложность: 32%)

Одна предприимчивая и очень симпатичная дамочка с прелестнейшим именем Горгона решила заработать себе денег на роскошную жизнь. N молодых людей так влюблены в нее, что предложили руку и сердце. К несчастью для них, Горгона видит в них только мешок с деньгами. Она планирует выйти замуж и почти сразу же развестись с некоторыми из молодых людей ради денежной выгоды. Все, что ей нужно, это подзаработать как можно больше денег (и уж, конечно, остаться незамужней). По законам этой прекрасной страны при разводе каждый из супругов получает половину всего имущества.

Вы планируете опубликовать статью, в которой опишете всю подлость и меркантильность этой особы. Для того чтобы статья получилась особенно красочной, нужно указать максимальную сумму денег, которую сможет получить Горгона.

Входные данные

В первой строке входного файла INPUT.TXT записано целое число N — количество молодых людей, без памяти влюбленных в Горгону ($1 < N \leq 40$). Далее следует N чисел — сумма денег на счету каждого молодого человека. В последней строке записано целое число A — сумма денег на счету Горгоны. Суммы денег на счету — целые неотрицательные числа, не превосходящие 10^9 .

Выходные данные

В выходной файл OUTPUT.TXT выведите единственное число — максимальную сумму денег, которой сможет обладать Горгона после своей махинации. Ответ выводите с точностью до шести знаков ровно в формате без мантиссы.

Примеры

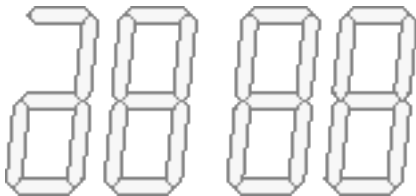
№	INPUT.TXT	OUTPUT.TXT
1	2 5 10 5	7.500000
2	3 1 3 2 0	2.125000

Задача №175

Наручные часы

(Время: 1 сек. Память: 16 Мб Сложность: 37%)

Вы приобрели новые электронные наручные часы с жидкокристаллическим дисплеем. Дисплей отображает часы и минуты с помощью четырех элементов, каждый из которых отображает одну цифру.



Три из них состоят из семи полосок, каждая из которых может быть либо белой (неотличимой от фона), либо черной. Вид такого элемента и отображаемые им цифры показаны на рисунке:



Четвертый элемент предназначен для отображения старшей цифры часа. Если она равна нулю, то элемент полностью неактивен (все полоски белые), иначе показывается соответствующая цифра. Вот как выглядит этот элемент с цифрами:



Вам хочется проверить: все ли в порядке с новым приобретением, а именно, нет ли таких полосок в каком-либо из элементов, которые либо всегда белые, либо всегда черные. Вы хотите начать проверку в некоторое начальное время. Требуется определить, сколько Вам потребуется минут для убеждения в исправности часов.

Входные данные

В первой строке входного файла INPUT.TXT находится время начала проверки в формате HH:MM. Часы и минуты записаны с лидирующими нулями, если таковые имеются. ($00 \leq HH \leq 23$, $00 \leq MM \leq 59$).

Выходные данные

В выходной файл OUTPUT.TXT выведите минимальное число минут, необходимое для проверки Ваших часов, если она началась в заданное время.

Примеры

--	--	--

№	INPUT.TXT	OUTPUT.TXT
1	00:00	1200
2	02:39	1041

Задача №176
Скобочки

(Время: 1 сек. Память: 16 Мб Сложность: 69%)

Строка, состоящая из символов «(» и «)», называется скобочной последовательностью. Скобочная последовательность называется правильной, если она может быть получена из некоторого корректного арифметического выражения удалением всех символов, кроме скобок. Например, правильная скобочная последовательность «(())()» может быть получена из выражения «(2-(3+4)*6)*(1+1)».

Глубиной правильной скобочной последовательности называется максимальная разность между количеством открывающихся и закрывающихся скобок в префиксе последовательности (префиксом строки S называется строка, которую можно получить из S удалением некоторого количества последних символов, например, префиксами строки «ABCAВ» являются строки «», «A», «AB», «ABC», «ABCA» и «ABCAВ»). Например, глубина последовательности «()()()» равна двум, т.к. префикс «()()()» имеет 4 открывающиеся и 2 закрывающиеся скобки.

Требуется написать программу, определяющую по заданным значениям N и K количество правильных скобочных последовательностей с N открывающимися скобками, которые имеют глубину, равную K.

Входные данные

Входной файл INPUT.TXT содержит в одной строке целые числа N и K ($1 \leq K \leq N \leq 50$), разделенные пробелом.

Выходные данные

Выходной файл OUTPUT.TXT должен содержать одно число — количество правильных скобочных последовательностей с n открывающимися скобками, которые имеют глубину k.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	3 2	3
2	37 23	203685956218528

Задача №177
Склад

(Время: 1 сек. Память: 16 Мб Сложность: 58%)

На роботизированном складе имеется n отсеков, в которые робот может размещать грузы. Отсек с номером i имеет вместимость c_i. Груз с номером i имеет размер s_i, поступает на склад в момент времени a_i и забирается со склада в момент времени d_i. Вместимость отсека и размер груза имеют одну и ту же размерность. Если в отсеке с вместимостью c находится несколько грузов с суммарным размером d, то свободное место в этом отсеке равно c – d.

Когда груз с номером i поступает на склад, робот сначала пытается найти отсек, в котором достаточно свободного места для размещения этого груза. Если отсеков, в которых достаточно

свободного места, несколько, то робот помещает груз в тот из них, в котором свободного места меньше. Если и таких отсеков несколько, то робот выбирает отсек с минимальным номером.

Если отсеков с достаточным количеством свободного места нет, робот пытается переместить грузы, уже расположенные в отсеках. Для этого он пытается найти такой отсек и такой груз в нем, что перемещение его в другой отсек обеспечивает достаточное количество свободного места для размещения поступившего груза. Если таких вариантов перемещения грузов несколько, то выбирается тот вариант, в котором потребуется перемещение груза с минимальным размером. Если и таких вариантов несколько, то выбирается вариант перемещения, при котором в отсеке, из которого перемещается груз, свободное место после перемещения этого груза будет минимально, а при прочих равных условиях — тот вариант, при котором в отсеке, куда осуществляется перемещение, свободное место после этого перемещения будет также минимально. Если и после этого остается более одного варианта, то выбирается тот вариант, при котором номер перемещаемого груза минимален и номер отсека, в который он перемещается, — также минимален. Если варианта с перемещением одного груза найти не удалось, то груз не принимается на склад.

Требуется написать программу, которая по списку грузов, поступающих для размещения на складе, выводит последовательность действий, выполняемых роботом.

Входные данные

Первая строка входного файла содержит два целых числа: n — количество отсеков, и m — количество грузов ($1 \leq n \leq 10, 1 \leq m \leq 100$). Вторая строка содержит n целых чисел c_i , определяющих вместимости отсеков ($1 \leq c_i \leq 10^9$). Последующие m строк описывают грузы: каждый груз описывается тремя целыми числами: своим размером s_i , временем поступления на склад a_i и временем, когда его забирают со склада d_i ($1 \leq s_i \leq 10^9, 1 \leq a_i < d_i \leq 1000$, все времена во входном файле различны, грузы упорядочены по возрастанию времени поступления на склад). Все числа в строках разделены пробелом.

Выходные данные

Выведите последовательность действий робота в том порядке, в котором они выполняются. Следуйте формату выходного файла, приведенного в примере. Возможны следующие сообщения:

- put cargo X to cell Y - разместить груз с номером X в отсеке с номером Y;
- move cargo X from cell Y to cell Z - переместить груз с номером X из отсека с номером Y в отсек с номером Z;
- take cargo X from cell Y - взять груз с номером X из отсека с номером Y.
- cargo X cannot be stored - груз X невозможно переместить

Пример

№	INPUT.TXT	OUTPUT.TXT
1	1 1 3 3 1 2	put cargo 1 to cell 1 take cargo 1 from cell 1
2	3 5 3 2 10 1 1 6 3 2 8 9 3 5 2 4 9 12 7 10	put cargo 1 to cell 2 put cargo 2 to cell 1 put cargo 3 to cell 3 move cargo 1 from cell 2 to cell 3 put cargo 4 to cell 2 take cargo 3 from cell 3 take cargo 1 from cell 3

	cargo 5 cannot be stored take cargo 2 from cell 1 take cargo 4 from cell 2
--	--

Задача №178

Преобразование последовательности

(Время: 1 сек. Память: 16 Мб Сложность: 42%)

Задана последовательность, содержащая n целых чисел. Необходимо найти число, которое встречается в этой последовательности наибольшее количество раз, а если таких чисел несколько, то найти минимальное из них, и после этого переместить все такие числа в конец заданной последовательности. Порядок расположения остальных чисел должен остаться без изменения.

Например, последовательность 1, 2, 3, 2, 3, 1, 2 после преобразования должна превратиться в последовательность 1, 3, 3, 1, 2, 2, 2.

Требуется написать программу, которая решает данную задачу.

Входные данные

Первая строка входного файла INPUT.TXT содержит число n — количество чисел во входной последовательности ($3 \leq n \leq 200000$). Следующая строка содержит входную последовательность, состоящую из n целых чисел, не превышающих по модулю 10^6 . Все числа в строке разделены пробелом.

Выходные данные

В выходной файл OUTPUT.TXT выводится последовательность чисел, которая получается в результате названного преобразования. Все числа в последовательности должны быть разделены пробелом.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	7 1 2 3 2 3 1 2	1 3 3 1 2 2 2

Задача №179

Последовательность

(Время: 1 сек. Память: 16 Мб Сложность: 47%)

Рассмотрим числовую последовательность, первоначально состоящую из двух единиц: 1, 1. Далее на каждом последующем шаге будем вставлять между соседними элементами их сумму. В примере добавляемые элементы выделены:

Номер шага	Последовательность
0	1, 1
1	1, 2, 1
2	1, 3, 2, 3, 1
3	1, 4, 3, 5, 2, 5, 3, 4, 1

Требуется написать программу, которая подсчитает сумму членов последовательности, построенной за K шагов.

Входные данные

Входной файл INPUT.TXT содержит одно натуральное число K ($0 \leq K \leq 100$) – номер последнего шага.

Выходные данные

Выходной файл OUTPUT.TXT должен содержать одно натуральное число – сумму элементов последовательности, построенной за K шагов.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	3	28
2	10	59050

Задача №180 Счастливая страница

(Время: 1 сек. Память: 16 Мб Сложность: 46%)

В новом выпуске Большой Галактической Энциклопедии N страниц. Петя считает страницу счастливой, если произведение цифр, входящих в ее номер, равно K . Например, если $N=100$, то для $K=42$ есть счастливая страница (например, с номером 76), а для $K=128$ счастливой страницы нет.

Требуется написать программу, которая поможет Пете определить, есть ли счастливые страницы в новом выпуске энциклопедии.

Входные данные

Входной текстовый файл INPUT.TXT содержит числа N ($1 \leq N \leq 10^9$) и K ($1 \leq K \leq 10^9$), записанные через пробел.

Выходные данные

Выходной текстовый файл OUTPUT.TXT должен содержать «YES», если счастливые страницы есть, и «NO» иначе.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	100 42	YES
2	100 128	NO

Задача №181 Космический мусорщик

(Время: 1 сек. Память: 16 Мб Сложность: 52%)

В околоземном космическом пространстве накопилось много мусора, поэтому ученые сконструировали специальный аппарат ловушку для космического мусора. Аппарат должен двигаться по достаточно сложной траектории, сжигая по пути мусор. Ловушка может передвигаться в пространстве по 6 направлениям: на север (N), на юг (S), на запад (W), на восток (E), вверх (U) и вниз (D). Движением ловушки управляет процессор. Программа движения задается

шестью правилами движения, которые соответствуют каждому из указанных направлений. Каждое такое правило представляет собой строку символов из множества {N, S, W, E, U, D}.

Команда ловушки состоит из символа направления и целого положительного числа M. Если параметр больше 1, то ловушка перемещается на один метр в направлении, которое указано в команде, а затем последовательно выполняет команды, заданные правилом для данного направления, с параметром меньше на 1. Если же параметр равен 1, то просто перемещается на один метр в указанном направлении.

Пусть, например, заданы правила, отраженные в таблице справа. Тогда при выполнении команды S(3) мусорщик сначала переместится на 1 метр в направлении S, а потом выполнит последовательно команды N(2), U(2), S(2), D(2), D(2), U(2), S(2), E(2).

Если далее проанализировать действия мусорщика, получим, что в целом он совершит ровно 34 перемещения.

Направление	Правило
N	N
S	NUSDDUSE
W	UEWWD
E	
U	U
D	WED

Входные данные

Первые шесть строк входного файла INPUT.TXT задают правила для команд с направлением N, S, W, E, U и D соответственно. Каждая строка содержит не более 100 символов (и может быть пустой). Следующая строка содержит команду ловушки: сначала символ из множества {N, S, W, E, U, D}, затем пробел и параметр команды – целое положительное число, не превышающее 100.

Выходные данные

Выведите в выходной файл OUTPUT.TXT единственное число - количество перемещений, которое совершит ловушка. Гарантируется, что ответ не превышает 10⁹.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	N NUSDDUSE UEWWD U WED S 3	34

Задача №182 Прямоугольник - 2

(Время: 1 сек. Память: 16 Мб Сложность: 27%)

Заданы координаты трех вершин прямоугольника. Необходимо определить координаты четвертой вершины.

Входные данные

Во входном файле INPUT.TXT записаны через пробел координаты трех вершин прямоугольника в произвольном порядке в формате x₁ y₁ x₂ y₂ x₃ y₃. Все числа целые, не

превосходящие 1000 по абсолютной величине.

Выходные данные

В выходной файл OUTPUT.TXT нужно вывести через пробел координаты четвертой вершины прямоугольника.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	0 3 0 0 5 0	5 3
2	1 4 8 3 7 6	2 1

Задача №183 Энты

(Время: 1 сек. Память: 16 Мб Сложность: 39%)

Энты были созданы в Первоначальную эпоху вместе с другими обитателями Средиземья. Эльфийские легенды гласят, что когда Варда зажгла звёзды и пробудились Эльфы, вместе с ними пробудились и Энты в Великих Лесах Арды.

Когда Энты пришли в Арду, они ещё не умели говорить — этому искусству их обучали Эльфы, и Энтам это ужасно нравилось. Им доставляло удовольствие изучать разные языки, даже щебетание Людей.

Эльфы выработали хорошую технику обучения энтов своему языку. Первый энт, которого обучили эльфы, выучил всего два слова — «tancave» (да) и «la» (нет). Обученный энт выбрал одного старого и одного молодого энта, не умеющих говорить, и обучил их всем словам, которые знал сам. Затем обучение этих двух энтов продолжили сами эльфы. Каждый обучившийся у эльфов энт снова выбирал из неговорящих сородичей одного старого и одного молодого, обучал их всем словам, которые знал, передавал эльфам и так далее.

Выяснилось, что более молодые энты выучивали у эльфов ещё ровно столько же слов, сколько они узнали от обучавшего их энта. А вот более старые, уже склонные к одревенению энты, пополняли свой запас всего лишь одним словом. После обучения у эльфов энты до конца света уже не могли выучить ни одного нового слова.

Общее число энтов в Средиземье больше, чем вы думаете. Интересно, а сколько из них знают ровно 150 квенийских слов? Похожую задачу вам предстоит решить.

Входные данные

Входной файл INPUT.TXT содержит натуральные числа K и P ($K \leq 10^6$; $1 \leq P \leq 10^9$), записанные через пробел.

Выходные данные

Мы понимаем, что число энтов, знающих в точности K слов, может быть слишком велико, поэтому просим вывести в выходной файл OUTPUT.TXT лишь количество энтов, знающих ровно K слов, по модулю P.

Примеры

№	INPUT.TXT	OUTPUT.TXT

1	4 10	2
2	8 10	5
3	360 1000	179

Задача №184

Рабочее время

(Время: 1 сек. Память: 16 Мб Сложность: 26%)

Иван Иванович – очень ответственный, но очень рассеянный человек. Поэтому когда он начинает очередное дело, он на отдельном листочке пишет дату и время начала (например, 29.01. 10:30), а когда заканчивает, то так же на отдельном листочке – дату и время окончания (например, 02.02. 12:15). Листочки аккуратно укладываются в стопку один на другой. А так как одновременно Иван Иванович может заниматься только одним делом, то листочки однозначно упорядочены в стопке: листок начала какого-то дела, листок окончания этого дела, листок начала, листок окончания... и т.д. Дело начинается в начале минуты, указанной в листочке начала этого дела, а заканчивается в конце минуты, указанной на листочке окончания. Иван Иванович ходит на работу каждый день и его рабочий день продолжается с 10:00 до 18:00. Таким образом, пара листочков «18.11. 15:13» — «20.11. 10:27» была написана при начале и окончании дела длительностью 11ч.15м.

Однажды в конце декабря уборщица Дуся нечаянно уронила эту стопку на пол и, не зная важной закономерности их укладки, собрала листочки обратно в каком-то произвольном порядке. Иван Иванович обнаружил этот прискорбный факт только 31 декабря, когда ему надо было произвести учет своего рабочего времени за год. Год был невисокосный.

Помогите Ивану Ивановичу найти его суммарные затраты времени за год.

Входные данные

Входной файл INPUT.TXT содержит в первой строке число листочков K, (K – четное число, не большее 5000). Далее записаны K строк с данными на листочках в формате DD.MM. hh:mm, где DD – число, MM – номер месяца, hh – часы и mm – минуты.

Выходные данные

Выходной файл OUTPUT.TXT должен содержать запись вида h:mm – количество часов и минут, отработанных Иваном Ивановичем. При этом число h >= 0 выводится без ведущих нулей, а число 0 <= mm <= 59 выводится с ведущими нулями.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	<div>4</div> <div>15.01. 17:00</div> <div>16.01. 12:00</div> <div>11.02. 14:00</div> <div>30.01. 10:00</div>	103:02

Задача №185

Скачки

(Время: 1 сек. Память: 16 Мб Сложность: 32%)

Иван Иванович любит ходить на скачки, надеясь на них заработать кругленькую сумму. Ему

приглянулась лошадь с номером K, и он решил проверить, сможет ли она выиграть у всех остальных лошадей. Иван Иванович раздобыл информацию, в которой для некоторых пар лошадей сообщается, какая из этих лошадей быстрее. Также он узнал, что у всех лошадей разные скорости.

Требуется написать программу, которая поможет Ивану Ивановичу точно определить может ли выиграть выбранная им лошадь.

Входные данные

Входной файл INPUT.TXT содержит в первой строке два целых числа N ($1 \leq N \leq 100$) и K ($1 \leq K \leq N$), где N – количество лошадей, принимающих участие в скачках, K – номер лошади, на которую хочет сделать ставку Иван Иванович. Следующие строки содержат по два числа X и Y ($1 \leq X, Y \leq N$), обозначающие, что лошадь с номером X быстрее лошади с номером Y. Пары X и Y не повторяются. Набор данных завершается строкой, содержащей единственный ноль. Эту строку обрабатывать не надо.

Гарантируется, что информация, раздобытая Иваном Ивановичем, корректна.

Выходные данные

Выходной файл OUTPUT.TXT должен содержать слово «Yes», если Иван Иванович уверен в своем выигрыше и «No» в противном случае.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	3 1 1 2 1 3 0	Yes
2	3 2 2 3 0	No
3	4 2 3 1 2 3 0	No

Задача №186 Субботник

(Время: 1 сек. Память: 16 Мб Сложность: 44%)

В этом году Иван Иванович решил отметить приход осени субботником, чтобы убрать весь мусор во дворе дома номер 31 по улице Осенней. На субботник он пригласил N знакомых старушек, живущих в том же самом доме. Однако, в самом начале мероприятия выяснилось, что по одиночке старушки работают плохо, так как им хочется во время работы еще и поговорить друг с другом.

Иван Иванович подумал и принял волевое решение разбить старушек на группы так, чтобы в каждой группе было не менее 2 старушек. Старушки отличаются друг от друга уровнем разговорчивости, и если в одну группу попадут две старушки, у одной из которых маленький уровень разговорчивости, а у второй - большой, то они не могут поговорить друг с другом и работа будет стопориться.

Назовем разговорчивостью группы разность между максимальным и минимальным уровнями

разговорчивости старушек в группе. Например, если уровни разговорчивости старушек в группе равны 7, 3 и 11, то разговорчивость группы равна $11 - 3 = 8$. Разговорчивостью разбиения старушек на группы назовем максимальную из разговорчивостей групп, входящих в разбиение.

Требуется написать программу, которая поможет Ивану Ивановичу найти разбиение старушек на группы, разговорчивость которого минимальна.

Входные данные

Входной файл INPUT.TXT содержит в первой строке число N ($2 \leq N \leq 1000$) – количество старушек. Во второй строке записано N чисел от 1 до 10^9 – разговорчивости старушек.

Выходные данные

Выходной текстовый файл OUTPUT.TXT должен содержать одно целое число, равное минимально возможной разговорчивости разбиения старушек на группы.

Примеры

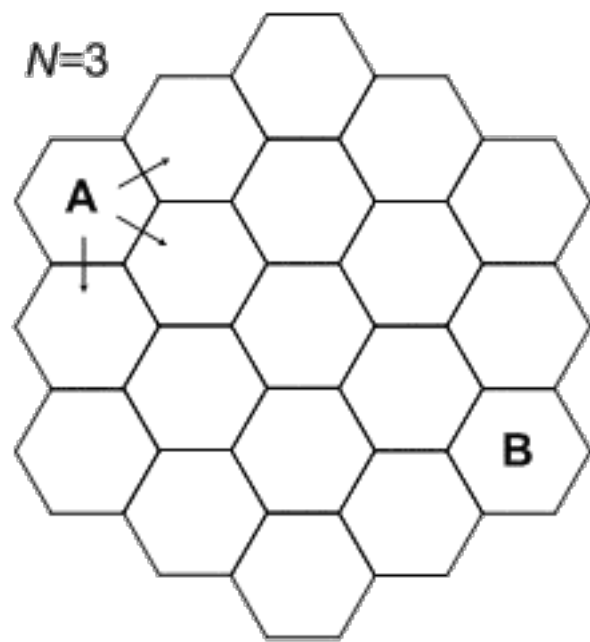
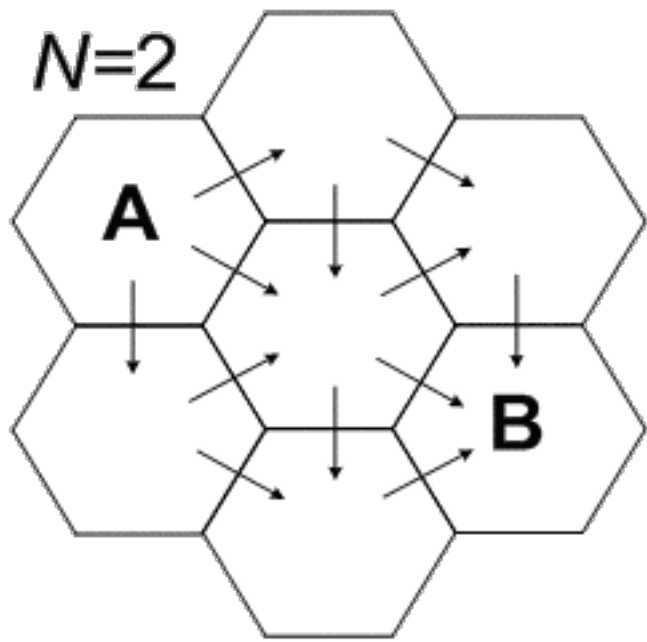
№	INPUT.TXT	OUTPUT.TXT
1	2 1 1000000000	999999999
2	3 1 2 3	2
3	8 1 10 100 1000 1000 100 10 1	0
4	10 258 740 156 244 458 680 390 694 844 817	102

Задача №187 Пчелка

(Время: 1 сек. Память: 16 Мб Сложность: 53%)

Представьте себе пчелиные соты – поле из шестиугольных клеток со стороной N . В верхней левой клетке A находится пчелка. За один ход она может переползти на клетку вниз, на клетку вниз-вправо или на клетку вверх-вправо (вверх и влево пчелка не ползает).

Требуется написать программу, которая найдет количество способов, которыми пчелка может доползти из клетки A в противоположную клетку B .



Входные данные

Входной файл INPUT.TXT содержит единственное число N – размеры шестиугольного поля ($2 \leq N \leq 12$).

Выходные данные

Выходной файл OUTPUT.TXT должен содержать единственное целое число – количество способов.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	2	11
2	3	291

Задача №188 День рождения

(Время: 1 сек. Память: 16 Мб Сложность: 75%)

Иван Иванович пригласил на свой день рождения много гостей. Он написал на карточках фамилии всех гостей и разложил эти карточки на столе, полагая, что каждый гость сядет там, где обнаружит карточку со своей фамилией (фамилии у всех гостей различны). Однако гости не обратили внимания на карточки и сели за стол в произвольном порядке. При этом Иван Иванович с удивлением обнаружил, что ни один гость не сел на предназначенное ему место.

Требуется написать программу, которая найдет сколькими способами можно рассадить гостей так, чтобы ни один из них не сидел там, где лежала карточка с его фамилией.

Входные данные

Во входном файле INPUT.TXT задано целое число N – количество гостей ($1 \leq N \leq 100$).

Выходные данные

Выходной файл OUTPUT.TXT должен содержать одно целое число – количество способов рассадить гостей.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	1	0
2	2	1
3	5	44
4	20	895014631192902121

Задача №189 Перестановка по номеру

(Время: 1 сек. Память: 16 Мб Сложность: 47%)

Перестановкой из N элементов называется упорядоченный набор из N различных чисел от 1 до N . Количество всех перестановок порядка N равно $P_N = N!$

Требуется найти перестановку по ее номеру в лексикографическом порядке (по алфавиту). Например, для $N=3$ лексикографический порядок перестановок выглядит следующим образом:

(1,2,3), (1,3,2), (2,1,3), (2,3,1), (3,1,2), (3,2,1).

Таким образом, перестановка (2,3,1) имеет номер 4 в этой последовательности.

Входные данные

В первой строке входного файла INPUT.TXT записано число N ($1 \leq N \leq 12$) - количество элементов в перестановке, во второй - число K ($1 \leq K \leq N!$) - номер перестановки.

Выходные данные

В выходной файл OUTPUT.TXT выведите через пробел N чисел - искомую перестановку.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	1 1	1
2	3 2	1 3 2

Задача №190 По размещению!

(Время: 1 сек. Память: 16 Мб Сложность: 56%)

Перестановкой из N элементов называется упорядоченный набор из N различных чисел от 1 до N .

Размещением порядка K называют подмножество элементов некоторой перестановки порядка N . Например, (1, 3) – размещение порядка 2 для перестановки (1, 2, 3) порядка 3.

Требуется по заданному размещению определить его позицию в лексикографическом порядке всех возможных размещений, образованных из всевозможных перестановок порядка N .

Например, лексикографическая последовательность всевозможных размещений для $K=2$ и $N=3$

выглядит следующим образом:

(1,2), (1,3), (2,1), (2,3), (3,1), (3,2)

Таким образом, перемещение (2,3) имеет номер 4 в этой последовательности.

Входные данные

В первой строке входного файла INPUT.TXT находятся числа N и K ($1 \leq K \leq N \leq 12$). Во второй строке записаны K чисел из диапазона от 1 до N - размещение.

Выходные данные

В выходной файл OUTPUT.TXT выведите единственное число - номер данного размещения.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	3 2 3 2	6
2	6 4 1 3 2 5	14

Задача №191 Гладкие числа

(Время: 1 сек. Память: 16 Мб Сложность: 60%)

Назовем число *гладким*, если его цифры, начиная со старшего разряда, образуют неубывающую последовательность. Упорядочим все такие числа в возрастающем порядке и присвоим каждому номер.

Вам требуется по номеру N вывести N-ое гладкое число.

Входные данные

Во входном файле INPUT.TXT содержится номер N ($1 \leq N \leq 2147483647$).

Выходные данные

Выходной файл OUTPUT.TXT должен содержать искомое N-е гладкое число.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	1	1
2	11	12
3	239	1135

Задача №192 Следующая перестановка ...

(Время: 1 сек. Память: 16 Мб Сложность: 46%)

Перестановкой из N элементов называется упорядоченный набор из N различных чисел от 1 до

N.

Найдите по заданной перестановке следующую в лексикографическом порядке (будем считать, что за перестановкой (N, N-1, ... , 3, 2, 1) следует тождественная перестановка, то есть (1, 2, 3, ... , N)).

Входные данные

В первой строке входного файла INPUT.TXT содержится число N ($1 \leq N \leq 10^4$). Во второй строке содержится перестановка (последовательность натуральных чисел от 1 до N, разделенных пробелами).

Выходные данные

Выходной файл OUTPUT.TXT должен содержать искомую перестановку.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	1 1	1
2	5 2 4 5 3 1	2 5 1 3 4

Задача №193 Поиск прямоугольников

(Время: 0,5 сек. Память: 16 Мб Сложность: 34%)

На поле NхM клеток (N строк и M столбцов) положили K прямоугольников один поверх другого в случайном порядке. Длины сторон прямоугольников выражаются целым числом клеток. Прямоугольники не выходят за границы поля. Границы прямоугольников совпадают с границами клеток поля.

Получившуюся ситуацию записали в таблицу чисел (каждой клетке поля соответствует клетка таблицы). Если клетка поля не закрыта прямоугольником, то в соответствующую клетку таблицы записали число 0. Если же клетка закрыта одним или несколькими прямоугольниками, то в соответствующую клетку таблицы записали число, соответствующее номеру самого верхнего прямоугольника, закрывающего эту клетку.

Требуется написать программу, которая определит положение и размеры прямоугольников. Гарантируется, что во входных данных содержится информация, которой достаточно для однозначного определения размеров прямоугольников.

Входные данные

Входной файл INPUT.TXT содержит в первой строке целые числа N, M, K ($1 \leq N \leq 200$, $1 \leq M \leq 200$, $1 \leq K \leq 255$). Далее следует N строк по M чисел в каждой — содержимое таблицы. Все числа в таблице целые, находятся в диапазоне от 0 до K включительно.

Выходные данные

Выходной файл OUTPUT.TXT должен содержать K строк. Каждая строка должна описывать соответствующий ее номеру прямоугольник четырьмя числами X1 Y1 X2 Y2 (X1 и Y1 должны описывать координаты левого нижнего угла прямоугольника, а X2 и Y2 — координаты правого верхнего угла). Числа должны разделяться пробелом.

0	2	2	2	2
0	2	2	2	2
1	1	2	2	2
1	1	0	0	0

Начало координат расположено в левом нижнем углу таблицы. Таким образом, координаты левого нижнего угла поля — (0,0), правого верхнего — (M,N).

Пример

№	INPUT.TXT	OUTPUT.TXT
1	4 5 2 0 2 2 2 2 0 2 2 2 2 1 1 2 2 2 1 1 0 0 0	0 0 2 2 1 1 5 4

Задача №194 Фотограф-зануда

(Время: 1 сек. Память: 16 Мб Сложность: 55%)

Однажды глава семейства заказал фотографию своей большой семьи, состоящей из N человек, возраст которых 1 год, 2 года, ..., N-1 лет и N лет. На фотографии должны присутствовать все родственники, и для этого они должны расположиться в один ряд. Сначала было решено расположить родственников по старшинству, начиная с самого младшего. Но фотограф сказал, что, возможно, на фото это будет выглядеть неестественно. Тогда было решено использовать следующее размещение:

- 1. слева сидит ребенок возрастом в 1 год
- 2. разность возрастов двух соседних родственников не превышает 2 года

Действительно, на фотографии, таким образом, все будут все равно выглядеть, будто расположенные по старшинству (ведь среди людей возрастом, к примеру, 25 и 27 лет не так легко определить старшего). Способов такой посадки существует, понятно, несколько. Фотограф снял все такие способы. Сколько же фотографий получилось в итоге?

Входные данные

Во входном файле INPUT.TXT содержится число N (1 <= N <= 55) – количество членов большой семьи.

Выходные данные

Выходной файл OUTPUT.TXT должен содержать искомое число фотографий.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	4	4
2	7	14

Задача №195

Эния

(Время: 1 сек. Память: 16 Мб Сложность: 3%)

Неспокойно сейчас на стапелях шестого дока межгалактического порта планеты Торна. Всего через месяц закончится реконструкция малого броненосущего корвета “Эния”. И снова этому боевому кораблю и его доблестной команде предстоят тяжелые бои за контроль над плутониевыми рудниками Сибелиуса. Работа не прекращается ни на секунду, лазерные сварочные аппараты работают круглые сутки. От непрерывной работы плавятся шарниры роботов-ремонтников. Но задержаться нельзя ни на секунду.

И вот в этой суматохе обнаруживается, что термозащитные панели корвета вновь требуют срочной обработки сульфидом тория. Известно, что на обработку одного квадратного метра панели требуется 1 нанограмм сульфида. Всего необходимо обработать N прямоугольных панелей размером A на B метров. Вам необходимо как можно скорее подсчитать, сколько всего сульфида необходимо на обработку всех панелей “Энии”. И не забудьте, что панели требуют обработки с обеих сторон.

Входные данные

Во входном файле INPUT.TXT содержатся 3 целых положительных числа N ($N \leq 100$), A ($A \leq 100$), B ($B \leq 100$)

Выходные данные

В выходной файл OUTPUT.TXT нужно вывести единственное число – вес необходимого для обработки сульфида тория в нанограммах.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	5 2 3	60
2	14 23 5	3220

Задача №196

Спираль

(Время: 1 сек. Память: 16 Мб Сложность: 38%)

Требуется совершить обход квадратной матрицы по спирали так, как показано на рисунке справа: заполнение происходит с единицы из левого верхнего угла и заканчивается в центре числом N^2 , где N – порядок матрицы.

1	2	3	4	5
16	17	18	19	6
15	24	25	20	7
14	23	22	21	8
13	12	11	10	9

Входные данные

Во входном файле INPUT.TXT задано натуральное число N – размер квадратной матрицы ($N \leq 100$).

Выходные данные

В выходной файл OUTPUT.TXT выведите матрицу, заполненную числами от 1 до N^2 по спирали, при этом между числами может быть любое количество пробелов.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	5	1 2 3 4 5 16 17 18 19 6 15 24 25 20 7 14 23 22 21 8 13 12 11 10 9

Задача №197

Змейка

(Время: 1 сек. Память: 16 Мб Сложность: 40%)

Требуется заполнить змейкой квадратную матрицу так, как показано на рисунке справа: заполнение происходит с единицы из левого верхнего угла и заканчивается в правом нижнем числом N², где N – порядок матрицы.

1	3	4	10
2	5	9	11
6	8	12	15
7	13	14	16

Входные данные

Во входном файле INPUT.TXT задано натуральное число N – размер квадратной матрицы (N<=100).

Выходные данные

В выходной файл OUTPUT.TXT выведите матрицу, заполненную числами от 1 до N² змейкой, при этом между числами может быть любое количество пробелов.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	4	1 3 4 10 2 5 9 11 6 8 12 15 7 13 14 16

Задача №198

Система линейных уравнений

(Время: 1 сек. Память: 16 Мб Сложность: 57%)

Требуется решить невырожденную систему, состоящую из N линейных уравнений с N неизвестными:

$$\begin{cases} a_{11}x_1+a_{12}x_2+\dots+a_{1n}x_n=b_1 \\ a_{21}x_1+a_{22}x_2+\dots+a_{2n}x_n=b_2 \\ \dots \\ a_{i1}x_1+a_{i2}x_2+\dots+a_{in}x_n=b_i \\ \dots \\ a_{n1}x_1+a_{n2}x_2+\dots+a_{nn}x_n=b_n \end{cases}$$

Входные данные

В первой строке входного файла INPUT.TXT задано натуральное число N – ранг системы,

далее следуют N строк, каждая из которых состоит из N+1 целых чисел: коэффициенты i-й строки уравнения – N чисел a_{ij} и b_i . ($N \leq 100$, $|a_{ij}| < 10$, $|b_i| < 10^4$).

Выходные данные

В единственную строку выходного файла OUTPUT.TXT нужно вывести через пробел корни приведенной во входном файле системы линейных уравнений. Корни следует выводить в порядке возрастания их номеров (x_1, x_2, \dots, x_n). Гарантируется, что все корни целые и не превосходят значения 10 по абсолютной величине.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	2 5 9 62 9 -3 54	7 3
2	3 7 -9 1 2 0 6 5 -15 3 -3 2 -9	8 5 -9

Задача №199 Римские числа

(Время: 1 сек. Память: 16 Мб Сложность: 50%)

Необходимо сократить дробь, записанную в римской системе счисления. Напомним, что в римской записи используются символы M, D, C, L, X, V и I. Приведем таблицу с примерами перевода римских чисел в арабскую систему:

I – 1	VII – 7	XLVI – 46	CCCII – 302
II – 2	VIII – 8	L – 50	CDXLI – 441
III – 3	IX – 9	LXXV – 75	ID – 499
IV – 4	X – 10	XCII – 92	D – 500
V – 5	XVIII – 18	IC – 99	DCXCV – 695
VI – 6	XXXI – 31	C – 100	CM – 900

Входные данные

Во входном файле INPUT.TXT записана дробь в римской системе счисления. Формат записи считается корректным, если запись представляет собой: римское число, деление, римское число (без пробелов), и каждое из чисел находится в диапазоне от 1 до 999.

Выходные данные

В выходной файл OUTPUT.TXT выведите сокращенную дробь в римской системе счисления. В тех случаях, когда первое число делится на второе, следует выводить результат в виде только одного римского числа. В том случае, когда во входных данных содержится ошибка, следует вывести ERROR.

Примеры

--	--	--

№	INPUT.TXT	OUTPUT.TXT
1	II/IV	I/II
2	XXIV/VIII	III
3	12/16	ERROR

Задача №200
Марсианские факториалы

(Время: 1 сек. Память: 16 Мб Сложность: 77%)

В 3141 году очередная экспедиция на Марс обнаружила в одной из пещер таинственные знаки. Они однозначно доказывали существование на Марсе разумных существ. Однако смысл этих таинственных знаков долгое время оставался неизвестным. Недавно один из ученых, профессор Очень-Умный, заметил один интересный факт: всего в надписях, составленных из этих знаков, встречается ровно K различных символов. Более того, все надписи заканчиваются на длинную последовательность одних и тех же символов.

Вывод, который сделал из своих наблюдений профессор, потряс всех ученых Земли. Он предположил, что эти надписи являются записями факториалов различных натуральных чисел в системе счисления с основанием K . А символы в конце – это конечно же нули – ведь, как известно, факториалы больших чисел заканчиваются большим количеством нулей. Например, в нашей десятичной системе счисления факториалы заканчиваются на нули начиная с $5! = 1 \cdot 2 \cdot 3 \cdot 4 \cdot 5 = 120$. А у числа $100!$ в конце следует 24 нуля в десятичной системе счисления и 48 нулей в системе счисления с основанием 6 – так что у предположения профессора есть разумные основания!

Теперь ученым срочно нужна программа, которая по заданным числам N и K найдет количество нулей в конце записи в системе счисления с основанием K числа $N! = 1 \cdot 2 \cdot 3 \cdot \dots \cdot (N-1) \cdot N$, чтобы они могли проверить свою гипотезу. Вам придется написать им такую программу!

Входные данные

Входной файл INPUT.TXT содержит числа N и K , разделенные пробелом. ($1 \leq N \leq 10^9$, $2 \leq K \leq 1000$).

Выходные данные

Выведите в выходной файл OUTPUT.TXT число X - количество нулей в конце записи числа $N!$ в системе счисления с основанием K .

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	5 10	1
2	100 10	24
3	100 6	48
4	3 10	0

Задача №201
Пакетная обработка процессов

(Время: 1 сек. Память: 16 Мб Сложность: 48%)

Для ускорения прохождения «коротких» заданий на ЭВМ выбран пакетный режим работы с

квантованием времени процессора. Это значит, что всем заданиям пакета по очереди представляется процессор на одинаковое время 10 с (круговой циклический алгоритм разделения времени). Если в течение этого времени заканчивается выполнение задания, оно покидает систему и освобождает процессор. Если же очередного кванта времени не хватает для завершения задания, оно помещается в конец очереди — пакета. Последнее задание пакета выполняется без прерываний. Пакет считается готовым к вводу в ЭВМ, если в нем содержится К заданий. Новый пакет вводится в ЭВМ после окончания обработки предыдущего. Задания поступают в систему с интервалом времени 60 ± 30 с и характеризуются временем работы процессора 50 ± 45 с.

Требуется смоделировать процесс обработки N заданий и определить время начала и окончания каждого процесса.

Входные данные

На первой строке входного файла INPUT.TXT находятся числа N и K - число процессов и количество процессов в пакете ($1 \leq N \leq 1000$, $1 \leq K \leq 100$). Гарантируется, что N делится на K. Далее следуют N строк с информацией о времени формирования и необходимое время на выполнение для каждого процесса. Все процессы стартуют в один день и следуют в порядке возрастания времени ввода их в систему.

Выходные данные

Выведите в выходной файл OUTPUT.TXT для каждого процесса в отдельной строке время его старта и время окончания через пробел в формате ЧЧ:ММ:СС.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	6 2	00:02:14 00:04:17
	00:01:02 63	00:02:24 00:04:33
	00:02:14 76	00:04:33 00:04:57
	00:03:16 14	00:04:43 00:05:06
	00:04:02 19	00:05:45 00:06:12
	00:04:36 17	00:05:55 00:07:17
	00:05:45 75	

Задача №202 Поиск подстроки

(Время: 1 сек. Память: 16 Мб Сложность: 38%)

Найти все вхождения строки T в строке S.

Входные данные

В первой строке входного файла INPUT.TXT записана строка S, во второй строке записана строка T. Обе строки состоят только из латинских букв. Длины строк больше 0 и меньше 50 000.

Выходные данные

В выходной файл OUTPUT.TXT нужно вывести все вхождения строки T в строку S в порядке возрастания. Нумерация позиций строк начинается с нуля.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	ababbababa aba	0 5 7

Задача №203
Сдвиг текста

(Время: 1 сек. Память: 16 Мб Сложность: 31%)

Мальчик Кирилл написал однажды на листе бумаги строчку, состоящую из больших и маленьких латинских букв, а после этого ушел играть в футбол. Когда он вернулся, то обнаружил, что его друг Дима написал под его строкой еще одну строчку такой же длины. Дима утверждает, что свою строчку он получил циклическим сдвигом строки Кирилла направо на несколько шагов(циклический сдвиг строки abcde на 2 позиции направо даст строку deabc). Однако Дима известен тем, что может случайно ошибиться в большом количестве вычислений, поэтому Кирилл в растерянности - верить ли Диме? Помогите ему!

По данным строкам выведите минимально возможный размер сдвига или -1, если Дима ошибся.

Входные данные

Первые две строки входного файла INPUT.TXT содержат строки Кирилла и Димы соответственно. Строки состоят только из латинских символов. Длины строк одинаковы, не превышают 10000 и не равны 0.

Выходные данные

В выходной файл OUTPUT.TXT выведите единственное число - ответ на поставленную задачу.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	abcde deabc	2

Задача №204
Циклическая строка

(Время: 1 сек. Память: 16 Мб Сложность: 53%)

Строка S была записана много раз подряд, после чего из получившейся строки взяли подстроку и дали Вам. Ваша задача определить минимально возможную длину исходной строки S.

Входные данные

В единственной строке входного файла INPUT.TXT записана строка, которая содержит только латинские буквы, длина строки не превышает 50000 символов.

Выходные данные

В выходной файл OUTPUT.TXT нужно вывести одно число - ответ на задачу.

Пример

--	--	--

№	INPUT.TXT	OUTPUT.TXT
1	abababa	2

Задача №205

Таймер

(Время: 1 сек. Память: 16 Мб Сложность: 31%)

Таймер - это часы, которые умеют подавать звуковой сигнал по прошествии некоторого периода времени. Напишите программу, которая определяет, когда должен быть подан звуковой сигнал.

Входные данные

В первой строке входного файла INPUT.TXT записано текущее время в формате ЧЧ:ММ:СС (с ведущими нулями). При этом оно удовлетворяет ограничениям: ЧЧ - от 00 до 23, ММ и СС - от 00 до 60.

Во второй строке записан интервал времени, который должен быть измерен. Интервал записывается в формате Ч:М:С (где Ч, М и С - от 0 до 10⁹, без ведущих нулей). Дополнительно если Ч=0 (или Ч=0 и М=0), то они могут быть опущены. Например, 100:60 на самом деле означает 100 минут 60 секунд, что то же самое, что 101:0 или 1:41:0. А 42 обозначает 42 секунды. 100:100:100 - 100 часов, 100 минут, 100 секунд, что то же самое, что 101:41:40.

Выходные данные

В выходной файл OUTPUT.TXT выведите в формате ЧЧ:ММ:СС время, во сколько прозвучит звуковой сигнал. При этом если сигнал прозвучит не в текущие сутки, то дальше должна следовать запись +<кол во> days. Например, если сигнал прозвучит на следующий день – то +1 days.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	01:01:01 48:0:0	01:01:01+2 days
2	01:01:01 58:119	02:01:00
3	23:59:59 1	00:00:00+1 days

Задача №206

Домой на электричках

(Время: 1 сек. Память: 16 Мб Сложность: 40%)

Одна из команд-участниц олимпиады решила вернуться домой на электричках. При этом ребята хотят попасть домой как можно раньше. К сожалению, не все электрички идут от города, где проводится олимпиада, до станции, на которой живут ребята. И, что еще более обидно, не все электрички, которые идут мимо их станции, останавливаются на ней (равно как вообще, электрички останавливаются далеко не на всех станциях, мимо которых они идут).

Все станции на линии пронумерованы числами от 1 до N. При этом станция номер 1 находится в городе, где проводится олимпиада, и в момент времени 0 ребята приходят на станцию. Станция, на которую нужно попасть ребятам, имеет номер E.

Напишите программу, которая по данному расписанию движения электричек вычисляет минимальное время, когда ребята могут оказаться дома.

Входные данные

Во входном файле INPUT.TXT записаны сначала числа N ($2 \leq N \leq 100$) и E ($2 \leq E \leq N$). Затем записано число M ($0 \leq M \leq 100$), обозначающее число рейсов электричек. Далее идет описание M рейсов электричек. Описание каждого рейса электрички начинается с числа K_i ($2 \leq K_i \leq N$) — количества станций, на которых она останавливается, а далее следует K_i пар чисел, первое число каждой пары задает номер станции, второе — время, когда электричка останавливается на этой станции (время выражается целым числом из диапазона от 0 до 10^9). Станции внутри одного рейса упорядочены в порядке возрастания времени. В течение одного рейса электричка все время движется в одном направлении — либо от города, либо к городу.

Выходные данные

В выходной файл OUTPUT.TXT выведите одно число — минимальное время, когда ребята смогут оказаться на своей станции. Если существующими рейсами электричек они добраться не смогут, выведите -1 .

Пример

№	INPUT.TXT	OUTPUT.TXT
1	5 3 4 2 1 5 2 10 2 2 10 4 15 4 5 0 4 17 3 20 2 35 3 1 2 3 40 4 45	20

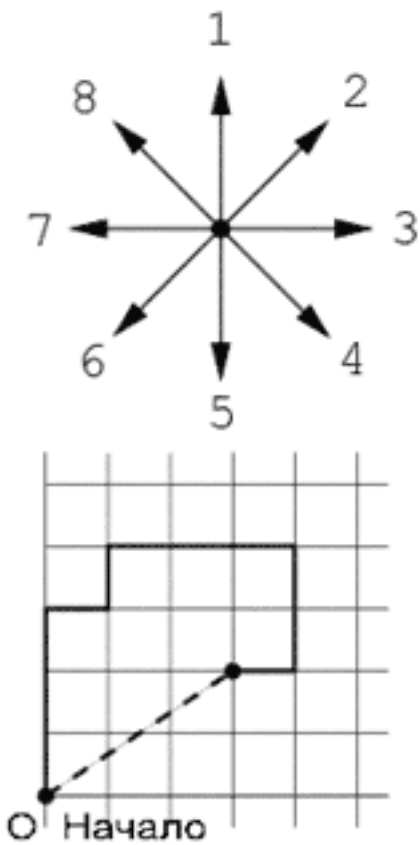
Задача №207 Клад

(Время: 1 сек. Память: 16 Мб Сложность: 28%)

Найти закопанный пиратами клад просто: всё, что для этого нужно — это карта. Как известно, пираты обычно рисуют карты от руки и описывают алгоритм действий. Большая часть таких действий просто сводится к прохождению какого-то количества шагов в одном из восьми направлений (1 — север, 2 — северо-восток, 3 — восток, 4 — юго-восток, 5 — юг, 6 — юго-запад, 7 — запад, 8 — северо-запад) (см. рис). Длина шага в любом направлении равна 1.

Путешествие по такому пути обычно является прекрасным способом посмотреть окрестности, однако в наше время постоянной спешки ни у кого нет времени на это. Поэтому кладоискатели хотят идти напрямую в точку, где зарыт клад. Например, вместо того, чтобы проходить три шага на север, один шаг на восток, один шаг на север, три шага на восток, два шага на юг и один шаг на запад, можно пройти напрямую, использовав около 3.6 шага (см. рисунок).

Вам необходимо узнать точку, в которой находится клад согласно указаниям пиратов.



Входные данные

Первая строка входного файла INPUT.TXT содержит число N – число указаний ($1 \leq N \leq 40$). Последующие N строк содержат сами указания – номер направления (целое число от 1 до 8) и количество шагов (целое число от 1 до 1000). Числа разделены пробелами.

Выходные данные

В выходной файл OUTPUT.TXT выведите координаты X и Y точки (два вещественных числа, разделённые пробелом), где зарыт клад, считая, что ось OX направлена на восток, а ось OY – на север. В начале кладоискатель должен стоять в начале координат. Координаты необходимо вывести с точностью 10^{-3} .

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	6 1 3 3 1 1 1 3 3 5 2 7 1	3.000 2.000
2	1 8 10	-7.071 7.071

Задача №208 Забавная игра

(Время: 1 сек. Память: 16 Мб Сложность: 30%)

Легендарный учитель математики Юрий Петрович придумал забавную игру с числами. А именно, взяв произвольное целое число, он переводит его в двоичную систему счисления, получая некоторую последовательность из нулей и единиц, начинающуюся с единицы. (Например, десятичное число $19_{10} = 1 \cdot 2^4 + 0 \cdot 2^3 + 0 \cdot 2^2 + 1 \cdot 2^1 + 1 \cdot 2^0$ в двоичной системе запишется как 10011_2 .) Затем учитель начинает сдвигать цифры полученного двоичного числа по циклу (так, что последняя цифра становится первой, а все остальные сдвигаются на одну позицию вправо), выписывая образующиеся при этом последовательности из нулей и единиц в столбик — он подметил, что независимо от выбора исходного числа получающиеся последовательности начинают с некоторого момента повторяться. И, наконец, Юрий Петрович отыскивает максимальное из выписанных чисел и переводит его обратно в десятичную систему счисления, считая это число результатом проделанных манипуляций. Так, для числа 19 список последовательностей будет таким:

10011
11001
11100
01110
00111
10011
...

и результатом игры, следовательно, окажется число $1 \cdot 2^4 + 1 \cdot 2^3 + 1 \cdot 2^2 + 0 \cdot 2^1 + 0 \cdot 2^0 = 28$.

Поскольку придуманная игра с числами все больше занимает воображение учителя, отвлекая

тем самым его от работы с ну очень одаренными школьниками, Вас просят написать программу, которая бы помогла Юрию Петровичу получать результат игры без утомительных ручных вычислений.

Входные данные

Входной файл INPUT.TXT содержит одно целое число N ($0 \leq N \leq 32767$).

Выходные данные

Ваша программа должна вывести в выходной файл OUTPUT.TXT одно целое число, равное результату игры.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	19	28
2	1212	1938

Задача №209 Целые точки

(Время: 1 сек. Память: 16 Мб Сложность: 64%)

Многоугольник (не обязательно выпуклый) на плоскости задан координатами своих вершин. Требуется подсчитать количество точек с целочисленными координатами, лежащих внутри него (но не на его границе).

Входные данные

В первой строке входного файла INPUT.TXT содержится N ($3 \leq N \leq 10^3$) – число вершин многоугольника. В последующих N строках идут координаты (X_i, Y_i) вершин многоугольника в порядке обхода по часовой стрелке. X_i и Y_i - целые числа, по модулю не превосходящие 10^6 .

Выходные данные

Ваша программа должна вывести в выходной файл OUTPUT.TXT одно целое число - ответ на задачу.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	4 -1 -1 -1 1 1 1 1 -1	1
2	3 0 0 0 2 2 0	0

Задача №210

Степень

(Время: 1 сек. Память: 16 Мб Сложность: 59%)

Для того чтобы проверить, как её ученики умеют считать, Мария Ивановна каждый год задаёт им на дом одну и ту же задачу – для заданного натурального A найти минимальное натуральное N такое, что N в степени N (N , умноженное на себя N раз) делится на A . От года к году меняется только число A .

Вы решили помочь будущим поколениям. Для этого вам необходимо написать программу, решающую эту задачу.

Входные данные

Во входном файле INPUT.TXT содержится единственное число A ($1 \leq A \leq 10^9$ – на всякий случай, вдруг Мария Ивановна задаст большое число, чтобы «завалить» кого-нибудь...).

Выходные данные

В выходной файл OUTPUT.TXT необходимо вывести единственное число N .

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	8	4
2	13	13

Задача №211

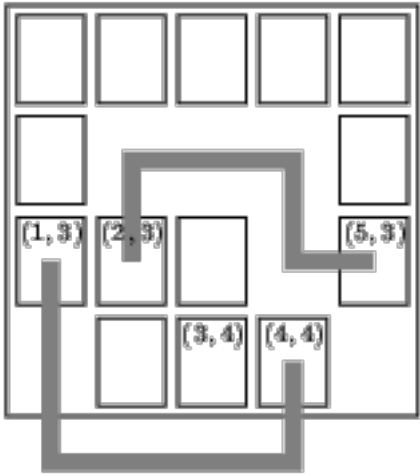
Игра с фишками

(Время: 1 сек. Память: 16 Мб Сложность: 52%)

Вы являетесь одним из разработчиков новой компьютерной игры. Игра происходит на прямоугольной доске, состоящей из $W \times H$ клеток. Каждая клетка может либо содержать, либо не содержать фишку. Важной частью игры является проверка того, соединены ли две фишки путем, удовлетворяющим следующим свойствам:

Путь должен состоять из отрезков вертикальных и горизонтальных прямых.

Путь не должен пересекать других фишек. При этом часть пути может оказаться вне доски. Например:



Фишки с координатами (1,3) и (4,4) могут быть соединены. Фишки с координатами (2,3) и (5,3) тоже могут быть соединены. А вот фишки с координатами (2,3) и (3,4) соединить нельзя – любой соединяющий их путь пересекает другие фишки.

Вам необходимо написать программу, проверяющую, можно ли соединить две фишки путем, обладающим вышеуказанными свойствами, и, в случае положительного ответа, определяющую минимальную длину такого пути (считается, что путь имеет изломы, начало и конец только в центрах клеток (или «мнимых клеток», расположенных вне доски), а отрезок, соединяющий центры двух соседних клеток, имеет длину 1).

Входные данные

Первая строка входного файла INPUT.TXT содержит два натуральных числа: W – ширина доски, H – высота доски ($1 \leq W, H \leq 75$). Следующие H строк содержат описание доски: каждая строка состоит ровно из W символов: символ «X» (заглавная английская буква «экс») обозначает фишку, символ «.» (точка) обозначает пустое место. Все остальные строки содержат описания запросов: каждый запрос состоит из четырёх натуральных чисел, разделённых пробелами – X1, Y1, X2, Y2, причём $1 \leq X1, X2 \leq W$, $1 \leq Y1, Y2 \leq H$. Здесь (X1, Y1) и (X2, Y2) – координаты фишек, которые требуется соединить (левая верхняя клетка имеет координаты (1,1)). Гарантируется, что эти координаты не будут совпадать (кроме последнего запроса; см. далее). Последняя строка содержит запрос, состоящий из четырёх чисел 0; этот запрос обрабатывать не надо. Количество запросов не превосходит 20.

Выходные данные

Для каждого запроса в выходной файл OUTPUT.TXT необходимо вывести одно целое число на отдельной строке – длину кратчайшего пути, или 0, если такого пути не существует.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	5 4	5
	XXXXX	6
	X...X	0
	XXX.X	
	.XXX.	
	2 3 5 3	
	1 3 4 4	
	2 3 3 4	
	0 0 0 0	

Задача №212 Деревни

(Время: 1 сек. Память: 16 Мб Сложность: 79%)

В тридесятом государстве есть N деревень. Некоторые пары деревень соединены дорогами. В целях экономии, «лишних» дорог нет, т.е. из любой деревни в любую можно добраться по дорогам единственным образом.

Новейшие исследования показали, что тридесятое государство находится в сейсмически опасной зоне. Поэтому глава государства захотел узнать, какой именно ущерб может принести его державе землетрясение. А именно, он хочет узнать, какое минимальное число дорог должно быть разрушено, чтобы образовалась изолированная от остальных группа ровно из P деревень такая, что из любой деревни из этой группы до любой другой деревни из этой группы по-прежнему можно будет добраться по неразрушенным дорогам (группа изолирована от остальных, если никакая неразрушенная дорога не соединяет деревню из этой группы с деревней не из этой группы).

Вы должны написать программу, помогающую ему в этом.

Входные данные

Первая строка входного файла INPUT.TXT содержит два числа: N и P ($1 \leq P \leq N \leq 150$). Все остальные строки содержат описания дорог, по одному на строке: описание дороги состоит из двух номеров деревень (от 1 до N), которые эта дорога соединяет. Все числа во входном файле разделены пробелами и/или переводами строки.

Выходные данные

В выходной файл OUTPUT.TXT выведите единственное число – искомое количество дорог.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	3 2 1 2 3 2	1
2	11 6 1 2 1 3 1 4 1 5 2 6 2 7 2 8 4 9 4 10 4 11	2

Задача №213 Подсчет баллов

(Время: 1 сек. Память: 16 Мб Сложность: 42%)

Решение каждой задачи заочного тура проверяется на наборе заранее заготовленных тестов. По результатам работы программы на каждом тесте участнику либо начисляются баллы за этот тест (когда программа выдала правильный ответ), либо не начисляются (когда во время работы программы произошли ошибки или выданный ответ не верен). Тесты могут иметь разную стоимость.

Дополнительные баллы начисляются участнику, если его программа прошла все тесты.

Участник может исправлять свое решение, и посылать его на проверку повторно (при этом решение проверяется на том же наборе тестов). При этом за каждую попытку из количества набранных по задаче баллов вычитается штраф, который равен 0 при 1-й попытке, а при каждой следующей возрастает на 2 (то есть 2 при второй, 4 — при третьей, 6 — при четвертой и т.д.).

Из баллов, полученных участником за каждую из попыток (с учетом начисленных штрафов), выбирается максимальный результат, который и засчитывается как результат данного участника по этой задаче. Это нужно, в частности, для того, чтобы последующие попытки не ухудшали уже полученный участником результат по задаче.

Например, если участник делает первую попытку и набирает 10 баллов, его результат по задаче равен 10 баллов. Пусть на второй попытке участник посылает решение, которое набирает 8 баллов. С учетом штрафа за эту попытку участник имеет 6 баллов, однако результат команды по задаче остается равным 10. Пусть с 3-й попытки решение набрало 20 баллов, тогда (с учетом штрафа) результат участника по задаче становится равен 16 баллам. Наконец, пусть с 4-й попытки решение проходит все тесты, тогда участник получает сумму баллов за все тесты, плюс призовые баллы за прохождение всех тестов, минус 6 баллов штрафа (если, конечно, эта величина не меньше 16 баллов, которые уже были у данного участника).

Напишите программу, которая определяет результат данного участника по этой задаче.

Входные данные

Во входном файле INPUT.TXT записано сначала число N — количество тестов, на которых проверяются решения данной задачи ($1 \leq N \leq 100$). Далее идет N натуральных чисел, не превышающих 100, — баллы, которые начисляются за прохождение каждого из тестов. Далее идет целое число из диапазона от 0 до 100 — количество баллов, которое дополнительно начисляется за прохождение всех тестов.

Далее идет натуральное число M — количество попыток сдачи задачи ($1 \leq M \leq 100$). После чего идет M наборов по N чисел в каждом, задающих результаты проверки каждой из M попыток сдачи задачи на тестах. 0 обозначает, что соответствующий тест не пройден, 1 — пройден.

Выходные данные

В выходной файл OUTPUT.TXT выведите M чисел. i -ое число должно соответствовать результату участника после совершения им первых i попыток.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	4	0
	1 2 3 4	13
	5	13
	3	
	0 0 0 0	
	1 1 1 1	
	0 1 0 1	

Задача №214 Великая сеча

(Время: 1 сек. Память: 16 Мб Сложность: 58%)

Алеша Попович и Добрыня Никитич сражаются со стаей двух- и трехголовых драконов. Они по очереди взмахивают мечами, и одним махом могут отрубить любое (по своему желанию) число голов, но только у одного дракона. Отрубивший последнюю голову у последнего дракона получает в жены прекрасную принцессу.

Кто из богатырей (начинающий или второй) может получить в жены принцессу независимо от действий другого?

Входные данные

Во входном файле INPUT.TXT записано два числа N и M — количество двух- и трехголовых драконов соответственно (оба числа целые из диапазона от 0 до 100, $N+M>0$).

Выходные данные

В выходной файл OUTPUT.TXT выведите число 1 или 2 определяющее, кто из богатырей имеет все шансы получить в жены принцессу (1 — тот, кто начинает, 2 — второй).

Примеры

№	INPUT.TXT	OUTPUT.TXT

1	2 2	2
2	1 2	1

Задача №215
Водостоки

(Время: 1 сек. Память: 16 Мб Сложность: 55%)

Карту местности условно разбили на квадраты, и посчитали среднюю высоту над уровнем моря для каждого квадрата.

Когда идет дождь, вода равномерно выпадает на все квадраты. Если один из четырех соседних с данным квадратом квадратов имеет меньшую высоту над уровнем моря, то вода с текущего квадрата стекает туда (и, если есть возможность, то дальше), если же все соседние квадраты имеют большую высоту, то вода скапливается в этом квадрате.

Разрешается в некоторых квадратах построить водостоки. Когда на каком-то квадрате строят водосток, то вся вода, которая раньше скапливалась в этом квадрате, будет утекать в водосток.

Если есть группа квадратов, имеющих одинаковую высоту и образующих связную область, то если хотя бы рядом с одним из этих квадратов есть квадрат, имеющий меньшую высоту, то вся вода утекает туда, если же такого квадрата нет, то вода стоит во всех этих квадратах. При этом достаточно построить водосток в любом из этих квадратов, и вся вода с них будет утекать в этот водосток.

Требуется определить, какое минимальное количество водостоков нужно построить, чтобы после дождя вся вода утекала в водостоки.

Входные данные

Во входном файле INPUT.TXT записаны сначала числа N и M, задающие размеры карты — натуральные числа, не превышающие 100. Далее, идет N строк, по M чисел в каждой, задающих высоту квадратов карты над уровнем моря. Высота задается натуральным числом, не превышающим 10000. Считается, что квадраты, расположенные за пределами карты, имеют высоту 10001 (то есть вода никогда не утекает за пределы карты).

Выходные данные

В выходной файл OUTPUT.TXT выведите минимальное количество водостоков, которое необходимо построить.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	4 5 1 2 3 1 10 1 4 3 10 10 1 5 5 5 5 6 6 6 6 6	2

Задача №216
Коллекционирование этикеток

(Время: 1 сек. Память: 16 Мб Сложность: 41%)

Вася коллекционирует спичечные этикетки. Для этого у него есть N альбомов вместимостью

K_1, K_2, \dots, K_N этикеток. Вася хочет, чтобы в случае утери одного любого альбома каждая этикетка осталась у него хотя бы в одном экземпляре. Для этого он покупает каждую этикетку в двух экземплярах, и наклеивает их в два разных альбома. Какое максимальное количество различных этикеток при этом может оказаться в его коллекции?

Входные данные

Входной файл INPUT.TXT содержит сначала число N — количество альбомов, а затем N чисел K_1, K_2, \dots, K_N , задающих вместимости альбомов. N — натуральное число из диапазона от 2 до 1000. Вместимость каждого альбома задается натуральным числом, суммарная вместимость всех альбомов не превышает 100 000 этикеток.

Выходные данные

В выходной файл OUTPUT.TXT выведите число E — максимальное количество различных этикеток, которое может собрать Вася с соблюдением выдвинутого условия.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	4 1 2 1 1	2

Задача №217 Еловая аллея

(Время: 1 сек. Память: 16 Мб Сложность: 47%)

Мэр города Урюпинска решил посадить на главной аллее города, которая проходит с запада на восток, голубые ели. Причем сажать ели можно не во всех местах, а только на специально оставленных при асфальтировании аллеи клумбах.

Как оказалось, голубые ели бывают M различных сортов. Для ели каждого сорта известна максимальная длина ее тени в течение дня в западном и в восточном направлении (W_i и E_i соответственно). При этом известно, что ели растут гораздо лучше, если в течение дня они не оказываются в тени других елей.

Координатная ось направлена вдоль аллеи с запада на восток.

По заданным координатам клумб вычислите максимальное число елей, которое можно посадить, соблюдая условие о том, что никакая ель не должна попадать в тень от другой ели.

Входные данные

Во входном файле INPUT.TXT записано сначала натуральное число M — количество сортов елей ($1 \leq M \leq 100$). Затем идет M пар чисел W_i, E_i , описывающих максимальную длину тени в западном и в восточном направлении в течение дня для каждого сорта ели (числа W_i, E_i — целые, из диапазона от 0 до 30000). Далее идет натуральное число N — количество клумб, в которых можно сажать ели ($1 \leq N \leq 100$). Далее идет N чисел, задающих координаты клумб (координаты — целые числа, по модулю не превышающие 30000). Клумбы перечислены с запада на восток (в порядке возрастания их координат).

Примечание

Если на клумбе с координатой X мы посадили ель, максимальная тень которой в восточном

направлении равна E, то все клумбы с координатами от X+1 до X+E−1 попадают в тень от этой ели, а клумба с координатами X+E — уже нет. Аналогично для тени в западном направлении.

Выходные данные

В выходной файл OUTPUT.TXT выведите число A — максимальное количество елей, которые удастся посадить.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	3 1000 3 1 200 128 256 3 1 2 4	2

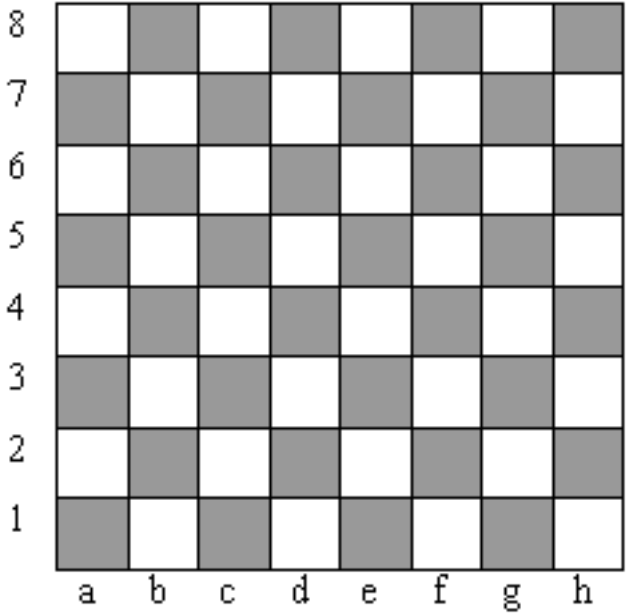
Задача №218 Шашки

(Время: 1 сек. Память: 16 Мб Сложность: 45%)

Петя и Вася играли в шашки по описанным ниже правилам. В какой-то момент забежавший в комнату кот перевернул доску, на которой играли Петя и Вася. К счастью, у них осталась запись сделанных ходов, используя которую можно восстановить расположение шашек к моменту, когда забежал кот.

Напишите программу, которая выведет положение шашек на доске после выполнения описанных ходов.

Игра происходит на стандартной доске (8x8), которая располагается так, что у игрока, играющего белыми, левая нижняя клетка является черной, и с нее начинается нумерация как строк, так и столбцов. Строки доски нумеруются числами от 1 до 8, столбцы — латинскими буквами от a до h.



В начале игры каждый из двух игроков имеет по 12 шашек своего цвета (белого или черного соответственно). Белые шашки располагаются на клетках a1, a3, b2, c1, c3, d2, e1, e3, f2, g1, g3, h2. Черные шашки располагаются на клетках a7, b6, b8, c7, d6, d8, e7, f6, f8, g7, h6, h8.

Игроки совершают ходы по очереди. Игрок, играющий белыми, ходит первым.

Шашки в процессе игры бывают двух видов: обычная шашка и дамка. В начале игры все шашки обычные. Белая шашка становится дамкой, если она оказывается в строке 8. Соответственно, черная шашка становится дамкой, если она оказывается в строке 1.

Шашка может совершать ходы двух типов:

1. *Простой ход* заключается в перемещении одной из шашек на одну клетку вперед по диагонали. Например, белая шашка с e3 может сходить на d4 или f4 (если соответствующая клетка свободна). А черная шашка с e3 может сходить на d2 или f2.

2. *Рубка* заключается в том, что шашка перепрыгивает через шашку (или дамку) противника, находящуюся в диагонально соседней с ней клетке при условии, что следующая клетка этой

диагонали свободна. Шашка противника, которую срубили, убирается с доски. Если сразу после окончания рубки та же самая шашка может продолжить рубку, она ее продолжает этим же ходом. Рубка возможна в любом из 4-х диагональных направлений. Если в процессе рубки шашка оказывается в 1-й строке (для черных) или в 8-й (для белых), она становится дамкой.

Дамка может совершать следующие ходы:

3. *Простой ход* заключается в перемещении дамки по диагонали на любое число клеток (при этом все клетки, через которые происходит перемещение, должны быть свободны).

4. *Рубка* заключается в том, что шашка перепрыгивает через шашку (или дамку) противника, находящуюся на той же диагонали, что и рубящая дамка. Это можно делать при условии, что все клетки между рубящей дамкой и шашкой, которую рубят, а также клетка, следующая за шашкой, которую рубят, свободны. После рубки дамка может встать на любую клетку данной диагонали за клеткой, на которой стояла срубленная шашка (при условии, что все клетки на ее пути свободны). Если из своего нового положения дамка может совершить рубку, она должна ее совершить этим же ходом.

Входные данные

В первой строке входного файла INPUT.TXT записано одно число N — количество ходов, которое было сделано с начала партии. Это количество не превышает 1000.

В каждой из следующих N строк записаны описания ходов (нечетные ходы совершались белыми, четные — черными). Описание каждого хода занимает ровно одну строку и записывается в следующем виде.

Простой ход записывается в виде <начальная клетка>—<конечная клетка>. Например, ход с c3 на d4 записывается как c3-d4.

Рубка записывается в виде <начальная клетка>:<клетка после рубки>. Если рубка продолжается, то ставится еще одно двоеточие, и записывается клетка, где окажется шашка после следующей рубки и т.д. Например, e3:c5:e7 (шашка, изначально расположенная на e3, рубит шашку на d4 и оказывается на c5, после чего рубит шашку на d6 и оказывается на e7).

Рубка a1:h8 может быть осуществлена только дамкой (например, дамка с a1 рубит шашку, стоящую в c3, и заканчивает ход в h8). Рубка дамкой может быть и такой a1:d4:f6:h4 (дамка рубит шашку, стоящую на b2, затем продолжает рубку и рубит шашку на e5, и, наконец, рубит шашку на g5). При этом после каждой рубки указывается клетка, на которой останавливается дамка перед следующей рубкой.

Строки с описанием ходов не содержат пробельных символов.

Выходные данные

В выходной файл OUTPUT.TXT выведите изображение доски с расположенными на ней шашками. В первой строке выходного файла должна быть выведена 8-я строка доски, во второй — 7-я и т.д. В каждой строке должно быть ровно 8 символов, описывающих клетки столбцов с a по h.

Белая клетка должна быть изображена символом “.” (точка), пустая черная клетка — символом “-” (минус). Черная клетка, в которой стоит белая шашка — символом “w” (маленькая латинская буква w), а клетка с белой дамкой — символом “W” (заглавная латинская буква W). Аналогично клетка с черной шашкой должна быть изображена символом “b” (маленькая латинская буква b), а клетка с черной дамкой — символом “B” (большая латинская буква B).

Пример

№	INPUT.TXT	OUTPUT.TXT
---	-----------	------------

1	3	.b.b.b.b
	c3-d4	b.b.b.b.
	f6-e5	.b.b.w.b
	d4:f6	-.-.-.-.
		.-.-.-.-
		w.-.w.w.
		.w.w.w.w
		w.w.w.w.

Задача №219

Симпатичные таблицы

(Время: 1 сек. Память: 16 Мб Сложность: 75%)

Рассмотрим таблицу размера $M \times N$, в клетках которой стоят целые неотрицательные числа. Скажем, что таблица является симпатичной, если для всех i сумма чисел ее i -ой строки не превышает R_i и для всех j сумма чисел ее j -го столбца не превышает C_j .

Вам задана таблица Z размера $M \times N$ (N строк и M столбцов), в некоторых клетках которой уже стоят целые неотрицательные числа. Найдите симпатичную таблицу с максимальной суммой элементов такую, что она совпадает с Z на тех клетках, в которых в Z стоят числа.

Входные данные

Первая строка входного файла INPUT.TXT содержит числа N и M ($1 \leq M, N \leq 20$). Следующая строка содержит N целых неотрицательных чисел - R_1, R_2, \dots, R_N . Следующая строка содержит M целых неотрицательных чисел C_1, C_2, \dots, C_M . Все числа не превышают 10^6 . Следующие N строк содержат по M целых чисел, которые задают Z . Если на некотором месте в таблице отсутствует число, то на этом месте во входном файле стоит число -1.

Выходные данные

Выведите в выходной файл OUTPUT.TXT выведите целое число - сумму элементов найденной таблицы. Если решение не существует, то выведите единственное число -1.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	<div> <div>22</div> <div>22</div> <div>12</div> <div>1-1</div> <div>-1-1</div> </div>	3

Задача №220

Мышка с колесиком

(Время: 1 сек. Память: 16 Мб Сложность: 54%)

Пользователь просматривает таблицу в Internet Explorer и пользуется для прокрутки изображения колесиком на мышке. При этом все изображение сдвигается вверх или вниз на T пикселей. Пользователю очень не нравится, когда курсор мыши оказывается на горизонтальных линиях, разделяющих строки таблицы. Поэтому он хочет выбрать такое положение для курсора мыши на экране, чтобы в процессе прокрутки до конца таблицы курсор как можно меньшее число

раз пересекался с линиями таблицы.

При этом если в каком-то положении курсор оказывается на двух линиях таблицы, то это считается за два пересечения курсора с линиями таблицы. Если какую-то линию курсор мыши пересекает в двух положениях (то есть, например, высота курсора 10 пикселей, а при прокрутке таблица сдвигается на 7 пикселей, тогда курсор мыши может оказываться на одной линии в двух состояниях прокрутки), то это также считается за два пересечения.

Экран монитора имеет разрешение по вертикали U пикселей. Координаты введены так, что самые верхние точки экрана имеют координату 0, а нижние — координату $U-1$. Курсор мыши имеет высоту H пикселей. Расположением курсора считается самая верхняя точка курсора. Таким образом, если мы говорим, что он расположен, например, в точке с координатами 0 на экране, то его изображение расположено в точках с координатами от 0 до $H-1$. Курсор мыши всегда целиком помещается на экране, то есть допустимыми координатами для его расположения являются координаты от 0 до $U-H$. Таблица, которую просматривает пользователь, имеет высоту L пикселей и состоит из $N-1$ строки, и, следовательно, в ней N горизонтальных линий, которые имеют координаты X_1, X_2, \dots, X_N . При этом $0 = X_1 < X_2 < X_3 < \dots < X_N = L-1$.

В начальный момент времени таблица расположена так, что линия, имеющая координату 0 в таблице отображается в 0-й строке пикселей монитора. Далее при прокрутке таблица каждый раз сдвигается на T пикселей (то есть в 0-й строке монитора оказывается строка пикселей, имеющая в таблице координату T , координату $2T$ и т.д.). Так происходит до тех пор, пока на экране не окажется нижняя линия таблицы (которая имеет координату X_N). После этого дальнейшая прокрутка не происходит (если изначально $X_N < U$, то прокрутка вообще не происходит).

Входные данные

Во входном файле INPUT.TXT задано сначала разрешение монитора по вертикали U , затем высота курсора мыши H , затем шаг прокрутки T . Далее задана высота таблицы L . Далее задано количество разделительных линий в таблице N , и координаты X_1, X_2, \dots, X_N , где расположены эти линии относительно начала таблицы. Ограничения: $10 \leq U \leq 512, 1 \leq H \leq U, 1 \leq T \leq U, 2 \leq N \leq 200000, 0 = X_1 < X_2 < X_3 < \dots < X_N = L-1 \leq 10^9$.

Выходные данные

В выходной файл OUTPUT.TXT нужно вывести наименьшее возможное количество пересечений курсора мыши с линиями таблицы.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	10 3 10 10 4 0 1 6 9	0
2	10 6 2 21 11 0 2 4 6 8 10 12 14 16 18 20	21

Задача №221 Левый лабиринт

(Время: 1 сек. Память: 16 Мб Сложность: 49%)

В спортзале размером $N \times M$ метров построили современный аттракцион под названием "Левый

лабиринт". Для этого на полу спортзала с интервалом в 1 метр начертили линии, параллельные стенам спортзала. Таким образом, спортзал разбили на NхМ клеток. Дальше некоторые из этих клеток покрасили в черный цвет. Аттракцион заключается в том, что участника ставят в некоторой клетке спортзала и просят как можно быстрее добежать до некоторой другой клетки. При этом накладываются следующие условия:

- Участнику запрещено ходить по черным клеткам.
- Придя в какую-то клетку, участник может пойти либо прямо, либо налево, либо направо (если в соответствующем направлении клетка не покрашена в черный цвет): ходить назад, а также ходить по диагонали запрещается.
- За все время пути участнику разрешается повернуть направо (то есть пойти из текущей клетки направо относительно того, откуда он пришел в данную клетку) не более К раз.
- В начальной клетке участник может встать лицом в ту сторону, в какую ему захочется. С какой стороны участник прибежит в конечную клетку также не важно.

Известно, что на то, чтобы перебежать из клетки в соседнюю, участник тратит ровно 1 секунду. Требуется вычислить минимальное время, за которое участник сможет достичь конечной клетки.

Входные данные

Во входном файле INPUT.TXT сначала записано число К — количество разрешенных поворотов направо (целое число из диапазона от 0 до 5), затем записаны числа N и M, задающие размеры спортзала — натуральные числа, не превышающие 20. Далее записано N строк по M чисел в каждой. Число 0 обозначает неокрашенную клетку, число 1 — покрашенную, число 2 — клетку, откуда стартует участник и число 3 — клетку, куда нужно добежать (клетки, помеченные 2 и 3 являются неокрашенными). В лабиринте всегда есть ровно одна начальная клетка и ровно одна клетка, в которую нужно попасть.

Выходные данные

В выходной файл OUTPUT.TXT выведите минимальное время, за которое можно добраться в конечную клетку. Если попасть в конечную клетку нельзя, выведите -1.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	1 5 5 2 0 0 1 1 0 1 0 0 1 1 1 0 0 1 0 0 0 0 1 3 1 0 0 1	12
2	0 1 3 2 1 3	-1

Задача №222 Дремучий лес

(Время: 1 сек. Память: 16 Мб Сложность: 59%)

Будем говорить, что для наблюдателя лес является дремучим, если из своего текущего положения наблюдатель видит только деревья. Вам дана карта леса и координаты точки, в которой находится наблюдатель. Требуется определить, кажется ли лес дремучим данному наблюдателю.

На карте леса все деревья изображаются кругами. При этом в лесу бывают сросшиеся деревья (изображения таких деревьев на карте пересекаются), также одно дерево может находиться внутри другого. Точка, в которой стоит наблюдатель, не лежит внутри или на границе ни одного из деревьев.

Входные данные

Во входном файле INPUT.TXT содержится сначала целое число N — количество деревьев ($1 \leq N \leq 50000$). Затем идут два числа, задающих координаты наблюдателя. Затем идет N троек чисел, задающих деревья. Первые два числа задают координаты центра, а третье — радиус. Все координаты задаются точно, и выражаются вещественными числами не более чем с 2 знаками после десятичной точки, по модулю не превосходящими 100000.

Выходные данные

В выходной файл OUTPUT.TXT нужно вывести сообщение YES, если лес является дремучим, и NO иначе.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	4 0 0 2 2 2 -2 2 2 -2 -2 2 2 -2 2	YES
2	2 10 10 0 0 1 0.5 0 2	NO

Задача №223 Анаграммер

(Время: 1 сек. Память: 16 Мб Сложность: 75%)

Анаграммер — специальное устройство для получения из слова его анаграмм (то есть слов, записанных теми же буквами, но в другом порядке). Это устройство умеет выполнять 2 операции:

1. Взять очередную букву исходного слова и поместить ее в стек.
2. Взять букву из стека и добавить ее в конец выходного слова.

Стек — это хранилище данных, которое работает по принципу "первый пришел — последний ушел". Стек можно представить себе в виде пирамидки. Когда мы добавляем букву в стек, это соответствует тому, что на стержень пирамидки сверху мы надеваем кольцо, на котором написана соответствующая буква. Когда берем букву из стека, то это соответствует тому, что мы снимаем со стержня верхнее кольцо, и смотрим, какая буква на нем написана.

Например, слово TROT в слово TORT может быть преобразовано анаграммером двумя различными последовательностями операций: 11112222 или 12112212.

Напишите программу, которая по двум заданным словам вычисляет количество различных последовательностей операций анаграммера, которые преобразуют первое из этих слов во второе.

Входные данные

Первая строка входного файла INPUT.TXT содержит исходное слово, а вторая — слово, которое необходимо получить. Слова состоят только из заглавных латинских букв и имеют длину не более 50 символов. Оба слова имеют одинаковую длину. В этих строках не содержится пробелов.

Выходные данные

В первой строке выходного файла OUTPUT.TXT должно содержаться количество последовательностей операций анаграммера, с помощью которых можно преобразовать первое слово во второе.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	TROT TORT	2
2	MADAM ADAMM	4
3	LONG GONG	0
4	AAAAAAAA AAAAAAAA	1430

Задача №224 Наибольшее произведение

(Время: 1 сек. Память: 16 Мб Сложность: 38%)

Дано N целых чисел. Требуется выбрать из них три таких числа, произведение которых максимально.

Входные данные

Во входном файле INPUT.TXT записано сначала число N — количество чисел в последовательности ($3 \leq N \leq 10^6$). Далее записана сама последовательность: N целых чисел, по модулю не превышающих 30000.

Выходные данные

В выходной файл OUTPUT.TXT выведите значение наибольшего произведения искомых трех чисел.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	9 3 5 1 7 9 0 9 -3 10	810
2	3 -5 -30000 -12	-1800000

Задача №225

Покупка билетов

(Время: 1 сек. Память: 16 Мб Сложность: 43%)

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).

Входные данные

Во входном файле INPUT.TXT 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.

Выходные данные

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

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	5 5 10 15 2 10 15 5 5 5 20 20 1 20 1 1	12
2	2 3 4 5 1 1 1	4

Задача №226

Перегоны

(Время: 1 сек. Память: 16 Мб Сложность: 50%)

На некоторой железнодорожной ветке расположено N станций, которые последовательно пронумерованы числами от 1 до N . Известны расстояния между некоторыми станциями. Требуется точно вычислить длины всех перегонов между соседними станциями или указать, что это сделать невозможно (то есть приведенная информация является противоречивой или ее недостаточно).

Входные данные

Во входном файле INPUT.TXT записаны сначала числа N — количество станций ($2 \leq N \leq 100$) и E — количество пар станций, расстояния между которыми заданы ($0 \leq E \leq 10000$). Далее, идет E троек чисел, первые два числа каждой тройки задают номера станций (это числа из диапазона от 1 до N), а третье — расстояние между этими станциями (все эти расстояния заданы точно и выражаются вещественными неотрицательными числами не более чем с 3-я знаками после десятичной точки).

Выходные данные

В случае, когда восстановить длины перегонов можно однозначно, в выходной файл OUTPUT.TXT выведите сначала «YES», а затем $N-1$ вещественное число. Первое из этих чисел должно соответствовать расстоянию от 1-й станции до 2-й, второе — от 2-й до 3-й, и так далее. Все числа должны быть выведены с точностью до 3-х знаков после десятичной точки (например, число 2.3 следует выводить как 2.300). Если приведенная информация о расстояниях между станциями является противоречивой или не позволяет однозначно точно восстановить длины перегонов, выведите в выходной файл «NO».

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	3 2 1 2 1.250 3 1 3	YES 1.250 1.750
2	4 4 1 2 1.250 3 1 1.255 2 4 0.010 1 1 0.000	YES 1.250 0.005 0.005
3	3 1 1 1 1	NO
4	3 3 1 2 1.250 1 3 1.300 2 3 1.000	NO

Задача №227 Сломанный калькулятор

(Время: 1 сек. Память: 16 Мб Сложность: 62%)

У калькулятора есть две ячейки памяти: содержимое первой из них всегда отображается на табло, вторая является буфером. В начальный момент времени на табло калькулятора отображается целое число X , а в буфере записано число 0. У калькулятора работают только две клавиши: «+» и «=». При нажатии на «+» число, которое в данный момент отображено на табло, копируется в буфер. При нажатии на «=» число из буфера прибавляется к числу, отображенному на табло, и результат отображается на табло, число в буфере при этом не меняется.

Требуется за наименьшее число нажатий клавиш на калькуляторе добиться того, чтобы на табло было отображено число Y .

Входные данные

Входной файл INPUT.TXT содержит два целых числа X и Y. Каждое из этих чисел по модулю не превышает 10^9 .

Выходные данные

В выходной файл OUTPUT.TXT выведите одно число — количество нажатий клавиш, которое потребуется для получения числа Y. Если из числа X получить число Y с помощью указанных операций невозможно, в выходной файл выведите одно число −1.

Примеры

№	INPUT.TXT	OUTPUT.TXT	Пояснение
1	1 1	0	
2	−2 −6	3	+=+
3	1 8	6	+=+=+=
4	2 5	−1	

Задача №228 Валютные махинации

(Время: 1 сек. Память: 16 Мб Сложность: 39%)

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.

Входные данные

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.

Выходные данные

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

Пример

--	--	--

№	INPUT.TXT	OUTPUT.TXT
1	4 1 10 10 5.53 5.53 1.25 6 5	4000.00

Задача №229 Двухтуровая олимпиада

(Время: 1 сек. Память: 16 Мб Сложность: 46%)

Как известно, личная олимпиада по информатике проходит в два тура. На каждом из туров участники получают какие-то баллы, при этом итоговый балл определяется как сумма полученных баллов. Известны баллы, которые каждый участник получил на каждом из туров. Жюри хочет фальсифицировать итоги олимпиады так, чтобы победил «нужный» участник.

При этом жюри может делать следующие «подтасовки» (можно делать несколько «подтасовок» применительно как к одному и тому же, так и к разным турам):

1. Прибавить к результатам всех участников по одному из туров одно и то же положительное число.
2. Умножить результаты участников по одному из туров на некоторый коэффициент, больший 1.

При этом должна сохраниться правдоподобность результатов, которая заключается в том, что никто из участников не должен получить больше 100 баллов за каждый из туров.

Определите список участников, которые в результате таких фальсификаций могут оказаться победителями олимпиады (то есть в сумме за два тура иметь не меньше баллов, чем каждый из остальных участников).

Входные данные

Во входном файле INPUT.TXT записано сначала число участников N ($1 \leq N \leq 1000$), затем N пар чисел — результаты каждого участника за 1-й и за 2-й туры (результат участника за тур — это вещественное число от 0 до 100) не более, чем с 3 знаками после десятичной точки.

Выходные данные

В выходной файл OUTPUT.TXT выведите сначала количество участников, которые смогут стать победителями олимпиады, а затем в возрастающем порядке их номера.

Пример

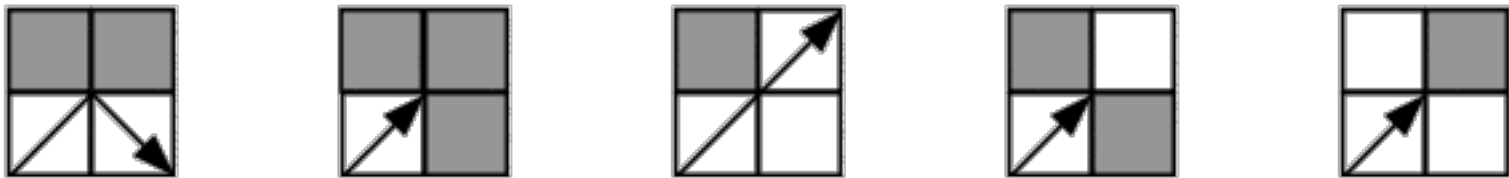
№	INPUT.TXT	OUTPUT.TXT
1	4 45 90 70 80 0 0 75 75	2 2 4

Задача №230 Луч света в темном царстве

(Время: 1 сек. Память: 16 Мб Сложность: 63%)

Темное царство представляет собой лабиринт N×M, некоторые клетки которого окружены зеркальными стенами, а остальные — пустые. Весь лабиринт также окружен зеркальной стеной. В одной из пустых клеток лабиринта поставили светофор, который испускает лучи в 4 направлениях: под 45 градусов относительно стен лабиринта. Требуется изобразить траекторию этих лучей.

Когда луч приходит в угол, через который проходят зеркальные стены, дальше он идет так, как показано на рисунках (серым цветом показаны клетки, которые окружены зеркальными стенами). Аналогичным образом луч ведет себя, когда приходит на границу лабиринта.



Входные данные

В первой строке входного файла INPUT.TXT записаны два натуральных числа N и M — число строк и столбцов в лабиринте (каждое из чисел не меньше 1 и не больше 100). В следующих N строках записано ровно по M символов в каждой — карта лабиринта. Символ * (звездочка) обозначает клетку, окруженную зеркальными стенками, . (точка) — пустую клетку, символ X (заглавная латинская буква X) — клетку, в которой расположен светофор (такая клетка ровно одна).

Выходные данные

В выходной файл OUTPUT.TXT выведите N строк по M символов в каждой — изображение лабиринта с траекториями лучей. Здесь, как и раньше, * (звездочка) должна обозначать клетки, окруженные зеркальными стенами, . (точка) — пустые клетки, через которые лучи света не проходят, / (слеш) — клетки, через которые луч света проходит из левого нижнего угла в правый верхний (или обратно — из правого верхнего в левый нижний), \ (обратный слеш) — клетки, через которые луч проходит из левого верхнего угла в правый нижний (или обратно), а символ X (заглавная латинская буква X) — клетки, через которые лучи проходят по обеим диагоналям.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	5 6 ..*...* *X...**	./*./\ /..X./ \./..X* *X.//* />\X/*.
2	3 3X. ...	\./ .X. /.\

Задача №231 Распаковка строки

(Время: 1 сек. Память: 16 Мб Сложность: 25%)

Будем рассматривать только строчки, состоящие из заглавных латинских букв. Например,

рассмотрим строку AAAABCCCCCDDDD. Длина этой строки равна 14. Поскольку строка состоит только из латинских букв, повторяющиеся символы могут быть удалены и заменены числами, определяющими количество повторений. Таким образом, данная строка может быть представлена как 4AB5C4D. Длина такой строки 7. Описанный метод мы назовем упаковкой строки.

Напишите программу, которая берет упакованную строчку и восстанавливает по ней исходную строку.

Входные данные

Входной файл INPUT.TXT содержит одну упакованную строку. В строке могут встречаться только конструкции вида nA, где n — количество повторений символа (целое число от 2 до 99), а A — заглавная латинская буква, либо конструкции вида A, то есть символ без числа, определяющего количество повторений. Максимальная длина строки не превышает 80.

Выходные данные

В выходной файл OUTPUT.TXT выведите восстановленную строку. При этом строка должна быть разбита на строчки длиной ровно по 40 символов (за исключением последней, которая может содержать меньше 40 символов).

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	3A4B7D	AAABBBBDDDDDDDD
2	22D7AC18FGD	DDDDDDDDDDDDDDDDDDDDDDDDDDDDDDAAAAAACFFFFFFF FFFFFFFGD
3	95AB	AA AA AAAAAAAAAAAAAAAAAB
4	40AB39A	AA BAA

Задача №232 Дремучий лес - 2

(Время: 1 сек. Память: 16 Мб Сложность: 77%)

Просека — это такая прямая линия, которая проходит через лес (то есть деревья есть как с одной стороны от этой линии, так и с другой), и при этом она не проходит ни через одно из деревьев леса, а также не касается деревьев. Будем говорить, что лес является дремучим, если в нем нет ни одной просеки.

На плане леса все деревья изображаются кругами. Никакие два круга не пересекаются и не касаются друг друга. Требуется по этому плану определить, является ли лес дремучим.

Входные данные

Во входном файле INPUT.TXT содержится сначала целое число N — количество деревьев (1<=N<=200). Затем идет N троек чисел, задающих деревья. Первые два числа задают координаты центра, а третье — радиус. Все данные задаются точно, и выражаются вещественными числами, не более чем с 2 знаками после десятичной точки, по модулю не превосходящими 1000.

Выходные данные

В первой строке выходного файла OUTPUT.TXT должно содержаться сообщение YES, если лес является дремучим, и NO иначе.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	3 0 10 2 5 11 2 12.04 7 2	NO
2	3 0 0 1 2.05 0 1 1.02 -1.9 1	YES

Задача №233 Автобусная экскурсия

(Время: 1 сек. Память: 16 Мб Сложность: 14%)

Оргкомитет Московской городской олимпиады решил организовать обзорную экскурсию по Москве для участников олимпиады. Для этого был заказан двухэтажный автобус (участников олимпиады достаточно много и в обычный они не умещаются) высотой 437 сантиметров. На экскурсионном маршруте встречаются N мостов. Жюри и оргкомитет олимпиады очень обеспокоены тем, что высокий двухэтажный автобус может не проехать под одним из них. Им удалось выяснить точную высоту каждого из мостов. Автобус может проехать под мостом тогда и только тогда, когда высота моста превосходит высоту автобуса.

Помогите организаторам узнать, закончится ли экскурсия благополучно, а если нет, то установить, где произойдет авария.

Входные данные

Во входном файле INPUT.TXT сначала содержится число N ($1 \leq N \leq 1000$). Далее идут N натуральных чисел, не превосходящих 10000 - высоты мостов в сантиметрах в том порядке, в котором они встречаются на пути автобуса.

Выходные данные

В единственную строку выходного файла OUTPUT.TXT нужно вывести фразу "No crash", если экскурсия закончится благополучно. Если же произойдет авария, то нужно вывести сообщение "Crash k", где k - номер моста, где произойдет авария. Фразы выводить без кавычек ровно с одним пробелом внутри.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	1 763	No crash
2	3 763 245 113	Crash 2
3	1 437	Crash 1

Задача №234

Сапер

(Время: 1 сек. Память: 16 Мб Сложность: 28%)

Мальчику Васе очень нравится известная игра "Сапер" ("Minesweeper").

В "Сапер" играет один человек. Игра идет на клетчатом поле (далее будем называть его картой) $N \times M$ (N строк, M столбцов). В K клетках поля стоят мины, в остальных клетках записано либо число от 1 до 8 — количество мин в соседних клетках, либо ничего не написано, если в соседних клетках мин нет. Клетки являются соседними, если они имеют хотя бы одну общую точку, в одной клетке не может стоять более одной мины. Изначально все клетки поля закрыты. Игрок за один ход может открыть какую-нибудь клетку. Если в открытой им клетке оказывается мина — он проигрывает, иначе игроку показывается число, которое стоит в этой клетке, и игра продолжается. Цель игры — открыть все клетки, в которых нет мин.

У Васи на компьютере есть эта игра, но ему кажется, что все карты, которые в ней есть, некрасивые и неинтересные. Поэтому он решил нарисовать свои. Однако фантазия у него богатая, а времени мало, и он хочет успеть нарисовать как можно больше карт. Поэтому он просто выбирает N , M и K и расставляет мины на поле, после чего все остальные клетки могут быть однозначно определены. Однако на определение остальных клеток он не хочет тратить свое драгоценное время. Помогите ему!

По заданным N , M , K и координатам мин восстановите полную карту.

Входные данные

В первой строке входного файла INPUT.TXT содержатся числа N , M и K ($1 \leq N \leq 200$, $1 \leq M \leq 200$, $0 \leq K \leq N \cdot M$). Далее идут K строк, в каждой из которых содержится по два числа, задающих координаты мин. Первое число в каждой строке задает номер строки клетки, где находится мина, второе число — номер столбца. Левая верхняя клетка поля имеет координаты (1,1), правая нижняя — координаты (N,M).

Выходные данные

Выходной файл OUTPUT.TXT должен содержать N строк по M символов — соответствующие строки карты. j -й символ i -й строки должен содержать символ '*' (звездочка) если в клетке (i,j) стоит мина, цифру от 1 до 8, если в этой клетке стоит соответствующее число, либо '.' (точка), если клетка (i,j) пустая.

Пример

№	INPUT.TXT	OUTPUT.TXT
	10 9 23	.111..1*1
	1 8	13*2..111
	2 3	1**3.....
	3 2	13*2.111.
	3 3	.111.2*2.
	4 3	233335*41
	5 7	*****1
	6 7	*6*7*8*41
	7 1	13*4***2.
	7 2	.1122321.
	7 3	

1	7	4
	7	5
	7	6
	7	7
	7	8
	8	1
	8	3
	8	5
	8	7
	9	3
	9	5
	9	6
	9	7

Задача №235
Робот К-79

(Время: 1 сек. Память: 16 Мб Сложность: 30%)

Петя написал программу движения робота К-79. Программа состоит из следующих команд:

- S — сделать шаг вперед
- L — повернуться на 90 градусов влево
- R — повернуться на 90 градусов вправо

Напишите программу, которая по заданной программе для робота определит, сколько шагов он сделает прежде, чем впервые вернется на то место, на котором уже побывал до этого, либо установит, что этого не произойдет.

Входные данные

Во входном файле INPUT.TXT записана одна строка из заглавных латинских букв S, L, R, описывающая программу для робота. Общее число команд в программе не превышает 200, при этом команд S — не более 50.

Выходные данные

В выходной файл OUTPUT.TXT выведите, сколько шагов будет сделано (то есть выполнено команд S) прежде, чем робот впервые окажется в том месте, через которое он уже проходил. Если такого не произойдет, выведите в выходной файл число −1.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	SSLSSLSSRSRS	5
2	LSSSS	−1

Задача №236
Многочлен

(Время: 1 сек. Память: 16 Мб Сложность: 41%)

Васе задали несколько однотипных задач по математике: «найти значение многочлена». Он хочет написать программу, которая по заданному многочлену и значению x находила бы ответ.

Напишите такую программу!

Входные данные

В первой строке входного файла INPUT.TXT записан многочлен в виде суммы одночленов. Между одночленами находится знак + или −. Перед первым одночленом может быть знак −. Одночлен записывается как [**<Коэффициент>***]**x**[**^<Степень>**] или **<Коэффициент>**, где **<Коэффициент>** – натуральное число, не превосходящее 100, **x** – символ переменной (всегда маленькая латинская буква x), **<Степень>** — натуральное число, не превосходящее 4. Параметры, взятые в квадратные скобки, могут быть опущены. Общее число символов в записи многочлена не превышает 80. Во второй строке записано одно целое число — значение x, не превышающее 100 по абсолютной величине.

Выходные данные

В выходной файл OUTPUT.TXT нужно записать одно число – значение данного многочлена при данном значении x.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	8*x+5 7	61
2	-2+x^1-3*x^2+x^2+100*x^3-2*x 0	-2

Задача №237 Головоломка

(Время: 1 сек. Память: 16 Мб Сложность: 35%)

Петя разгадывает головоломку, которая устроена следующим образом. Дана квадратная таблица размера N×N, в каждой клетке которой записана какая-нибудь латинская буква. Кроме того, дан список ключевых слов. Пете нужно, взяв очередное ключевое слово, найти его в таблице. То есть найти в таблице все буквы этого слова, причем они должны быть расположены так, чтобы клетка, в которой расположена каждая последующая буква слова, была соседней с клеткой, в которой записана предыдущая буква (клетки называются соседними, если они имеют общую сторону — то есть соседствуют по вертикали или по горизонтали). Например, на рисунке показано, как может быть расположено в таблице слово olympiad.

P	O	L	T	E
R	W	Y	M	S
O	A	I	P	T
B	D	A	N	R
L	E	M	E	S

Когда Петя находит слово, он вычеркивает его из таблицы. Использовать уже вычеркнутые буквы в других ключевых словах нельзя. После того, как найдены и вычеркнуты все ключевые слова, в таблице остаются еще несколько букв, из которых Петя должен составить слово, зашифрованное в головоломке.

Помогите Пете в решении этой головоломки, написав программу, которая по данной таблице и списку ключевых слов выпишет, из каких букв Петя должен сложить слово, то есть какие буквы останутся в таблице после вычеркивания ключевых слов.

Входные данные

Во входном файле INPUT.TXT записаны два числа N (1≤N≤10) и M (0≤M≤100). Следующие N строк по N заглавных латинских букв описывают ребус. Следующие M строк

содержат слова. Слова состоят только из заглавных латинских букв, каждое слово не длиннее 100 символов. Гарантируется, что в таблице можно найти и вычеркнуть по описанным выше правилам все ключевые слова.

Выходные данные

В выходной файл OUTPUT.TXT выведите в алфавитном порядке оставшиеся в таблице буквы.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	5 3 POLTE RWYMS OAIPT BDANR LEMES OLYMPIAD PROBLEM TEST	AENRSW
2	3 2 ISQ ABC IQW I IS	ABCQQW

Задача №238

Побег с космической станции

(Время: 1 сек. Память: 16 Мб Сложность: 52%)

Представьте, что вы состоите на службе во внешней разведке Межгалактического Альянса Республиканских Сил (МАРС). Одному из агентов разведки крупно не повезло, и он был захвачен на засекреченной космической базе. К счастью, внешней разведке МАРС удалось заполучить план этой базы. И вот теперь вам поручено разработать план побега.

База представляет собой прямоугольник размером $N \times M$, со всех сторон окружённый стенами, и состоящий из квадратных отсеков единичной площади. База снабжена K выходами, до одного из которых агенту необходимо добраться. В некоторых отсеках базы находятся стены. Ваш агент может перемещаться из отсека в любой из четырех соседних с ним, если в том отсеке, куда он хочет переместиться, нет стены.

Кроме того, база снабжена системой гипертуннелей, способных перемещать агента из одного отсека базы (вход в гипертуннель) в другой (выход из гипертуннеля). Когда агент находится в отсеке, где есть вход в гипертуннель, он может (но не обязан) им воспользоваться.

Начальное положение вашего агента известно. Вам необходимо найти кратчайший путь побега (то есть путь, проходящий через минимальное количество отсеков).

Входные данные

В первой строке входного файла INPUT.TXT записаны числа N и M ($2 \leq N \leq 100$, $2 \leq M \leq 100$), задающие размеры базы: N — количество строк в плане базы, M — количество столбцов. Во второй строке записаны начальные координаты агента X_A, Y_A ($1 \leq X_A \leq N$, $1 \leq Y_A \leq M$). Первая

координата задает номер строки, вторая — номер столбца. Строки нумеруются сверху вниз, столбцы слева направо. Далее следуют N строк по M чисел, задающих описание стен внутри базы: 1 соответствует стенке, 0 — её отсутствию. Далее в отдельной строке записано число N ($0 \leq N \leq 1000$) — количество гипертуннелей. В последующих N строках идут описания гипертуннелей. Каждый гипертуннель задается 4 числами: X_1, Y_1, X_2, Y_2 ($1 \leq X_1, X_2 \leq N; 1 \leq Y_1, Y_2 \leq M$) — координатами входа и выхода гипертуннеля. Никакие два гипертуннеля не имеют общего входа. После этого в отдельной строке следует число K ($1 \leq K \leq 10$) — количество выходов с базы. В последующих K строках идут описания выходов с базы. Каждый выход задается двумя координатами X и Y ($1 \leq X \leq N; 1 \leq Y \leq M$).

Гарантируется, что начальные координаты агента не совпадают ни с одним из выходов и он не стоит в отсеке, занятом стеной. Никакие входы и выходы гипертуннелей, а также выходы с базы не находятся в отсеках, занятых стенами. Никакой вход в гипертуннель не совпадает с выходом с базы.

Выходные данные

Если побег невозможен, выведите в выходной файл OUTPUT.TXT "Impossible". В противном случае следует вывести количество отсеков в кратчайшем пути побега.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	4 5 2 1 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 2 1 2 1 4 3 1 1 4 1 2 4	4

Задача №239

Узор

(Время: 1 сек. Память: 16 Мб Сложность: 80%)

В комнате решили сделать паркетный пол. Причем есть задумка выложить на полу некоторый узор. Плитки паркета, которыми выкладывается пол комнаты, состоят из квадратиков 1x1, каждый из которых может быть либо белым, либо черным. В свою очередь, комната имеет размеры NxM. На плане комнаты указано, какой квадрат комнаты какого цвета должен быть. Существует несколько форм паркетных плиток:



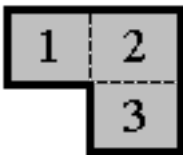
Форма 1



Форма 2



Форма 3



Форма 4

Квадратики одной паркетной плитки могут быть окрашены по-разному. Может существовать несколько типов плиток одинаковой формы, но окрашенных по-разному. Плитки разных типов могут иметь разную стоимость. Количество плиток каждого типа не ограничено. Плитки разрешается как угодно поворачивать (на углы, кратные 90 градусам). Не разрешается разламывать плитки, а также класть их лицевой стороной вниз.

Изначально, какая-то часть пола может уже быть выложена плиткой. Требуется определить минимальную стоимость плитки, необходимой для того, чтобы замостить оставшуюся часть комнаты.

Входные данные

В первой строке входного файла INPUT.TXT записаны три числа: N, M (размеры комнаты) и K (количество доступных видов плитки). $1 \leq N \leq 8$, $1 \leq M \leq 8$, $1 \leq K \leq 10$. Далее идет описание желаемой раскраски пола. Описание представляет собой N строчек по M чисел в каждой, где 0 обозначает белый цвет, 1 — черный, 2 — то, что квадрат уже выложен плиткой. В последних K строчках находятся описания доступных типов плитки в следующем формате:

<форма> <стоимость> <окраска>

<Форма> — это число от 1 до 4, описывающее форму плитки (см. рисунок выше)

<Стоимость> — это натуральное число, не превосходящее 10000, задающее стоимость одной плитки такого типа

<Окраска> — это от одного до трех чисел 0 или 1. Количество чисел совпадает с количеством квадратилов, из которых состоит плитка. Числа задают цвета квадратилов плитки в том порядке, в каком квадратики пронумерованы на рисунке.

Выходные данные

В выходной файл OUTPUT.TXT выведите единственное число — минимальную стоимость укладки или −1, если требуемым образом уложить плитку невозможно.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	4 3 3 2 2 2 2 0 0 2 1 2 2 2 2 2 10 0 0 1 5 1 4 6 0 0 1	15

Задача №240 Кубическая гостиница

(Время: 1 сек. Память: 16 Мб Сложность: 60%)

В связи с проведением межпланетного шашечного турнира было принято решение о строительстве орбитальной гостиницы. Она должна была представлять собой большой куб из $N \times N \times N$ блоков – маленьких кубиков $1 \times 1 \times 1$, и каждый блок должен был быть окрашен снаружи со всех сторон в какой-то один цвет. При этом некоторые блоки могли быть покрашены в один и тот же цвет.

Через год были сделаны фотографии гостиницы с каждой из 6 сторон: спереди, слева, сзади, справа, сверху, снизу. За год эксплуатации могло случиться так, что из-за непрочного крепления некоторые блоки, из которых была построена гостиница, оторвались и улетели в открытый космос. Комиссия по восстановлению гостиницы хочет по сделанным снимкам установить максимальное возможное количество оставшихся блоков.

Итак, вам необходимо по видам гостиницы (куба $N \times N \times N$, из которого, возможно, выкинуты некоторые кубики $1 \times 1 \times 1$) с 6 сторон определить, из какого максимального количества блоков $1 \times 1 \times 1$ она может состоять. Может оказаться так, что гостиница состоит из двух или более не связанных между собой частей.

Входные данные

В первой строке входного файла INPUT.TXT находится число N — размер гостиницы ($1 \leq N \leq 10$). На следующих N строках записаны виды гостиницы с 6 сторон (в следующем порядке: спереди, слева, сзади, справа, сверху, снизу). Каждый такой вид представляет собой таблицу $N \times N$, в которой различными заглавными латинскими буквами обозначены различные цвета, а символом «.» (точка) — то, что в этом месте можно будет смотреть прямо сквозь гостиницу. Два последовательных вида отделяются друг от друга ровно одним пробелом в каждой из N строк.

Нижняя граница вида сверху соответствует верхней границе вида спереди, а верхняя граница вида снизу — нижней границе вида спереди. Для видов спереди, сзади и с боков верх и низ вида соответствуют верху и низу гостиницы.

Входные данные корректны, то есть во входном файле описано состояние, которое может получиться.

Выходные данные

Выведите в выходной файл OUTPUT.TXT одно число — искомое максимальное количество оставшихся блоков в гостинице.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	3 .R. YYR .Y. RYY .Y. .R. GRB YGR BYG RBY GYB GRB .R. YRR .Y. RRY .R. .Y.	11
2	2 ZZ ZZ ZZ ZZ ZZ ZZ ZZ ZZ ZZ ZZ ZZ ZZ	8

Задача №241 Праздники

(Время: 1 сек. Память: 16 Мб Сложность: 39%)

Парламент некоторой страны принял новый закон о праздничных днях. Согласно этому закону первые K дней года, а также 23 февраля и 8 марта объявляются праздничными, а все остальные праздники отменяются. При этом все выходные (суббота и воскресенье), попавшие на праздничные дни, переносятся на следующие за этими праздниками рабочие дни.

В зависимости от того, на какой день недели приходится 1 января, количество нерабочих дней, которые идут подряд, может меняться.

Требуется определить, какое наибольшее количество нерабочих дней может идти подряд.

Входные данные

Во входном файле INPUT.TXT записано единственное число K ($1 \leq K \leq 50$).

Выходные данные

В выходной файл OUTPUT.TXT требуется записать единственное число — наибольшее количество нерабочих дней, идущих подряд.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	2	4
2	10	16

Задача №242 Раскраска плиток

(Время: 1 сек. Память: 16 Мб Сложность: 66%)

После того, как к удивлению тётушки Полли, её забор был покрашен, она поручила Тому Сойеру обновить краску на плитках, которыми был вымощен их квадратный двор. Двор был покрыт $N \times N$ одинаковыми квадратными плитками, каждая из которых когда-то давно была покрашена в один из K цветов ($K < N$). Краска на плитках потускнела и Тому Сойеру поручили их покрасить, на этот раз в один любой цвет (из тех же K цветов). Покрасить нужно все плитки, в том числе и те, которые уже были покрашены в этот цвет раньше.

Окунув кисть в ведро с краской один раз, можно перекрасить один горизонтальный или вертикальный ряд плиток. Чтобы разнообразить свою работу, Том придумал, что ряд плиток можно красить только цветом, которым на данный момент уже покрашены (старой или новой краской) по крайней мере две плитки выбранного ряда (вертикального или горизонтального). За один раз Том собирается красить допустимым цветом весь ряд целиком, независимо от того, были ли уже перекрашены какие-либо его плитки ранее. Помогите Тому определить, какое минимальное число раз ему придется обмакнуть кисть, чтобы перекрасить все плитки, следуя придуманным правилам, и в какой цвет окажутся окрашены все плитки.

Входные данные

В первой строке входного файла INPUT.TXT записаны через пробел два числа: N — количество плиток в одном ряду ($1 < N \leq 200$) и K ($1 \leq K < N$). В каждой из следующих N строк записаны N натуральных чисел, обозначающих номера цветов красок, в которые когда-то были выкрашены соответствующие плитки данного горизонтального ряда. Номера цветов — натуральные числа в диапазоне от 1 до K .

Выходные данные

В выходной файл OUTPUT.TXT выведите два числа: L — какое минимальное число раз придется окунать кисть в ведро с краской, и номер краски C , в которую в результате окажутся перекрашены все плитки двора. Если таких красок может быть несколько, то выведите любую из них. Если перекрасить все плитки, следуя придуманным Томом правилам, нельзя, выведите два раза число 0.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	3 2 1 2 1 2 1 1	4 1

	1 2 2	
2	2 1 1 1 1 1	2 1

Задача №243

Маскарад

(Время: 1 сек. Память: 16 Мб Сложность: 63%)

По случаю введения больших новогодних каникул устраивается великий праздничный бал-маскарад. До праздника остались считанные дни, поэтому срочно нужны костюмы для участников. Для пошивки костюмов требуется L метров ткани. Ткань продается в N магазинах, в которых предоставляются скидки оптовым покупателям. В магазинах можно купить только целое число метров ткани. Реклама магазина номер i гласит "Мы с радостью продадим Вам метр ткани за P_i рублей, однако если Вы купите не менее R_i метров, то получите прекрасную скидку — каждый купленный метр обойдется Вам всего в Q_i рублей". Чтобы воплотить в жизнь лозунг "экономика страны должна быть экономной", правительство решило потратить на закупку ткани для костюмов минимальное количество рублей из государственной казны. При этом ткани можно купить больше, чем нужно, если так окажется дешевле. Ответственный за покупку ткани позвонил в каждый магазин и узнал, что:

- реклама каждого магазина содержит правдивую информацию о ценах и скидках;
- магазин номер i готов продать ему не более F_i метров ткани.

Ответственный за покупку очень устал от проделанной работы и поэтому поставленную перед ним задачу «закупить ткань за минимальные деньги» переложил на своих помощников. Напишите программу, которая определит, сколько ткани нужно купить в каждом из магазинов так, чтобы суммарные затраты были минимальны.

Входные данные

В первой строке входного файла записаны два целых числа N и L ($1 \leq N \leq 100$, $0 \leq L \leq 100$). В каждой из последующих N строк находится описание магазина номер i — 4 целых числа P_i , R_i , Q_i , F_i ($1 \leq Q_i \leq P_i \leq 1000$, $1 \leq R_i \leq 100$, $0 \leq F_i \leq 100$).

Выходные данные

В выходной файл OUTPUT.TXT выведите минимально возможную стоимость материи. Если покупка невозможна, выведите -1.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	2 14 7 9 6 10 7 8 6 10	88
2	1 20 1 1 1 1	-1

Задача №244

Билетики

(Время: 1 сек. Память: 16 Мб Сложность: 37%)

В процессе установки турникетов в автобусах, разработчики столкнулись с проблемой проверки подлинности билета. Для ее решения был придуман следующий способ защиты от подделок.

Информация, записанная на билете, кодируется K числами (0 или 1). При этом непосредственно на билете записывается последовательность из N чисел ($N \geq K$) так, что числа, записанные на расстоянии K , совпадают. Таким образом, для проверки подлинности билета достаточно проверить, что все числа на расстоянии K совпадают. К сожалению, при считывании информации с билета иногда могут происходить ошибки — считается, что одно из чисел может исказиться (то есть 0 замениться на 1, или 1 — на 0). Такой билет все равно нужно считать подлинным. Во всех остальных случаях билет считается поддельным.

Напишите программу, которая по информации, считанной с билета, устанавливает его подлинность, и указывает на то, при считывании какого из чисел могла произойти ошибка.

Входные данные

В первой строке входного файла INPUT.TXT записаны числа N и K ($1 \leq N \leq 50000$, $1 \leq K \leq 1000$, $K \leq N$). Во второй строке записано N чисел, каждое из которых является 0 или 1 — информация, считанная с билета.

Выходные данные

В первой строке выходного файла OUTPUT.TXT должно быть записано OK, если билет подлинный и FAIL, если поддельный. В случае, если билет подлинный, во второй строке выведите 0, если все числа были считаны правильно, или номер числа, в котором при считывании произошла ошибка. Если возможных искаженных номеров несколько, выведите наименьший из них.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	6 2 1 0 1 0 1 0	OK 0
2	8 5 0 1 1 0 1 0 0 1	OK 2
3	6 2 1 1 1 0 0 0	FAIL

Задача №245 Сплоченная команда

(Время: 1 сек. Память: 16 Мб Сложность: 42%)

Как показывает опыт, для создания успешной футбольной команды важны не только умения отдельных её участников, но и сплочённость команды в целом. Характеристикой умения игрока является показатель его профессионализма (ПП). Команда является сплочённой, если ПП каждого из игроков не превосходит суммы ПП любых двух других (в частности, любая команда из одного или двух игроков является сплоченной). Перед тренерским составом молодёжной сборной была поставлена задача сформировать сплоченную сборную с максимальной суммой ПП игроков (ограничений на количество игроков в команде нет).

Ваша задача состоит в том, чтобы помочь сделать правильный выбор из N человек, для каждого из которых известен его ПП.

Входные данные

В первой строке входного файла INPUT.TXT записано целое число N ($0 \leq N \leq 30000$). В последующих N строках записано по одному целому числу P_i ($0 \leq P_i \leq 60000$), представляющему собой ПП соответствующего игрока.

Выходные данные

В выходной файл OUTPUT.TXT выведите максимально возможную сумму ПП игроков сплоченной команды.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	4 1 5 3 3	11
2	5 100 20 20 20 20	120

Задача №246 Вагоны

(Время: 1 сек. Память: 16 Мб Сложность: 28%)

Ежедневно диспетчеру железнодорожной станции приходится переставлять вагоны во многих поездах, чтобы они шли в заданном порядке. Для этого диспетчер может расцепить пришедший на станцию состав в произвольных местах и переставить образовавшиеся сцепки из одного или нескольких вагонов в произвольном порядке. Порядок вагонов в одной сцепке менять нельзя, также нельзя развернуть всю сцепку так, чтобы последний вагон в сцепке оказался первым в ней.

Диспетчер просит вашей помощи в определении того, какое минимальное число соединений между вагонами необходимо расцепить, чтобы переставить вагоны в составе в требуемом порядке.

Входные данные

В первой строке входного файла INPUT.TXT содержится целое число N ($1 \leq N \leq 100$). Во второй строке содержится перестановка натуральных чисел от 1 до N (то есть все натуральные числа от 1 до N в некотором порядке). Числа разделяются одним пробелом. Эта перестановка задает номера вагонов в приходящем на станцию составе. Требуется, чтобы в уходящем со станции составе вагоны шли в порядке их номеров.

Выходные данные

Программа должна записать в выходной файл OUTPUT.TXT единственное целое число, равное минимальному количеству соединений между вагонами, которое нужно расцепить в данном составе, чтобы их можно было переставить по порядку.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	4 3 1 2 4	2
2	5 5 4 3 2 1	4
3	2 1 2	0

Задача №247 Кафе

(Время: 1 сек. Память: 16 Мб Сложность: 46%)

Около Петиного университета недавно открылось новое кафе, в котором действует следующая система скидок: при каждой покупке более чем на 100 рублей покупатель получает купон, дающий право на один бесплатный обед (при покупке на сумму 100 рублей и меньше такой купон покупатель не получает).

Однажды Пете на глаза попался прейскурант на ближайшие N дней. Внимательно его изучив, он решил, что будет обедать в этом кафе все N дней, причем каждый день он будет покупать в кафе ровно один обед. Однако стипендия у Пети небольшая, и поэтому он хочет по максимуму использовать предоставляемую систему скидок так, чтобы его суммарные затраты были минимальны. Требуется найти минимально возможную суммарную стоимость обедов и номера дней, в которые Пете следует воспользоваться купонами.

Входные данные

В первой строке входного файла INPUT.TXT записано целое число N ($0 \leq N \leq 100$). В каждой из последующих N строк записано одно целое число, обозначающее стоимость обеда в рублях на соответствующий день. Стоимость — неотрицательное целое число, не превосходящее 300.

Выходные данные

В первой строке выходного файла OUTPUT.TXT выдайте минимальную возможную суммарную стоимость обедов. Во второй строке выдайте два числа $K1$ и $K2$ — количество купонов, которые останутся неиспользованными у Пети после этих N дней и количество использованных им купонов соответственно. Если существует несколько решений с минимальной суммарной стоимостью, то выдайте то из них, в котором значение $K1$ максимально (на случай, если Петя когда-нибудь ещё решит заглянуть в это кафе).

Пример

№	INPUT.TXT	OUTPUT.TXT
1	5 35 40 101 59 63	235 0 1

Задача №248

EuroEnglish

(Время: 1 сек. Память: 16 Мб Сложность: 55%)

Европейская комиссия планирует принять решение о том, что официальным языком Евросоюза станет английский. Был также разработан план упрощения английской письменности, который планируется реализовать за четыре года.

Первоочередной задачей будет избавление от буквы *s*, которая в сочетаниях *si* и *se* будет изменяться на *s*, в сочетании *sk* — опускаться, а в остальных случаях заменяться на *k*. При этом все замены будут производиться в строгом порядке слева направо. То есть, например, в слове *success* сначала первая из двух букв *s* заменится на *k*, а затем вторая — на *s*, то есть получится *suksess*. А слово *ssk* превратится в *kk*.

На второй год из английских слов изымут все удвоенные буквы: *ee* изменят на *i*, *oo* - на *u*, а в остальных комбинациях будут просто писать одну букву вместо двух одинаковых. Такие замены также будут делать строго в порядке слева направо. Так, слово *ooo* превратится в *uo*, а *oou* — просто в *u* (в нем сначала *oo* заменится на *u*, а затем *uu* — на *u*), слово *iee* превратится в *i* (в нем сначала *ee* заменится на *i*, а затем *ii* — на *i*).

На третий год на конце слова станут опускать букву *e*, если она не единственная буква в слове.

Наконец, завершением реформы станет отмена артиклей (в английском языке три артикля: *a*, *an* и *the*). При этом удаляться эти артикли будут только тогда, когда они в исходном тексте были словами *a*, *an*, *the*. То есть, например, текст *the table* после реформ первых трех лет превратиться в *th tabl*, а после реформы четвертого года — просто в *tabl*. А слово *aaaaa* после реформы первых лет станет словом *a*, но поскольку изначально оно не было словом *a* (артиклем), то оно в итоге так и останется словом *a*.

Напишите программу, которая будет переводить классический английский текст на Евроинглиш.

Входные данные

Во входном файле INPUT.TXT записана одна строка текста, состоящая не более чем из 200 символов: латинских строчных и заглавных букв, пробелов и знаков препинания (в тексте могут встречаться: точка, запятая, вопросительный и восклицательный знаки, двоеточие, тире, точка с запятой, открывающаяся и закрывающаяся скобки, апострофы, кавычки). Заглавные буквы могут встречаться только в начале слова. Нигде подряд не могут стоять два пробела. В начале и в конце строки не может стоять пробел. Слова отделяются друг от друга пробелами и/или знаками препинания.

Выходные данные

В выходной файл OUTPUT.TXT нужно выдать преобразованную строку при ограничениях:

- начинаться с заглавной буквы должны те и только те слова, которые начинались с заглавной буквы в исходном тексте;
- не должно встречаться двух пробелов подряд;
- пробелы между словами и знаками препинания должны остаться там и только там, где они были в исходной строке, в начале и в конце строки пробелов быть не должно.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	cacao and coffee	kakao and kofi
2	Cinderella! Where Is The Dress???	Sinderela! Wher Is Dres???

3	'A' is a letter	' ' is leter
4	!!!Hello!!!A-the-"word"	!!!Helo!!!--"word"
5	Aaaa then the ckckck	A then k
6	"A"-the an	" " _
7	A the an	
8	success	sukses

Задача №249

Скобки

(Время: 1 сек. Память: 16 Мб Сложность: 61%)

Назовем строку S правильной скобочной последовательностью, если она состоит только из символов '{', '}', '[', ']', '(', ')' и выполнено хотя бы одно из следующих трех условий:

- 1. S — пустая строка;
- 2. S можно представить в виде $S=S_1+S_2+S_3+...+S_N$ ($N>1$), где S_i — непустые правильные скобочные последовательности, а знак "+" обозначает конкатенацию (приписывание) строк;
- 3. S можно представить в виде $S='{'+C+'}'$ или $S='['+C+']'$ или $S='('+C+')'$, где C является правильной скобочной последовательностью.

Дана строка, состоящая только из символов '{', '}', '[', ']', '(', ')'. Требуется определить, какое минимальное количество символов надо вставить в эту строку для того, чтобы она стала правильной скобочной последовательностью.

Входные данные

В первой строке входного файла INPUT.TXT записана строка, состоящая только из символов '{', '}', '[', ']', '(', ')'. Длина строки не превосходит 100 символов.

Выходные данные

В выходной файл OUTPUT.TXT выведите ответ на поставленную задачу.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	{ () }	2
2	([{ }])	0

Задача №250

Двойкие числа

(Время: 1 сек. Память: 16 Мб Сложность: 27%)

Натуральное число называется двойким, если в его десятичной записи встречается не более двух различных цифр. Например, числа 3, 23, 33, 100, 12121 — двойкие, а числа 123 и 9980 — нет.

Для заданного натурального числа N требуется найти ближайшее к нему двойкое число.

Входные данные

Во входном файле INPUT.TXT записано одно натуральное число N , не превосходящее 30 000.

Выходные данные

В выходной файл OUTPUT.TXT требуется выдать единственное число — ближайшее двоякое к числу N. Если таких чисел несколько, то следует вывести наименьшее.

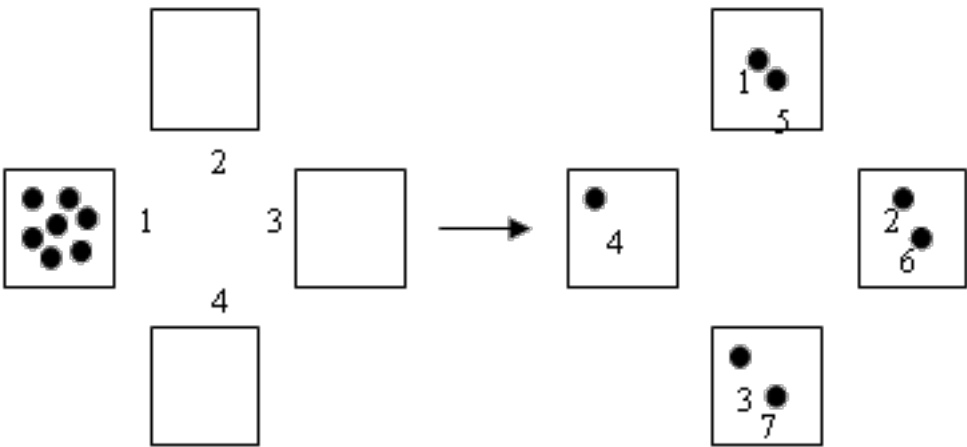
Примеры

№	INPUT.TXT	OUTPUT.TXT
1	123	122
2	2012	2020
3	11111	11111

Задача №251 Калах

(Время: 1 сек. Память: 16 Мб Сложность: 57%)

Для игры в калах используют несколько коробочек, расставленных по кругу, в которых лежат шарики. Ход осуществляется следующим образом. Берутся все шарики из одной коробочки, и начинают раскладываться по одному в коробочки подряд начиная со следующей по часовой стрелке. Если шариков больше, чем коробочек, то процесс продолжается (шарики раскладываются по второму кругу, по третьему и т.д.), пока не будут разложены все шарики. В коробочку, из которой взяли шарики, их тоже кладут. Пример одного хода приведен на рисунке. Справа шарики пронумерованы в том порядке, в котором они раскладывались по коробочкам.



Петя, тренируясь перед соревнованиями, разложил шарики по коробочкам произвольным образом, и стал делать произвольные ходы. После каждого хода он записывал номер коробочки, в которую попадал последний шарик. В некоторый момент он решил восстановить начальную конфигурацию по конечной и по тем записям, которые он делал. Напишите программу, которая поможет ему в этом.

Входные данные

В первой строке входного файла INPUT.TXT записано два натуральных числа: $N \leq 100$ — количество коробочек и $M \leq 100$ — количество сделанных Петей ходов. Коробочки пронумерованы последовательно по часовой стрелке числами от 1 до N. В следующих N строках (либо в одной строке через пробел) записано количество шариков в первой, второй, ..., N ой коробочках в конечной конфигурации. В следующих M строках (либо в одной строке через пробел) записаны номера коробочек, в которые был положен последний шарик на первом, втором, ..., M-ом ходу соответственно. Общее количество шариков не превосходит 10^9 .

Выходные данные

В выходной файл OUTPUT.TXT требуется вывести N чисел: первоначальное количество шариков в первой, второй, ..., N-ой коробочках.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	4 1 1 2 2 2 4	7 0 0 0
2	2 2 1 1 2 2	1 1

Задача №252 Сортировка масс

(Время: 1 сек. Память: 16 Мб Сложность: 36%)

Как известно, Россия является одним из ведущих экспортеров нефти. Разные страны мира, от достаточно больших до сравнительно маленьких, нуждаются в этой нефти как в воздухе. В ее состав в больших количествах входят ароматические углеводороды, которые обуславливают ее высокое качество. Доставка нефти в пункт назначения осуществляется с помощью нефтепровода. Считается, что количество нефти, отправленное в страну назначения, равно количеству полученной нефти. На самом деле это, конечно, не так. Как и многое другое, нефть воруют некоторые незаметные личности. Причем неофициально считается, что больше нефти воруют в нефтепроводах тех стран, куда нефти посылается больше (может быть, незаметные личности считают, что приносят, таким образом, меньше ущерба, кто знает...). Официальное руководство компании «Русская Нефть» решило узнать, правдивый это слух или нет, чтобы усилить (а может просто установить) охрану на тех нефтепроводах, где больше всего воруют нефть.

Для этого им нужно отсортировать нефтепроводы по количеству нефти, которая протекает в направлении какой-то страны за сутки. У компании «Русская Нефть», как и у любой уважающей себя компании, есть несколько штатных программистов, и руководство предложило им решить эту, в сущности, нетрудную задачу. Но программистов поставило в тупик то, что данные о количестве нефти представлены в разных единицах измерения (начиная от граммов и заканчивая тоннами).

Поэтому они решили найти человека, который был бы в силах решить эту задачу за них, и обещают взять его на работу в эту перспективную и процветающую компанию. Решите задачу, и, кто знает, может, повезет именно Вам?

Входные данные

В первой строке входного файла INPUT.TXT находится целое число N ($1 \leq N \leq 1000$) — количество нефтепроводов. В каждой из следующих N строк находится количество (точнее — масса) нефти, транспортированной по соответствующему нефтепроводу за сутки, по одному в строке. Масса нефти задана целым числом от 1 до 10000 с указанием соответствующей единицы измерения. Число и единица измерения разделены ровно одним пробелом. Единица измерения задается одной из трех букв: g (граммы), p (пуды), t (тонны), причем перед этой буквой может стоять одна из приставок: m (милли-), k (кило-), M (мега-), G (гига-). Напомним, что эти приставки обозначают умножение единицы измерения на 10^{-3} , 10^3 , 10^6 и 10^9 соответственно. 1 пуд = 16380 граммов, 1 тонна = 10^6 граммов.

Выходные данные

В выходной файл OUTPUT.TXT выведите N строк, в которых должны быть записаны массы

нефти в порядке неубывания. Каждая строка должна описывать массу нефти в одном из нефтепроводов. Массы должны быть описаны в том же формате, в котором записаны во входном файле. Приоритет равных масс, записанных в разных форматах должен соответствовать порядку, в котором они следуют во входном файле.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	5 234 g 4576 mp 2 t 32 mg 2 Mg	32 mg 234 g 4576 mp 2 t 2 Mg

Задача №253 Часы с боем

(Время: 1 сек. Память: 16 Мб Сложность: 25%)

Старинные часы бьют каждые полчаса. Причем в начале каждого часа они бьют столько раз, сколько сейчас часов (по 1 разу – в час ночи и в час дня, по 2 раза – в два часа ночи в два часа дня и т.д., в полночь и в полдень они бьют, соответственно, по 12 раз). И еще 1 раз они бьют в середине каждого часа.

Дан промежуток времени (известно, что прошло строго меньше 24 часов). Напишите программу, определяющую, сколько ударов сделали часы за это время.

Входные данные

В первой строке входного файла INPUT.TXT записан начальный момент времени, во второй строке — конечный. Моменты времени задаются двумя целыми числами, разделяющимися пробелом. Первое число задает часы (от 0 до 23), второе — минуты (от 1 до 59, при этом оно не равно 30).

Выходные данные

В выходной файл OUTPUT.TXT выведите одно число — сколько ударов сделали часы за этот отрезок времени.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	5 20 10 25	45
2	10 25 5 20	135
3	5 2 5 21	0

Задача №254 Выборы жрецов

В стране Олимпиадии снова выборы.

Страна состоит из маленьких графств. Графства объединяются в конфедерации. Каждая конфедерация раз в год выбирает себе покровителя – одного из 200 жрецов. Этот ритуал называется Великими Перевыборами Жрецов и выглядит так: конфедерации одновременно подают заявления (одно от конфедерации) в Совет Жрецов о том, кого они хотели бы видеть своим покровителем (если заявление не подано, то считают, что конфедерация хочет оставить себе того же покровителя). После этого все заявки удовлетворяются. Если несколько конфедераций выбирают одного и того же Жреца, то они навсегда объединяются в одну. Таким образом, каждый Жрец всегда является покровителем не более чем одной конфедерации. Требуется написать программу, позволяющую Совету Жрецов выяснить номер Жреца-покровителя каждого графства после Великих Перевыборов. В Совете все графства занумерованы (начиная с 1). Все Жрецы занумерованы числами от 1 до 200 (некоторые из них сейчас могут не быть ничьими покровителями).

Входные данные

Во входном файле INPUT.TXT записано число N – количество графств в стране ($1 \leq N \leq 5000$) – и далее для каждого графства записан номер Жреца-покровителя конфедерации, в которую оно входит (графства считаются по порядку их номеров). Затем указаны заявления от конфедераций. Сначала записано число M – количество поданных заявлений, а затем M пар чисел ($1 \leq M \leq 200$): первое число – номер текущего Жреца-покровителя, второе – номер желаемого Жреца-покровителя.

Все числа во входном файле разделяются пробелами и (или) символами перевода строки.

Выходные данные

В выходной файл OUTPUT.TXT вывести для каждого графства одно число – номер его Жреца-покровителя после Великих Перевыборов. Сначала – для первого графства, затем – для второго и т.д.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	7 1 1 5 3 1 5 1 2 5 1 1 3	3 3 1 3 3 1 3

Задача №255
Представление чисел

(Время: 1 сек. Память: 16 Мб Сложность: 40%)

Дано натуральное число N. Требуется представить его в виде суммы двух натуральных чисел A и B таких, что НОД (наибольший общий делитель) чисел A и B — максимален.

Входные данные

Во входном файле INPUT.TXT записано натуральное число N ($2 \leq N \leq 10^9$)

Выходные данные

В выходной файл OUTPUT.TXT выведите два искоемых числа А и В. Если решений несколько, выведите то из них, где А принимает наименьшее возможное значение.

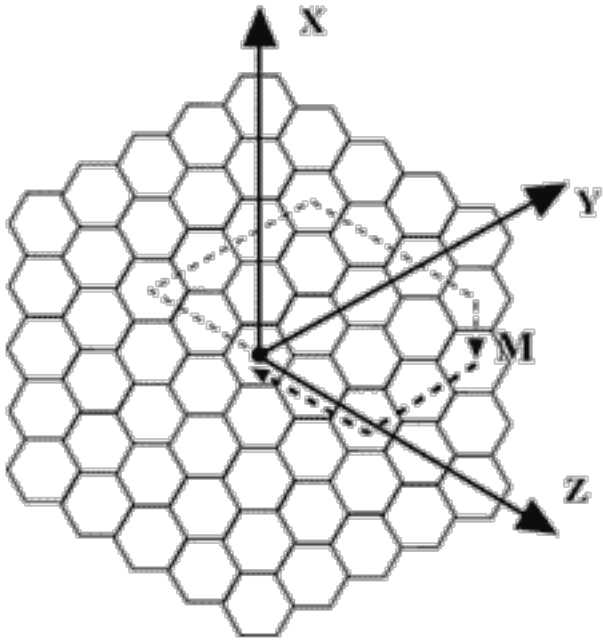
Примеры

№	INPUT.TXT	OUTPUT.TXT
1	15	5 10
2	16	8 8

Задача №256 Гексагон

(Время: 1 сек. Память: 16 Мб Сложность: 38%)

Поле для игры в новую игру "Гексагон" разбито на шестиугольники (см. рисунок). Игрок, стартуя из некоторого начального шестиугольника, сделал несколько ходов. Каждый ход заключается в перемещении фишки в соседний шестиугольник (имеющий с тем, где находилась фишка до начала хода, общую сторону) — тем самым, ход делается вдоль одного из направлений X, Y или Z (см. рисунок). Игрок записал все свои ходы, причем если фишка двигалась вдоль какого-либо направления несколько раз подряд, то в записи это обозначается указанием направления и количества ходов, которые были сделаны.



Напишите программу, которая найдет кратчайший (по количеству совершаемых ходов) путь в начальную клетку из той, где фишка оказалась после ходов игрока.

Входные данные

В первой строке входного файла INPUT.TXT записано число N — количество строк в записи перемещений фишки (1<=N<=100). Далее идет N строк с записью ходов: в каждой строке записана сначала большая буква X, Y или Z, задающая направление, затем пробел, и число, задающее количество ходов в данном направлении (число может быть и отрицательным, если игрок перемещал фишку параллельно оси, но в направлении, противоположном направлению оси). Все числа по модулю не превышают 200.

Выходные данные

В выходной файл OUTPUT.TXT выведите длину кратчайшего пути обратно в начальную клетку.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	4 Z -2 Y 3 Z 3 X -1	4

Задача №257

Кубическое уравнение

(Время: 1 сек. Память: 16 Мб Сложность: 56%)

Напишите программу, которая будет искать все целые X , удовлетворяющие уравнению $A \cdot X^3 + B \cdot X^2 + C \cdot X + D = 0$, где A, B, C, D – заданные целые коэффициенты.

Входные данные

Во входном файле INPUT.TXT записаны четыре целых числа: A, B, C, D . Все числа по модулю не превышают $2 \cdot 10^9$.

Выходные данные

В выходной файл OUTPUT.TXT выведите сначала количество различных корней этого уравнения в целых числах, а затем сами корни в возрастающем порядке. Если уравнение имеет бесконечно много корней, то следует вывести в выходной файл одно число -1 (минус один).

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	1 0 0 -27	1 3
2	0 1 2 3	0

Задача №258

Скорая помощь

(Время: 1 сек. Память: 16 Мб Сложность: 42%)

Бригада скорой помощи выехала по вызову в один из отделенных районов. К сожалению, когда диспетчер получил вызов, он успел записать только адрес дома и номер квартиры $K1$, а затем связь прервалась. Однако он вспомнил, что по этому же адресу дома некоторое время назад скорая помощь выезжала в квартиру $K2$, которая расположена в подъезде $P2$ на этаже $N2$. Известно, что в доме M этажей и количество квартир на каждой лестничной площадке одинаково. Напишите программу, которая вычисляет номер подъезда $P1$ и номер этажа $N1$ квартиры $K1$.

Входные данные

Во входном файле INPUT.TXT записаны пять положительных целых чисел $K1, M, K2, P2, N2$. Все числа не превосходят 1000.

Выходные данные

В выходной файл OUTPUT.TXT выведите два числа $P1$ и $N1$. Если входные данные не позволяют однозначно определить $P1$ или $N1$, вместо соответствующего числа напечатайте 0. Если входные данные противоречивы, то следует вывести два числа -1 (минус один).

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	89 20 41 1 11	2 3

2	11 1 1 1 1	0 1
3	3 2 2 2 1	-1 -1

Задача №259

А-функция от строки

(Время: 1 сек. Память: 16 Мб Сложность: 59%)

Дана строка S, состоящая из N символов. Определим функцию A(i) от первых i символов этой строки следующим образом:

A(i) = максимально возможному k, что равны следующие строки:

S[1]+S[2]+S[3]+...+S[k]

S[i]+S[i-1]+S[i-2]+...+S[i-k+1]

где S[i] – i-ый символ строки S, а знак + означает, что символы записываются в строчку непосредственно друг за другом.

Напишите программу, которая вычислит значения функции A для заданной строки для всех возможных значений i от 1 до N.

Входные данные

В первой строке входного файла INPUT.TXT записано одно число N. 1<=N<=200000. Во второй строке записана строка длиной N символов, состоящая только из больших и/или маленьких латинских букв.

Выходные данные

В выходной файл OUTPUT.TXT выведите N чисел — значения функции A(1), A(2), ... A(N).

Пример

№	INPUT.TXT	OUTPUT.TXT
1	5 aabaa	1 2 0 1 5

Задача №260

Олимпиада по алхимии

(Время: 1 сек. Память: 16 Мб Сложность: 53%)

В государстве алхимиков есть N населённых пунктов, пронумерованных числами от 1 до N, и M дорог. Населённые пункты бывают двух типов: деревни и города. Кроме того, в государстве есть одна столица (она может располагаться как в городе, так и в деревне). Каждая дорога соединяет два населённых пункта, и для проезда по ней требуется Ti минут. В столице было решено провести 1-ю государственную командную олимпиаду по алхимии. Для этого во все города из столицы были отправлены гонцы (по одному на город) с информацией про олимпиаду.

Напишите программу, которая посчитает, в каком порядке и через какое время каждый из гонцов доберётся до своего города. Считается, что гонец во время пути нигде не задерживается.

Входные данные

Во входном файле INPUT.TXT сначала записаны 3 числа N, M, K — количество населенных

пунктов, количество дорог и количество городов($2 \leq N \leq 1000$, $1 \leq M \leq 10000$, $1 \leq K \leq N$). Далее записан номер столицы C ($1 \leq C \leq N$). Следующие K чисел задают номера городов. Далее следуют M троек чисел S_i , E_i , T_i , описывающих дороги: S_i и E_i — номера населенных пунктов, которые соединяет данная дорога, а T_i — время для проезда по ней ($1 \leq T_i \leq 100$).

Гарантируется, что до каждого города из столицы можно добраться по дорогам (возможно, через другие населенные пункты).

Выходные данные

В выходной файл OUTPUT.TXT выведите K пар чисел: для каждого города должен быть выведен его номер и минимальное время, когда гонец может в нем оказаться (время измеряется в минутах с того момента, как гонцы выехали из столицы). Пары в выходном файле должны быть упорядочены по времени прибытия гонца, а в случае совпадения времени следует сортировать эти пары по номерам городов.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	5 4 5 1 1 2 3 4 5 1 2 1 2 3 10 3 4 100 4 5 100	1 0 2 1 3 11 4 111 5 211
2	5 5 3 1 2 4 5 2 1 1 2 3 10 3 4 100 4 5 100 1 5 1	2 1 5 1 4 101

Задача №261 Лотерея

(Время: 5 сек. Память: 16 Мб Сложность: 71%)

На одном из телеканалов каждую неделю проводится следующая лотерея. В течение недели участники делают свои ставки. Каждая ставка заключается в назывании какого-либо M -значного числа в системе счисления с основанием K (то есть, по сути, каждый участник называет M цифр, каждая из которых лежит в диапазоне от 0 до $K-1$). Ведущие нули в числах допускаются.

В некоторый момент прием ставок на текущий розыгрыш завершается, и после этого ведущий в телеэфире называет выигравшее число (это также M -значное число в K -ичной системе счисления). После этого те телезрители, у кого первая цифра их числа совпала с первой цифрой числа, названного ведущим, получают выигрыш в размере A_1 рублей. Те, у кого совпали первые две цифры числа — получают A_2 рублей (при этом если у игрока совпала вторая цифра, но не совпала первая, он не получает ничего). Аналогично угадавшие первые три цифры получают A_3 рублей. И так далее. Угадавшие все число полностью получают A_M рублей. При этом если игрок угадал t первых цифр, то он получает A_t рублей, но не получает призы за угадывание $t-1$, $t-2$ и т.д. цифр. Если игрок не угадал первую цифру, он не получает ничего.

Напишите программу, которая по известным ставкам, сделанным телезрителями, находит

число, которое должна назвать телеведущая, чтобы фирма-организатор розыгрыша выплатила в качестве выигрышей минимальную сумму. Для вашего удобства ставки, сделанные игроками, уже упорядочены по неубыванию.

Входные данные

В первой строке входного файла INPUT.TXT задаются числа N (количество телезрителей, сделавших свои ставки, $1 \leq N \leq 100000$), M (длина чисел $1 \leq M \leq 10$) и K (основание системы счисления $2 \leq K \leq 10$). В следующей строке записаны M чисел A_1, A_2, \dots, A_M , задающих выигрыши в случае совпадения только первой, первых двух, ..., всех цифр ($1 \leq A_1 \leq A_2 \leq \dots \leq A_M \leq 100000$). В каждой из следующих N строк либо в одной строке через пробел записано по одному M -значному K -ичному числу. Числа идут в порядке неубывания.

Выходные данные

В выходной файл OUTPUT.TXT выведите наименьшую сумму, которую придется выплатить в качестве выигрыша.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	10 3 2 1 3 100 000 000 001 010 100 100 100 100 110 111	6
2	1 1 10 100 0	0

Задача №262 Коммерческий калькулятор

(Время: 1 сек. Память: 16 Мб Сложность: 52%)

Фирма OISAC выпустила новую версию калькулятора. Этот калькулятор берет с пользователя деньги за совершаемые арифметические операции. Стоимость каждой операции в долларах равна 5% от числа, которое является результатом операции.

На этом калькуляторе требуется вычислить сумму N натуральных чисел (числа известны). Нетрудно заметить, что от того, в каком порядке мы будем складывать эти числа, иногда зависит, в какую сумму денег нам обойдется вычисление суммы чисел (тем самым, оказывается нарушен классический принцип «от перестановки мест слагаемых сумма не меняется» 😊).

Например, пусть нам нужно сложить числа 10, 11, 12 и 13. Тогда если мы сначала сложим 10 и 11 (это обойдется нам в \$1.05), потом результат — с 12 (\$1.65), и затем — с 13 (\$2.3), то всего мы заплатим \$5, если же сначала отдельно сложить 10 и 11 (\$1.05), потом — 12 и 13 (\$1.25) и, наконец, сложить между собой два полученных числа (\$2.3), то в итоге мы заплатим лишь \$4.6.

Напишите программу, которая будет определять, за какую минимальную сумму денег можно найти сумму данных N чисел.

Входные данные

Во входном файле INPUT.TXT записано число N ($2 \leq N \leq 100000$). Далее идет N натуральных чисел, которые нужно сложить, каждое из них не превышает 10000.

Выходные данные

В выходной файл OUTPUT.TXT выведите, сколько денег нам потребуется на нахождение суммы этих N чисел. Результат должен быть выведен с двумя знаками после десятичной точки без лидирующих нулей.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	4 10 11 12 13	4.60
2	2 1 1	0.10

Задача №263 Метро

(Время: 1 сек. Память: 16 Мб Сложность: 16%)

Витя работает недалеко от одной из станций кольцевой линии метро, а живет рядом с другой станцией той же линии. Требуется выяснить, мимо какого наименьшего количества промежуточных станций необходимо проехать Вите по кольцу, чтобы добраться с работы домой.

Входные данные

Во входном файле INPUT.TXT заданы три числа: сначала N – общее количество станций кольцевой линии, а затем i и j – номера станции, на которой Витя садится, и станции, на которой он должен выйти. Станции пронумерованы подряд натуральными числами 1, 2, 3, ..., N (1-я станция – соседняя с N-й), N не превосходит 100. Числа i и j не совпадают. Все числа разделены пробелом.

Выходные данные

В выходной файл OUTPUT.TXT требуется вывести минимальное количество промежуточных станций (не считая станции посадки и высадки), которые необходимо проехать Вите.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	100 5 6	0
2	10 1 9	1

Задача №264 Оттепель

(Время: 1 сек. Память: 16 Мб Сложность: 17%)

Уставшие от необычно теплой зимы, жители решили узнать, действительно ли это самая длинная оттепель за всю историю наблюдений за погодой. Они обратились к синоптикам, а те, в свою очередь, занялись исследованиями статистики за прошлые годы. Их интересует, сколько дней длилась самая длинная оттепель.

Оттепелью они называют период, в который среднесуточная температура ежедневно превышала 0 градусов Цельсия. Напишите программу, помогающую синоптикам в работе.

Входные данные

Во входном файле INPUT.TXT сначала записано число N – общее количество рассматриваемых дней ($1 \leq N \leq 100$). В следующей строке через пробел располагается N целых чисел, разделенных пробелами. Каждое число – среднесуточная температура в соответствующий день. Температуры – целые числа и лежат в диапазоне от –50 до 50.

Выходные данные

В выходной файл OUTPUT.TXT требуется вывести одно число – длину самой продолжительной оттепели, то есть наибольшее количество последовательных дней, на протяжении которых среднесуточная температура превышала 0 градусов. Если температура в каждый из дней была неположительной, выведите 0.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	6 -20 30 -40 50 10 -10	2
2	8 10 20 30 1 -10 1 2 3	4
3	5 -10 0 -10 0 -10	0

Задача №265 Шахматная доска

(Время: 1 сек. Память: 16 Мб Сложность: 36%)

Из шахматной доски по границам клеток выпилили связную (не распадающуюся на части) фигуру без дыр. Требуется определить ее периметр.

Входные данные

Во входном файле INPUT.TXT сначала записано число N ($1 \leq N \leq 64$) – количество выпиленных клеток. В следующих N строках указаны координаты выпиленных клеток, разделенные пробелом (номер строки и столбца – числа от 1 до 8). Каждая выпиленная клетка указывается один раз.

Выходные данные

В выходной файл OUTPUT.TXT выведите одно число – периметр выпиленной фигуры (сторона клетки равна единице).

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	3 1 1 1 2 2 1	8



2	1 8 8	4
---	----------	---

Задача №266
Кассы

(Время: 1 сек. Память: 16 Мб Сложность: 39%)

На одном из московских вокзалов билеты продают N касс. Каждая касса работает без перерыва определенный промежуток времени по фиксированному расписанию (одному и тому же каждый день). Требуется определить, на протяжении какого времени в течение суток работают все кассы одновременно.

Входные данные

Во входном файле INPUT.TXT сначала располагается одно целое число N ($0 < N \leq 1000$). В каждой из следующих N строк через пробел расположены 4 целых числа, первые два из которых обозначают время открытия кассы в часах и минутах (часы — целое число от 0 до 23, минуты — целое число от 0 до 59), остальные два — время закрытия в том же формате. Числа разделены пробелами.

Время открытия означает, что в соответствующую ему минуту касса уже работает, а время закрытия — что в соответствующую минуту касса уже не работает. Например, касса, открытая с 10 ч 30 мин до 18 ч 30 мин, ежедневно работает 480 минут.

Если времена открытия и закрытия совпадают, то это означает, что касса работает круглосуточно. Если первое время больше второго, то это означает, что касса начинает работу до полуночи, а заканчивает — на следующий день.

Выходные данные

В выходной файл OUTPUT.TXT требуется вывести одно число — суммарное время за сутки (в минутах), на протяжении которого работают все N касс.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	3 1 0 23 0 12 0 12 0 22 0 2 0	120
2	2 9 30 14 0 14 15 21 0	0
3	2 14 00 18 00 10 00 14 01	1

Задача №267
Ксерокопии

(Время: 1 сек. Память: 16 Мб Сложность: 38%)

Секретарша Ирочка сегодня опоздала на работу и ей срочно нужно успеть к обеду сделать N копий одного документа. В ее распоряжении имеются два ксерокса, один из которых копирует лист

за x секунд, а другой — за y секунд. (Разрешается использовать как один ксерокс, так и оба одновременно. Можно копировать не только с оригинала, но и с копии.) Помогите ей выяснить, какое минимальное время для этого потребуется.

Входные данные

Во входном файле INPUT.TXT записаны три натуральных числа N , x и y , разделенные пробелом ($1 \leq N \leq 2 \cdot 10^8$, $1 \leq x, y \leq 10$).

Выходные данные

В выходной файл OUTPUT.TXT выведите одно число — минимальное время в секундах, необходимое для получения N копий.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	4 1 1	3
2	5 1 2	4

Задача №268 Почти палиндром

(Время: 5 сек. Память: 64 Мб Сложность: 55%)

Слово называется палиндромом, если его первая буква совпадает с последней, вторая — с предпоследней и т.д. Например: "abba", "madam", "x".

Для заданного числа K слово называется почти палиндромом, если в нем можно изменить не более K любых букв так, чтобы получился палиндром. Например, при $K = 2$ слова "reactor", "kolobok", "madam" являются почти палиндромами (подчеркнуты буквы, заменой которых можно получить палиндром).

Подсловом данного слова являются все слова, получающиеся путем вычеркивания из данного нескольких (возможно, одной или нуля) первых букв и нескольких последних. Например, подсловами слова "cat" являются слова "c", "a", "t", "ca", "at" и само слово "cat" (а "ct" подсловом слова "cat" не является).

Требуется для данного числа K определить, сколько подслов данного слова S являются почти палиндромами.

Входные данные

В первой строке входного файла INPUT.TXT вводятся два натуральных числа: N ($1 \leq N \leq 5\,000$) — длина слова и K ($0 \leq K \leq N$). Во второй строке записано слово S , состоящее из N строчных латинских букв.

Выходные данные

В выходной файл OUTPUT.TXT требуется вывести одно число — количество подслов слова S , являющихся почти палиндромами (для данного K).

Примеры

№	INPUT.TXT	OUTPUT.TXT
---	-----------	------------

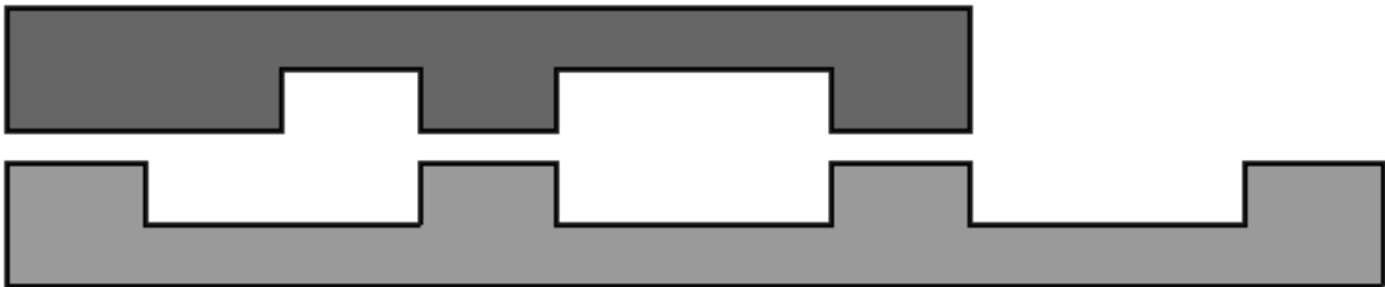
1	5 1 abcde	12
2	3 3 aaa	6

Задача №269 Тормозной механизм

(Время: 1 сек. Память: 16 Мб Сложность: 40%)

Исследовательская лаборатория одной известной автомобильной компании разработала специальный механизм, позволяющий повысить эффективность тормозов путем равномерной нагрузки деталей, используемых в тормозах.

Одним из основных компонентов механизма являются 2 прокладки, которые в процессе взаимодействия накладываются друг на друга. Каждая прокладка длины n разделена на n разделов, каждый из которых имеет высоту h или 2h. Таким образом, прокладки имеют зубчатую форму без закруглений.



В процессе взаимодействия прокладок важно, чтобы они накладывались друг на друга и при этом общая длина получившегося соединения была наименьшей.



По заданной конфигурации прокладок требуется определить наименьшую длину их возможного соединения, при котором общая высота конструкции не превышает значения 3h. При этом вращать прокладки и удалять зубцы запрещено.

Входные данные

Входной файл INPUT.TXT содержит 2 строки с описанием конфигурации 2х прокладок. Каждая конфигурация определяется последовательностью цифр 1 и 2, соответствующих высоте каждого зубца прокладки. Каждая из строк не пуста и имеет длину, не превышающую 100.

Выходные данные

В выходной файл OUTPUT.TXT требуется вывести наименьшую длину конструкции из заданных прокладок.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	2112112112 2212112	10

2	12121212 21212121	8
3	2211221122 21212	15

Задача №270

Java vs C++

(Время: 1 сек. Память: 16 Мб Сложность: 44%)

Сторонники языков Java и C++ часто спорят о том, какой язык лучше для решения олимпиадных задач. Одни говорят, что в Java есть масса полезных библиотек для работы со строками, хорошо реализованы механизмы чтения и вывода данных, а так же радует встроенные возможности для реализации длинной арифметики. С другой стороны, C++ является классическим языком, скорость выполнения программ благодаря существующим компиляторам (например, Intel Compiler 10.0) гораздо выше, чем у Java.

Но сейчас нас интересует лишь небольшие отличия, а именно соглашения, которыми пользуются программисты при описании имен переменных в Java и C++. Известно, что для понимания значений переменных часто используют английские слова или даже целые предложения, описывающие суть переменных, содержащих те или иные значения. Приведем ниже правила описания переменных, которыми руководствуются программисты, реализующие программы на Java и C++.

В языке Java принято первое слово, входящее в название переменной записывать с маленькой латинской буквы, следующее слово идет с большой буквы (только первая буква слова большая), слова не имеют разделителей и состоят только из латинских букв. Например, правильные записи переменных в Java могут выглядеть следующим образом: javaIdentifier, longAndMnemonicIdentifier, name, nEERC.

В языке C++ для описания переменных используются только маленькие латинские символы и символ «_», который отделяет непустые слова друг от друга. Примеры: java_identifier, long_and_mnemonic_identifier, name, n_e_e_r_c.

Вам требуется написать программу, которая преобразует переменную, записанную на одном языке в формат другого языка.

Входные данные

Во входном файле INPUT.TXT задано наименование переменной длиной не более 100 символов.

Выходные данные

В выходной файл OUTPUT.TXT требуется вывести аналог имени переменной в другом языке. Т.е. если переменная представлена в формате Java, то следует перевести в формат C++ и наоборот. В том случае, когда имя переменной не соответствует ни одному из вышеописанных языков, следует вывести «Error!»

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	long_and_mnemonic_identifier	longAndMnemonicIdentifier
2	anotherExample	another_example
3	i	i
4	bad_Style	Error!

Задача №271

Числа Фибоначчи - 2

(Время: 1 сек. Память: 16 Мб Сложность: 20%)

Числа Фибоначчи строятся следующим образом: 1, 1, 2, 3, 5, В этой последовательности, начиная с третьего числа, каждый следующий член равен сумме двух предыдущих. Получаем, что, например, шестое число равно 8, а десятое - 55.

Требуется написать программу, которая определяет, является ли заданное число числом Фибоначчи.

Входные данные

Входной текстовый файл INPUT.TXT содержит одно натуральное число в диапазоне от 2 до 1200000000.

Выходные данные

Выходной файл OUTPUT.TXT должен содержать в первой строке 1, если заданное число является числом Фибоначчи, и 0, иначе. В первом случае во вторую строку требуется вывести его порядковый номер.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	8	1 6
2	10	0

Задача №272

Сумма максимума и минимума

(Время: 1 сек. Память: 16 Мб Сложность: 26%)

Задана последовательность целых чисел. Числа нумеруются по порядку следования, начиная с единицы.

Требуется написать программу, которая найдет сумму максимума из чисел с четными номерами и минимума из чисел с нечетными номерами – $\max\{a_2, a_4, \dots\} + \min\{a_1, a_3, \dots\}$.

Входные данные

Входной текстовый файл INPUT.TXT содержит в единственной строке последовательность от 2 до $2 \cdot 10^5$ целых чисел, которые по модулю не превышают 10000.

Выходные данные

Выходной текстовый файл OUTPUT.TXT должен содержать одно целое число - сумму максимума из чисел с четными номерами и минимума из чисел с нечетными номерами.

Примеры

№	INPUT.TXT	OUTPUT.TXT
---	-----------	------------

1	1 2	3
2	1 -2 3 -4 5	-1

Задача №273

Вычеркивание

(Время: 1 сек. Память: 16 Мб Сложность: 28%)

Задано натуральное число N. Требуется написать программу, вычисляющую количество различных трехзначных чисел получающихся из N вычеркиванием цифр из его десятичной записи.

Входные данные

Входной текстовый файл INPUT.TXT содержит одно натуральное число N ($1 \leq N \leq 10^{100}$).

Выходные данные

Выходной текстовый файл OUTPUT.TXT должен содержать одно натуральное число - найденное количество трехзначных чисел.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	12	0
2	11111111111001111111	4

Задача №274

Дружные числа

(Время: 1 сек. Память: 16 Мб Сложность: 25%)

Будем называть два числа дружными, если они состоят из одних и тех же цифр. Например, числа 1132 и 32321 являются дружными, а 12 и 123 – нет (в первом числе нет цифры 3). Требуется написать программу, которая определит, являются ли два заданных числа дружными.

Входные данные

Входной текстовый файл INPUT.TXT содержит в первой строке натуральное число K – количество тестов. Количество тестов не превышает 10. В следующих K строках содержатся по два целых числа A и B, разделенные одним пробелом ($0 < A < 10^9$, $0 < B < 10^9$).

Выходные данные

Выходной текстовый файл OUTPUT.TXT должен содержать K строк. Для каждого теста в отдельной строке надо выдать сообщение “YES”, если A и B являются дружными, или “NO”, если не являются. В сообщениях кавычки не печатать.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	1 1132 32321	YES
2	2	NO

2	12 123 11 111	YES
---	------------------	-----

Задача №275
Делимость на 7

(Время: 1 сек. Память: 16 Мб Сложность: 42%)

Требуется определить делимость на 7 ряда целых чисел, записанных в двоичной системе счисления.

Входные данные

В первой строке входного файла INPUT.TXT содержится N – количество чисел ($N < 50$). В следующих N строках содержатся двоичные числа (по одному в каждой строке). Каждое двоичное число состоит не более чем из 1000 цифр.

Выходные данные

Выходной файл OUTPUT.TXT должен содержать N строк. Для каждого теста в отдельной строке надо выдать сообщение “Yes”, если соответствующее число кратно 7 или “No” в противном случае.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	3 1110 1010101 111111111111111111111111111111	Yes No Yes
2	1 11	No

Задача №276
Разбиение на части

(Время: 1 сек. Память: 16 Мб Сложность: 21%)

Необходимо представить целое число N в виде суммы M примерно равных целых чисел. Будем считать, что числа примерно равны, если они отличаются друг от друга не более чем на единицу.

Входные данные

Во входном файле INPUT.TXT записаны два натуральных числа N и M через пробел, каждое из которых не превосходит 30000.

Выходные данные

Выходной файл OUTPUT.TXT должен содержать M примерно равных целых чисел, сумма которых должна быть равна N. Все числа следует вывести в одной строке в порядке неубывания через пробел.

Примеры

--	--	--

№	INPUT.TXT	OUTPUT.TXT
1	13 4	3 3 3 4
2	72 8	9 9 9 9 9 9 9 9

Задача №277

Школьная алгебра

(Время: 1 сек. Память: 16 Мб Сложность: 27%)

Трёхчлен $a + bx + cy$ от двух переменных x и y однозначно определяется коэффициентами a , b и c . Написать программу, которая по заданным a , b и c выводит соответствующий трёхчлен, записанный с использованием алгебраических соглашений:

- коэффициент при члене, содержащем переменную, опускается, если его модуль равен единице;
- член, коэффициент при котором равен нулю, опускается (кроме случая, когда все коэффициенты равны нулю, тогда трехчлен состоит из одной цифры 0);
- знак "+" опускается, если он предшествует отрицательному коэффициенту;
- знак "+" опускается, если он стоит в начале выражения (так называемый унарный плюс);
- знак умножения между коэффициентом и переменной опускается.

При этом запрещено менять местами члены.

Входные данные

Во входном файле INPUT.TXT через пробел записаны целые коэффициенты a , b и c , каждое из которых не превосходит 30000 по абсолютной величине.

Выходные данные

Выходной файл OUTPUT.TXT должен содержать трехчлен, записанный с использованием алгебраических соглашений.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	0 2 -1	2x-y
2	3 0 -2	3-2y

Задача №278

Вычислительная биология

(Время: 2 сек. Память: 16 Мб Сложность: 28%)

В современной биологии ученым часто приходится иметь дело с последовательностями ДНК. Эти последовательности зачастую являются очень длинными, и их ручная обработка требует большого количества времени и сил. Поэтому возникает идея автоматизировать этот процесс.

Для этого можно применять компьютерные методы обработки данных, например, весьма полезными оказываются алгоритмы на строках. В этой задаче последовательность ДНК будет представляться в виде строки, все символы которой входят в множество $\{A, G, C, T\}$.

Пусть даны две последовательности ДНК: $s = s_1s_2 \dots s_n$ и $t = t_1t_2 \dots t_m$. Будем говорить, что t может получиться в результате эволюции из s , если s является подпоследовательностью t , то есть

существует такая последовательность индексов $1 \leq i_1 < i_2 < \dots < i_n \leq m$, что $s_1=t_{i_1}$, $s_2=t_{i_2}$, ..., $s_n=t_{i_n}$. Необходимо выяснить, может ли последовательность t получиться в результате эволюции из s .

Входные данные

Первая строка входного файла INPUT.TXT содержит последовательность s , вторая — последовательность t . Размер входного файла не превосходит 256 килобайт.

Выходные данные

В выходной файл OUTPUT.TXT выведите слово YES, если последовательность t могла получиться в результате эволюции из s , и слово NO — иначе.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	GTA AGCTA	YES
2	AAAG GAAAAAT	NO

Задача №279 Скобочки - 2

(Время: 1 сек. Память: 16 Мб Сложность: 57%)

Напомним, что называется правильной скобочной последовательностью:

- пустая строка является правильной скобочной последовательностью;
- если строка a — правильная скобочная последовательность, то строки (a) , $[a]$ — тоже правильные скобочные последовательности;
- если строки a и b — правильные скобочные последовательности, то строка ab — тоже правильная скобочная последовательность.

Задана строка S , состоящая из квадратных и круглых скобок. Разрешается заменять квадратную открывающую скобку ($[$) на круглую открывающую ($($) и наоборот, а также квадратную закрывающую скобку ($]$) на круглую закрывающую ($)$) и наоборот.

За одно действие разрешается изменить ровно один символ строки. Необходимо за минимальное число действий преобразовать S в правильную скобочную последовательность.

Входные данные

Входной файл INPUT.TXT содержит строку S . Ее длина не превосходит 100 000 символов.

Выходные данные

В выходной файл OUTPUT.TXT выведите искомое минимальное число действий или -1, если преобразовать S в правильную скобочную последовательность невозможно.

Примеры

№	INPUT.TXT	OUTPUT.TXT

1	(()) []	0
2	[()]	2
3	(([]])	-1

Задача №280

Количество делителей - 2

(Время: 1 сек. Память: 16 Мб Сложность: 51%)

Пусть x — натуральное число. Назовем y его делителем, если $1 \leq y \leq x$ и остаток от деления x на y равен нулю.

Задано число x . Найдите количество его делителей, делящихся на каждое из простых чисел, на которое делится x .

Входные данные

Входной файл INPUT.TXT содержит целое число x ($1 \leq x \leq 10^{18}$). Все простые делители числа x не превосходят тысячу.

Выходные данные

В выходной файл OUTPUT.TXT выведите ответ на задачу.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	12	2
2	239	1

Задача №281

Игра с монеткой

(Время: 1 сек. Память: 16 Мб Сложность: 39%)

Петя играет в интересную игру. Для этой игры необходима монетка. Петя подбрасывает ее n раз и считает, сколько раз выпадает «решка». Если решка выпадает хотя бы m раз, то Петя считает, что он выиграл игру.

Однажды Петя задумался, какова вероятность того, что он выиграет игру. Для этого он хочет найти количество последовательностей результатов подбрасывания монетки, содержащих ровно n подбрасываний, при которых «решка» выпала хотя бы m раз.

Помогите Пете — найдите это число, считая, что при каждом броске монетка может выпасть либо «орлом», либо «решкой».

Входные данные

Входной файл INPUT.TXT содержит два целых числа: n и m ($1 < n \leq 20, 0 \leq m \leq n$).

Выходные данные

В выходной файл OUTPUT.TXT выведите ответ на задачу.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	2 0	4
2	3 2	4

Задача №282

Прямоугольники

(Время: 1 сек. Память: 16 Мб Сложность: 47%)

Найдите количество невырожденных прямоугольников со сторонами, параллельными осям координат, вершины которых лежат в точках с целыми координатами внутри или на границе прямоугольника, противоположные углы которого находятся в точках (0, 0) и (W, H).

Входные данные

Входной файл INPUT.TXT содержит два натуральных числа W и H, не превосходящих 1000.

Выходные данные

В выходной файл OUTPUT.TXT выведите ответ на задачу.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	1 1	1
2	2 1	3
3	2 2	9

Задача №283

Рунные слова

(Время: 1 сек. Память: 16 Мб Сложность: 25%)

Руны — это древние магические знаки, которые наши предки использовали как буквы. Говорят, что рунные знаки обладают магическими свойствами, а при сложении рун в слова их магическая сила многократно возрастает. Если кузнец изготовит доспехи и начертит там определенные руны в определенном порядке, то доспехи будут наделены необычайными магическими силами.

Для того, чтобы стать обладателем таких доспехов достаточно просто принести кузнецу начертания этих рунных знаков. А вот, чтобы стать обладателем рунного знака приходилось немало потрудиться. Воины добывали начертания рун других языков и наречий в боях или получали их в качестве наград в благодарность за оказанные услуги.

Но так или иначе и в этом деле развелись жулики. По подозрениям ученых кузнец Игнатус Мошеникус изготавливал благородным воинам фальшивые рунные слова. Из древних преданий ученым стало достоверно известно, что каждая руна записывается из двух, трех или четырех латинских букв. Причем первая буква рунного слова всегда записывается как заглавная, а все остальные являются маленькими. Ученые перевели несколько, выкованных этим кузнецом, рунных слов на латинский язык и теперь нуждаются в Вашей помощи. Проверьте, является ли приведенное слово рунным.

Входные данные

В единственной строке входного файла INPUT.TXT содержится слово. Оно представляет собой непустую строку, длиной не более 100000 символов, содержащую только большие и маленькие буквы латинского алфавита.

Выходные данные

В выходной файл OUTPUT.TXT выведите «Yes», если слово является рунным и «No» в противном случае.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	IoIsTheBest	Yes
2	IoItIsWaste	No

Задача №284 Подмассив массива

(Время: 1 сек. Память: 16 Мб Сложность: 15%)

Пусть задан массив целых чисел a_1, a_2, \dots, a_n . Назовем его подмассивом $f(i, j)$ массив, составленный из чисел массива $a_i, a_{i+1}, \dots, a_{j-1}, a_j$. Напишите программу, которая будет выводить подмассивы массива a .

Входные данные

Первая строка входного файла INPUT.TXT содержит число n ($1 \leq n \leq 1000$) - количество элементов в массиве a . Во второй строке содержатся числа a_1, a_2, \dots, a_n разделенные пробелом. Все a_i находятся в диапазоне от -2^{31} до $2^{31} - 1$. В третьей строке находится m ($1 \leq m \leq 100$) — количество подмассивов, которые необходимо вывести. Следующие m строк содержат пары чисел i_k, j_k ($1 \leq i_k \leq j_k \leq n$).

Выходные данные

В выходной файл OUTPUT.TXT для каждой пары (i_k, j_k) в отдельной строке выведите подмассив $f(i_k, j_k)$.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	6 1 2 3 4 5 6 5 1 1 2 6 3 4 5 6 2 4	1 2 3 4 5 6 3 4 5 6 2 3 4

Задача №285 Костер

(Время: 1 сек. Память: 16 Мб Сложность: 34%)

Во время военного похода на морского пехотинца Джо было возложено ответственное задание – развести костёр и поддерживать в нём огонь ровно m минут. Для этого у Джо есть спички и n поленьев, причём Джо известно точное время сгорания каждого полена.

Джо разжигает огонь в момент времени $t = 0$ и сразу бросает в него одно или несколько поленьев. Затем он должен подбрасывать в огонь новые поленья, не позволяя костру угаснуть (т.е. если последнее полено в костре догорает в момент времени t , то новое полено может быть брошено в огонь не позднее $t - 1$). Поленья, брошенные в огонь, загораются мгновенно. Одновременно Джо может бросить в огонь любое количество поленьев. Джо должен бросить в огонь все n поленьев.

Помогите Джо определить, сможет ли он подбрасывать поленья в огонь таким образом, чтобы костер горел ровно m минут.

Входные данные

Первая строка входного файла INPUT.TXT содержит целые числа n и m ($1 \leq n \leq 100, 1 \leq m \leq 1000$) – количество поленьев и время, в течение которого Джо должен поддерживать огонь в костре. Вторая строка входного файла содержит n целых чисел a_1, a_2, \dots, a_n , где a_i ($2 \leq a_i \leq 1000$) – время сгорания i -ого полена в минутах.

Выходные данные

В выходной файл OUTPUT.TXT выведите строку «yes», если Джо сможет поддерживать огонь в костре ровно m минут, и строку «no» в противном случае.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	3 7 2 3 5	yes
2	2 5 3 9	no
3	4 10 3 3 3 3	no

Задача №286 Больше-меньше - 2

(Время: 1 сек. Память: 16 Мб Сложность: 37%)

Программист Билл занимается разработкой программного обеспечения для новейшего робота-исследователя, которого учёные планируют отправить на Марс с целью поиска там следов разумной жизни. Модули, которые отвечают за передвижение робота и сбор проб грунта, Билл уже скачал из Интернета. Оставалось лишь научить робота отличать разумные формы жизни от неразумных. Для этого Боб несколько месяцев посещал программистские форумы, и, наконец, нашёл подходящий модуль. Теперь, чтобы определить, является ли тот или иной объект представителем внеземной расы, роботу достаточно сравнить два вещественных числа.

Однако за несколько часов до запуска корабля на Марс обнаружилось, что робот неправильно сравнивает вещественные числа! Чтобы исправить эту ошибку, учёные обратились за помощью к Вам.

Входные данные

Входной файл INPUT.TXT состоит из двух строк, в каждой из которых записано по одному вещественному числу без ведущих нулей. Целая и дробная части отделяются точкой, которая может быть опущена, если число целое. Каждое из чисел содержит не более 10000 цифр.

Выходные данные

В выходной файл OUTPUT.TXT выведите один символ ‘<’, если первое число меньше второго, ‘>’, если больше, и ‘=’, если числа равны.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	2.39 3.61	<
2	123 12.3	>
3	12345678 12345678.0	=
4	-1.0 1.0	<

Задача №287 Профессор

(Время: 1 сек. Память: 16 Мб Сложность: 36%)

В одном очень известном университете один очень известный профессор очень быстро произносил свои лекции, так, что ничего невозможно было разобрать. Но недавно студент Вилли решил провести исследование по изучению словарного запаса профессора. С этой целью он даже посетил одну лекцию и записал всё сказанное на ней на диктофон. Затем, прокручивая дома запись с десятикратным замедлением, Вилли смог записать всё, что сказал профессор.

Но вот незадача – профессор говорил так быстро, что, даже прослушивая замедленную запись, нельзя было точно сказать, где он делал паузы между словами. Таким образом, у Вилли есть некоторый текст, состоящий из n маленьких латинских букв – лекция, которая была прочитана профессором. Теперь Вилли хочет знать, какое количество различных слов длины m мог использовать в своей лекции профессор.

Входные данные

Первая строка входного файла INPUT.TXT содержит два числа n и m ($1 \leq m \leq n \leq 100$) – длина лекции и длина слова. Вторая строка содержит n латинских символов - текст лекции профессора.

Выходные данные

В выходной файл OUTPUT.TXT выведите одно число – количество слов длины m , которые профессор мог использовать в своей лекции.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	3 1 abc	3

2	10 3 bbaabbbabb	6
---	--------------------	---

Задача №288

Комментарии

(Время: 1 сек. Память: 16 Мб Сложность: 52%)

Программист Билл недавно узнал, что, чем больше комментариев содержит исходный текст, тем он лучше. Теперь он хочет проверить, насколько хороши его собственные программы, написанные на языке Pascal. Но поскольку самому считать комментарии очень утомительно, Билл попросил Вас сделать эту работу за него.

Исходный текст может содержать комментарии трёх типов:

- 1. // ...
- 2. { ... }
- 3. (* ... *)

Комментарий первого типа начинается составным символом // и продолжается до конца строки. Комментарий второго типа начинается символом { и заканчивается символом }. Он может размещаться в нескольких строках. Комментарий третьего типа начинается составным символом (* и заканчивается составным символом *). Он также может размещаться в нескольких строках.

Комментарии не могут быть вложены друг в друга, так что запись вида {...//...(*...*)...} является одним комментарием второго типа. Комментарии не могут размещаться внутри символьных строк, так что запись ‘...(**)...{ }...’ не содержит ни одного комментария.

Входные данные

Во входном файле INPUT.TXT записан исходный текст программы на языке Pascal. Размер текста не превосходит 16 Кб.

Выходные данные

В выходной файл OUTPUT.TXT выведите одно число – количество комментариев в исходном тексте программы.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	{ my first program } begin writeln('hello world'); end.	1
2	const comments = '{ } (**)//'; begin writeln(comments); end.	0
3	// comment begin writeln('{ string }'); { while (true) do; } end.	2

Задача №289

Делители

(Время: 1 сек. Память: 16 Мб Сложность: 76%)

По заданному количеству делителей числа требуется найти само это число.

Входные данные

Во входном файле INPUT.TXT записано количество делителей D некоторого натурального числа N ($1 \leq D \leq 5000$).

Выходные данные

В выходной файл OUTPUT.TXT запишите число N . Если решений несколько, выведите наименьшее из них. Если решения нет, или наименьшее из решений превосходит $10^{15}+1$, запишите в файл число 0.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	3	4
2	4	6
3	12	60
4	60	5040
5	4911	0

Задача №290

База террористов

(Время: 1 сек. Память: 16 Мб Сложность: 42%)

Спецслужбы получили информацию о том, что в труднодоступной части Муравийской пустыни расположена хорошо замаскированная база террористов. В руки спецслужб попал и план этой базы, которая с большой высоты выглядит как группа скальных обломков, весьма часто встречающихся среди Муравийских песков. Президент отдал приказ уничтожить базу крылатыми ракетами. Ваша задача - по карте пустыни, полученной со спутника и плану базы определить количество возможных положений базы террористов. Помните, что террористы могли привезти на территорию базы камни!

Входные данные

В первой строке файла INPUT.TXT записаны числа N_b и M_b ($1 \leq N_b, M_b \leq 20$). В следующих N_b строках записан план базы. Каждая из этих строк содержит по M_b символов "#" (ASCII 35) или "." (ASCII 46). Символ "#" обозначает фрагмент базы, а символ "." - песок. В следующей строке записаны числа N_d и M_d ($1 \leq N_d, M_d \leq 100$). И остаток файла содержит карту участка пустыни на котором, предположительно, находится база террористов - N_d строк по M_d символов "#" или "." в каждой.

Выходные данные

Запишите в файл OUTPUT.TXT количество возможных положений базы террористов.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	2 2 #. ## 3 5 #.#.# ##### .###.	4
2	1 3 #.. 3 6 ##..## .#.#.# #.#...	6
3	3 3 #.. #.# #.. 5 36 #.....#.....#.. #.#.....#.#.....#..... #.....#.#.....##.....##.....###..#.....#...#.#.....##.....#.##.....#.....#.....#..	4

Задача №291
Словарь

(Время: 1 сек. Память: 16 Мб Сложность: 31%)

Дан некоторый набор букв и словарь. Ваша задача - подсчитать, сколько различных слов из словаря можно составить из этих букв.

Входные данные

В первой строке файла INPUT.TXT записано число N - количество слов в словаре (0 <= N <= 1000). В следующих N строках файла записано по одному слову из словаря. Слова содержат от 1 до 10 маленьких латинских букв. Все слова в словаре различны. В последней строке файла записан набор букв (не более 100 букв).

Выходные данные

Запишите в файл OUTPUT.TXT количество различных слов из словаря, которые можно составить из заданного набора букв.

Примеры

№	INPUT.TXT	OUTPUT.TXT
	7 ant bee	0

1	cat dog ewe fly gnu bew	
2	7 bee fly cat dog ant ewe gnu tancugd	3

Задача №292

Простой цифровой корень

(Время: 1 сек. Память: 16 Мб Сложность: 36%)

Определим простой цифровой корень (ПЦК) натурального числа N следующим образом. Если N - простое число, то ПЦК(N) = N. Если число однозначное, но не простое (то есть 1, 4, 6, 8 или 9), то ПЦК(N) = 0. В остальных случаях ПЦК(N) = ПЦК(S(N)), где S(N) - сумма цифр числа N.

Входные данные

Во входном файле INPUT.TXT записано число N ($1 \leq N \leq 2^{31}-1$).

Выходные данные

Запишите в файл OUTPUT.TXT простой цифровой корень числа N.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	1	0
2	3	3
3	128	11

Задача №293

Налоги

(Время: 1 сек. Память: 16 Мб Сложность: 20%)

В некотором государстве действует N фирм, конкурирующих между собой. У каждой фирмы есть некоторая прибыль в год, равная $V[i]$ американских рублей. У царя есть любимые фирмы, а есть нелюбимые. Соответственно, налог для всех фирм разный и назначается царем в индивидуальном порядке. Налог на i-ую фирму равен $p[i]$ процентов.

Собиратели статистики решили посчитать, с какой фирмы в государственную казну идет наибольший доход (в казну идут все налоги). К сожалению, они не учили в детстве ни математику, ни информатику (так что учитесь, дети!), и их задача резко осложняется.

Помогите им в этой нелегкой задаче.

Входные данные

Во входном файле INPUT.TXT сначала записано число N - число фирм ($0 < N \leq 100$). Далее идет N целых неотрицательных чисел, не превышающих 154 - доходы фирм, а затем еще N целых чисел от 0 до 100 - налоги фирм в процентах.

Выходные данные

В выходной файл OUTPUT.TXT выведите одно число - номер фирмы, от которой государство получает наибольший налог. Если таких фирм несколько, выведите фирму с наименьшим номером.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	1 1 1	1
2	2 1 2 3 2	2
3	3 100 1 50 0 100 3	3

Задача №294 Болты и гайки

(Время: 1 сек. Память: 16 Мб Сложность: 17%)

Вновь созданная фирма купила заброшенные склады на окраине города. Новому заведующему складами поручили произвести учёт в короткие сроки. Всё шло хорошо, пока случайно не рассыпали контейнеры с болтами и гайками на каждом складе, после чего собрали их в общие (для болтов и гаек) контейнеры, потеряв при этом несколько деталей.

Помогите оценить нанесённый ущерб на каждом складе, приняв во внимание, что, помимо потерянных деталей, болт (или гайка) считается непригодным, если он не имеет соответствующей гайки (или болта).

Входные данные

Во входном файле INPUT.TXT описано текущее положение на складе. В первой строке через пробел записаны три целых числа: k_1, l_1, m_1 – начальное число болтов ($100 \leq k_1 \leq 30000$, k_1 кратно 100), процент потерянных деталей ($0 \leq l_1 \leq 100$) и стоимость одного болта ($1 \leq m_1 \leq 100$) соответственно. Во второй строке через пробел записаны также три целых числа: k_2, l_2, m_2 – начальное число гаек ($100 \leq k_2 \leq 30000$, k_2 кратно 100), процент потерянных деталей ($0 \leq l_2 \leq 100$) и стоимость одной гайки ($1 \leq m_2 \leq 100$) соответственно.

Выходные данные

В выходной OUTPUT.TXT выведите одно целое число – размер ущерба.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	1000 10 100 1200 20 90	37000
2	5000 15 23 4000 17 22	53600

Задача №295

Шифровка

(Время: 1 сек. Память: 16 Мб Сложность: 29%)

Разведкой был перехвачен ряд шифровок, которые передавал Джеймс Бонд. Известно, что каждое послание зашифровано методом циклического сдвига. Суть которого в том, что каждая буква заменяется на букву, отстоящую в алфавите от первой на определенном расстоянии. Это расстояние называется знаменателем шифра. Так, при знаменателе шифра 2 буква D превратится в F, буква Q – в S, а Z – в B. Известно, что Бонд использует знаменатели от 0 до 25, и составляет послания исключительно из заглавных букв английского алфавита. Знаменатели в шифровках постоянно меняются, так что расшифровать содержимое послания будет не просто. После тщательного анализа удалось примерно определить предмет посланий. Теперь для каждого послания точно известно одно из входящих туда слов.

Входные данные

В первой строке входного файла INPUT.TXT содержится строка с перехваченным посланием, а во второй строке – слово, которое обязательно присутствует в этом послании. Обе строки состоят только из заглавных английских букв и содержат не больше 40 символов.

Выходные данные

В выходной файл OUTPUT.TXT выведите расшифрованный текст, либо сообщение «IMPOSSIBLE», если разгадать шифровку невозможно. В тех случаях, когда расшифровка возможна с различными знаменателями, то следует вывести вариант с наименьшим таким значением.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	HELLOAMERICA AMERICA	HELLOAMERICA
2	KHOORDPHULFD HELLOAMERICA	HELLOAMERICA
3	KHOORDPHULFD KHOORDPHULFC	IMPOSSIBLE

Задача №296

Лиса Алиса и кот Базилио

(Время: 1 сек. Память: 16 Мб Сложность: 22%)

Лиса Алиса и кот Базилио вырастили денежное дерево. И выросли на нем трехрублевые и пятирублевые золотые монеты. Лиса Алиса себе взяла трехрублевые монеты, а коту Базилио отдала пятирублевые монеты. Посетовав на свою скромность, она предложила впредь рассчитывать за покупки вместе, деньги давать без сдачи и минимальным числом монет.

Известно, что они сделали покупку стоимостью N рублей, при этом они рассчитались без сдачи.

Вам следует написать программу, которая определяет: сколько монет внес кот Базилио, и сколько монет внесла лиса Алиса.

Входные данные

Во входном файле INPUT.TXT записано одно натуральное число N – стоимость покупки в рублях ($7 < N < 1000$).

Выходные данные

В выходной OUTPUT.TXT выведите два целых числа через пробел: число монет, которые отдал кот Базилио и число монет, которые отдала лиса Алиса.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	8	1 1
2	11	1 2
3	15	3 0

Задача №297 Кругляши

(Время: 1 сек. Память: 16 Мб Сложность: 16%)

Однажды в просторах рунета появился следующий ребус:

157892 = 3

203516 = 2

409578 = 4

236271 = ?

Никто так и не смог его разгадать. Позже оказалось, что число в правом столбце равно сумме "кругляшей", которые есть в цифрах числа, расположенного слева. Ваша задача написать программу, которая определяет, сколько кругляшей в числе.

Входные данные

Во входном файле INPUT.TXT записано целое число N ($0 \leq N \leq 10^{100}$).

Выходные данные

В выходной файл OUTPUT.TXT выведите одно число – количество кругляшей в числе N.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	157892	3
2	203516	2
3	409578	4
4	236271	1

Задача №298

Стрелок

(Время: 1 сек. Память: 16 Мб Сложность: 28%)

Стрелок стоит в центре стрельбища. На стрельбище несколько мишеней. Пули стрелка пробивают мишени насквозь, не теряя скорости, и могут поразить все мишени, стоящие на одной линии.

Будем считать, что стрелок стоит в центре начала координат. Известны координаты всех мишеней (для простоты будем считать их геометрические размеры пренебрежимо малыми). Определите минимальное число выстрелов, необходимых стрелку для поражения всех мишеней.

Входные данные

Первая строка входного файла INPUT.TXT содержит натуральное число N – количество мишеней ($N \leq 20$). Далее идет N строк с информацией о координатах каждой мишени, при этом в каждой строке указывается два целых числа через пробел X и Y ($-10 \leq X, Y \leq 10$).

Выходные данные

В выходной файл OUTPUT.TXT выведите одно целое число – наименьшее количество выстрелов, необходимых для поражения всех мишеней.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	4 2 2 -2 2 -2 -2 2 -2	4
2	6 2 2 -2 2 -2 -2 2 -2 1 1 -1 3	5

Задача №299

Волейбол

(Время: 1 сек. Память: 16 Мб Сложность: 56%)

Партия в волейболе выигрывается командой, которая первой набирает 25 очков с преимуществом минимум в два очка. В случае равного счета 24-24, игра продолжается до достижения преимущества в 2 очка (26-24; 27-25).

Две сыгранные партии, закончившиеся с одинаковым счетом, будем считать разными, если строки, в которых выписан порядок набора очков командами, не равны.

Комитет по проведению соревнований по волейболу заинтересовался, количеством различных партий, заканчивающихся счетом 25:23. Их оказалось 16123801841550.

Определить, сколько существует различных партий, заканчивающихся заданным счетом.

Входные данные

Во входном файле INPUT.TXT указан конечный счет в партии (то есть такой, при котором победа в партии отдаётся одной из команд). Также известно, что ни одна из команд не набрала более 40 очков.

Выходные данные

В выходной файл OUTPUT.TXT выведите количество всевозможных партий, которые заканчиваются данным счетом.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	25:12	1251677700
2	20:25	1761039350070
3	25:23	16123801841550

Задача №300 Радар

(Время: 1 сек. Память: 16 Мб Сложность: 38%)

Радар подвергается атаке из четырех точек, являющихся вершинами квадрата, в центре которого и стоит радар. Радар укомплектован специальным щитом, позволяющим блокировать удар, но щит может защищать радар только с одной из четырех сторон, и поворот щита требует времени. Изначально щит направлен в сторону той вершины, откуда будет первая атака. Известно время запуска и скорость ракет, ведущих атаку.

Требуется определить, сколько ракет удастся отбить.

Входные данные

Первые четыре строки входного файла INPUT.TXT содержат время запуска в секундах T_x ($0 \leq T_x \leq 1000$) и скорость полета в метрах в секунду V_x x -ой ракеты ($0 < V_x \leq 1000$). Ракеты перечисляются по часовой стрелке Далее задано время в секундах, необходимое для поворота щита на 90 градусов $T_{пов}$ ($0 < T_{пов} \leq 1000$) и половина диагонали квадрата D – расстояние в метрах, предстоящее каждой из ракет ($0 < D < 1000$). Все числа – целые.

Выходные данные

В выходной файл OUTPUT.TXT выведите «ALIVE», если радар уцелеет при всех выстрелах, в противном случае следует вывести число успешно отраженных ракет.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	0 10 5 10 10 10 15 10 5 100	ALIVE
	0 10	1

2	10 10 5 10 15 10 5 100	
---	---------------------------------	--

Задача №301

Код

(Время: 1 сек. Память: 16 Мб Сложность: 46%)

В наши дни в космосе находятся сотни спутников, и все они обмениваются данными. При этом система распознавания сигналов работает по схеме «Свой-Чужой». Один из спутников отправляет запрос другому спутнику в формате двух целых чисел, а второй спутник отвечает первому так же двумя целыми числами. Первые два числа первого спутника представляют собой сумму цифр и количество цифр тех двух чисел, которыми должен ответить второй спутник. При этом в качестве ответа должны получиться числа, представляющие наибольшее и наименьшее возможные значения, которые могут быть сформированы по описанному выше методу.

Вам предстоит написать программу, формирующую ответ для второго спутника по известным числам, полученным от первого спутника.

Входные данные

Во входном файле INPUT.TXT записаны 2 натуральных числа S и K, представляющих сумму и количество цифр соответственно ($K \leq 100$). При этом гарантируется, что возможно составить хотя бы одно K-значное число, сумма цифр которого равна S.

Выходные данные

В выходной файл OUTPUT.TXT выведите два числа – ответ второго спутника. При этом следует помнить, что все числа не имеют лидирующих нулей.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	1 3	100 100
2	2 3	200 101
3	3 4	3000 1002

Задача №302

Города

(Время: 5 сек. Память: 16 Мб Сложность: 65%)

Для исследования поверхности Марса ученым необходимо разработать систему оповещения, которая смогла бы передавать информацию по цепному принципу между городами, которые планируется там построить.

При этом в каждом городе необходимо построить радиостанцию таким образом, чтобы была связь между всеми городами. При этом все такие станции должны передавать сигнал друг другу на равном расстоянии R. Таким образом, будет возможна передачи информации из одного города в другой только тогда, когда расстояние между ними не более R.

По заданным координатам городов, в целях экономии энергии радиостанций, Вам следует определить минимальное значение R, при котором информация сможет быть доставлена из любого

города во все остальные.

Входные данные

В первой строке входного файла INPUT.TXT сначала записано натуральное число N – количество городов ($N \leq 1000$). Далее идет N строк, содержащих вещественные координаты (X_i, Y_i) соответствующего города. ($-10000 \leq X_i, Y_i \leq 10000$). Предполагается, что все города находятся на плоскости.

Выходные данные

В выходной файл OUTPUT.TXT выведите одно вещественное число – наименьший радиус радиостанций. Число следует вывести с двумя знаками после запятой, без лидирующих нулей, в формате, приведенном в примерах.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	4 0 0 2 0 0 2 2 3	2 . 24
2	3 2 0 0 2 4 2	2 . 83

Задача №303 Цифры

(Время: 1 сек. Память: 16 Мб Сложность: 27%)

Составить программу, удаляющую одну цифру из N -значного числа, такую, чтобы плюс-минус сумма была наибольшей. Плюс-минус сумма – это сумма с чередованием цифр числа с разными знаками: для числа 764 это $+7-6+4$. Если удалить цифру 7, то будет $+6-4=2$, если удалить цифру 6, то будет $+7-4=3$, если удалить цифру 4, то будет $+7-6=1$. При этом видно, что максимум достигается при удалении средней цифры 6 и равен 3.

Входные данные

Во входном файле INPUT.TXT записано натуральное N -значное число ($2 \leq N \leq 50$).

Выходные данные

В выходной файл OUTPUT.TXT выведите значение наибольшей суммы.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	764	3
2	6274861	14

Задача №304

Волейбол - 2

(Время: 1 сек. Память: 16 Мб Сложность: 68%)

Партия в волейболе, выигрывается командой, которая первой набирает 25 очков с преимуществом минимум в два очка. В случае равного счета 24-24, игра продолжается до достижения преимущества в 2 очка (26-24; 27-25). Исключение составляет лишь пятая партия (когда счет по партиям 2:2), в этом случае счет ведется по такому же принципу, но до 15 очков, а в случае счета 14:14 игра продолжается так же до достижения преимущества в 2 очка.

Две сыгранные партии, закончившиеся с одинаковым счетом, будем считать разными, если строки, в которые вписан порядок набора очков командами, не равны.

Комитет по проведению соревнований по волейболу заинтересовало, сколько различных партий может быть, заканчивающихся со счетом 25:23, оказывается 16123801841550, далее им стало интересно, сколько же существует различных матчей в которых первая команда победила в 3 партиях со счетом 25:23 25:20 25:18, оказывается 10043105786927107686166271970998925000.

Определить, сколько существует различных матчей, заканчивающихся заданным счетом. Два матча закончившиеся одинаковым количеством партий с одинаковым счетом, считаются различными, если есть различно сыгранные партии.

Входные данные

Во входном файле INPUT.TXT сначала записано число N - количество партий в матче. Далее следует N пар чисел, описывающих счет в каждой партии. При этом результаты партий разделяются пробелом, а счет в каждой партии отделяется двоеточием. Гарантируется, что счет в каждой партии соответствует возможному, согласно правилам волейбола, и ни в какой партии, ни одна из команд не набирает более 40 очков.

Выходные данные

В выходной файл OUTPUT.TXT выведите количество различных матчей, которые могут оканчиваться данным счетом.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	3 25:23 25:20 25:18	10043105786927107686166271970998925000
2	4 25:23 20:25 26:24 25:18	323866095164273521651645790930981230216140667500000

Задача №305

Морской бой

(Время: 1 сек. Память: 16 Мб Сложность: 60%)

Участник игры в морской бой размещает на игровом поле свои корабли. По правилам этой разновидности игры корабли могут быть только прямоугольниками любых размеров, не могут пересекаться и иметь общих граничных точек. Количество уже размещённых кораблей равно K. Последний корабль он хочет сделать максимально большим.

Входные данные

В первой строке входного файла INPUT.TXT записаны три числа N, M и K – количество клеток по вертикали, количество клеток по горизонтали и число уже выстроенных кораблей соответственно ($1 \leq N, M \leq 200$, $1 \leq K \leq 10$). Следующие K строк содержат координаты K размещённых кораблей – 4 числа в каждой строке. 1-е и 2-е число – вертикальная и горизонтальная координаты левой верхней угловой клетки корабля, 3-е и 4-е число – вертикальная и горизонтальная координаты правой нижней угловой клетки корабля. Клетки поля нумеруются сверху вниз (от 1 до N) и слева направо (от 1 до M).

Выходные данные

В выходной файл OUTPUT.TXT выведите одно число – количество клеток в последнем корабле.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	8 7 3 1 1 2 2 3 5 3 7 4 2 4 3	21

Задача №306 Танец

(Время: 1 сек. Память: 16 Мб Сложность: 45%)

На городском празднике танцуют девушки в красных и синих юбках. Они двигаются цепочкой и выполняют сложный рисунок танца. Из цепочки девушки выделяются по одной. Первая становится на левом краю сцены, вторая уходит в конец исходной цепочки, третья – на левый край сцены (справа от первой), четвертая – в конец исходной цепочки и т.д., пока все девушки не выстроятся на краю сцены.

Помогите постановщику танца определить, каким должно быть исходное расположение девушек, если на краю сцены, они выстроены так, что их юбки чередуются по цвету (слева направо): синяя, красная, синяя, красная и т.д.

Входные данные

Во входном файле INPUT.TXT записано натуральное число N – количество танцующих девушек ($N \leq 1000$).

Выходные данные

В выходной файл OUTPUT.TXT выведите строку, содержащую цепочку из N символов, состоящую из заглавных букв B и R, соответствующих цветам юбок – синему и красному.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	2	BR
2	3	BBR
3	4	BBRR

Задача №307

Атлеты

(Время: 3 сек. Память: 16 Мб Сложность: 50%)

Художественная гимнастика – это вид спорта, где всё познаётся в сравнении, здесь нельзя, как в беге или плавании, измерить результат спортсмена с точностью до сотой доли секунды. Поэтому на выступлениях оценки выступлениям дают судьи. При выставлении оценок судьи ориентируются не только на текущее выступление, но, безусловно, сравнивают текущее выступление с выступлениями, показанными ранее. Кроме того, учитывается сложность показанного упражнения и рейтинг спортсмена.

Каждый выход спортсмена описывается двумя числами: номер спортсмена в рейтинге (наиболее профессиональные спортсмены имеют наибольший номер), и номер исполненного упражнения (упражнения нумеруются, начиная с самых простых). Судья сравнивает каждый выход спортсмена с каждым из выполненных ранее выходов. Если в результате сравнения получается, что спортсмен с большим номером показал более простое упражнение, чем спортсмен с меньшим номером, судья удивляется. Следует учитывать, что один выход может удивить судью несколько раз. Один спортсмен может выполнить несколько выходов, так же, как и одно упражнение может быть показано несколькими спортсменами – но такие выходы в сравнении судью не удивляют. Спортсмен не исполняет уже показанное упражнение повторно. Требуется подсчитать, сколько раз за время выступлений будет удивлён судья.

Входные данные

Первая строка входного файла INPUT.TXT содержит количество спортсменов N ($0 < N \leq 250$), количество упражнений M ($0 < M \leq 250$) и количество выходов P . Следующие P строк содержат по два числа, описывающие выход спортсмена – номер спортсмена и номер упражнения.

Выходные данные

В выходной файл OUTPUT.TXT выведите, сколько раз был удивлен судья.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	3 3 4 3 1 1 2 1 3 3 2	3
2	2 2 2 1 1 2 2	0

Задача №308

Вода

(Время: 1 сек. Память: 16 Мб Сложность: 70%)

Имеется три ведра, емкости которых известны и не равны. Самое большое ведро полное, остальные пусты. Требуется добиться, чтобы в самом большом ведре был заданный объем воды. За один шаг вода переливается из одного ведра в другое до тех пор, пока либо не закончится вода в ведре-источнике, либо не наполнится доверху вода в ведре-получателе.

Школьник Василий, чтобы занять себя, пытается решать эту задачу с разными входными данными, но не всегда находит решение. И даже если решение найдено, он хочет знать, является ли найденное решение оптимальным, а именно, используется ли минимальное количество шагов. Требуется написать программу, которая поможет Василию проверить его решение.

Входные данные

Во входном файле INPUT.TXT записаны 4 числа: емкости ведер B1, B2, B3 ($1000 \geq B1 > B2 > B3 > 0$) и требуемое количество воды T в первом ведре ($B1 > T > 0$).

Выходные данные

В выходной файл OUTPUT.TXT выведите либо минимальное количество переливаний, либо если задача не имеет решения, то слово IMPOSSIBLE.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	10 8 4 4	3
2	10 8 4 5	IMPOSSIBLE

Задача №309 К-удивительные числа

(Время: 3 сек. Память: 16 Мб Сложность: 30%)

Переворотом числа X назовем число, в котором все цифры числа X стоят в обратном порядке. Например, переворотом числа 6372 является число 2736, а числа 7800 - 87. Назовем К-удивительным такое число, которое в сумме со своим переворотом дает число К.

Например, у числа 222 имеется всего два К-удивительных числа: 111 и 210, а у числа 1050 имеется девять К-удивительных числа: 129, 228, 327, 426, 525, 624, 723, 822, 921.

Требуется написать программу, которая по заданному К определит количество К-удивительных чисел.

Входные данные

Входной файл INPUT.TXT содержит одно натуральное число К ($1 \leq K \leq 10^6$).

Выходные данные

Выходной файл OUTPUT.TXT должен содержать одно число - количество К-удивительных чисел.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	222	2
2	1050	9

Задача №310 Рамка из клеток

(Время: 1 сек. Память: 16 Мб Сложность: 33%)

Прямоугольник состоит из $X \times Y$ квадратных клеток одинакового размера. Из него вырезан прямоугольник размером $(X-2) \times (Y-2)$ так, что осталась рамка шириной в одну клетку. Определить, можно ли покрыть всю рамку плитками размером $A \times 1$. Запас плиток неограничен, плитки не накладываются одна на другую и за пределы рамки не выходят.

Требуется написать программу, которая решает эту задачу.

Входные данные

Входной текстовый файл INPUT.TXT содержит в первой строке натуральное число K – количество тестов ($1 \leq K \leq 10$). В следующих K строках записаны по три натуральных числа: X , Y – размеры рамки, A – размер плитки ($3 \leq X, Y \leq 2 \times 10^9, 1 \leq A \leq 2 \times 10^9$). Числа разделены пробелами.

Выходные данные

Выходной текстовый файл OUTPUT.TXT должен содержать одну строку из K символов 0 или 1 (1 - если покрытие существует, 0 - иначе).

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	1 3 3 1	1
2	2 3 3 2 3 3 3	10

Задача №311
Сумма факториалов

(Время: 1 сек. Память: 16 Мб Сложность: 45%)

Факториалом натурального числа K называется произведение $K! = 1 \times 2 \times 3 \times \dots \times K$.

Требуется написать программу, которая по заданному числу N вычислит сумму $1! + 2! + \dots + N!$.

Входные данные

Входной файл INPUT.TXT содержит одно натуральное число N ($N \leq 200$).

Выходные данные

Выходной файл OUTPUT.TXT должен содержать все десятичные знаки искомой суммы.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	1	1
2	2	3
3	3	9

Задача №312

Арифметическая прогрессия

(Время: 1 сек. Память: 16 Мб Сложность: 15%)

Заданы первый и второй элементы арифметической прогрессии. Требуется написать программу, которая вычислит элемент прогрессии по ее номеру.

Входные данные

Входной файл INPUT.TXT содержит три целых числа, разделенных пробелами – первый элемент прогрессии A_1 ($1 \leq A_1 \leq 1000$), второй элемент прогрессии A_2 ($1 \leq A_2 \leq 1000$), и номер требуемого элемента N ($1 \leq N \leq 1000$).

Выходные данные

Выходной файл OUTPUT.TXT должен содержать одно целое число - N -й элемент арифметической прогрессии.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	1 5 3	9

Пояснение к примеру

Здесь речь идет о следующей последовательности чисел:
1, 5, 9, 13, 17, 21, ...
В данной последовательности $D=4$, поэтому $A_1=1$, $A_2=A_1+D=5$, $A_3=A_1+2*D=9$, и т.д. Поскольку нам нужно было найти 3й элемент прогрессии, то ответом на задачу является число 9.

Задача №313 Ежеминутные автобусы

(Время: 1 сек. Память: 16 Мб Сложность: 30%)

На автобусную остановку каждую минуту подходит автобус одного из маршрутов. Диспетчерская служба собрала данные за N минут – номера маршрутов каждого автобуса.

Требуется определить максимально возможное время ожидания для пассажира, желающего уехать определенным маршрутом. Т.е. в данной последовательности номеров маршрутов нужно найти два самых удаленных числа, равных между собой. Например, для последовательности 2, 11, 2, 2, 25, 11, 25, 11 максимальное время ожидания равно 4 (для маршрута номер 11).

Входные данные

Входной файл INPUT.TXT содержит в первой строке число N ($2 \leq N \leq 10^6$). Во второй строке записаны N чисел – номера маршрутов. Все числа натуральные и не превышают 100. Каждый номер маршрута встречается не менее двух раз.

Выходные данные

В выходной файл OUTPUT.TXT выведите ответ на задачу.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	8 2 11 2 2 25 11 25 11	4
2	4 23 23 41 41	1

Задача №314

Лексикографический порядок чисел

(Время: 1 сек. Память: 16 Мб Сложность: 31%)

Натуральные числа от 1 до N упорядочены лексикографически. Например, для N=25 результат этого упорядочения будет таким: 1, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 2, 20, 21, 22, 23, 24, 25, 3, 4, 5, 6, 7, 8, 9.

Требуется написать программу, которая определит, на каком месте оказалось число K.

Входные данные

Входной файл INPUT.TXT содержит два натуральных числа N и K, записанных через пробел ($1 \leq K \leq N \leq 10^4$).

Выходные данные

Выходной файл OUTPUT.TXT должен содержать одно натуральное число – номер места, на котором оказалось число K.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	25 17	9

Задача №315

Наименьшая система счисления

(Время: 1 сек. Память: 16 Мб Сложность: 26%)

Известно, что основанием позиционной системы счисления называют количество различных символов, используемых для записи чисел в данной системе счисления. Также известно, что любое число x в b -ичной системе счисления имеет вид $x=a_0 \cdot b^0+a_1 \cdot b^1+...+a_n \cdot b^n$, где $b \geq 2$ и $0 \leq a_i < b$.

Для записи чисел в b -ичной системе счисления, где $b \leq 36$, могут быть использованы первые b символов из следующего списка 0,1,..., 9, A, B, ..., Z. Например, для записи чисел в троичной системы используются символы 0, 1, 2, а в двенадцатеричной - 0,1,..., 9, A, B.

Требуется написать программу, которая по входной строке S определит, является ли данная строка записью числа в системе счисления, с основанием не большим 36, и, если является, определит минимальное основание этой системы счисления.

Входные данные

Входной файл INPUT.TXT содержит в единственной строке входную строку. Длина строки не превышает 255. Все символы строки имеют коды от 32 до 127.

Выходные данные

Выходной файл OUTPUT.TXT должен содержать одно число. Если строка является записью числа в некоторой системе счисления, то нужно вывести минимальное основание такой системы счисления. Иначе вывести -1.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	123	4
2	ABCDEF	16
3	AD%AF	-1

Задача №316 Телеграфный перевод

(Время: 1 сек. Память: 16 Мб Сложность: 29%)

Телеграфный перевод оплачивается по 7 рублей за каждую полную и неполную сотню рублей. Например, за перевод 123 рублей надо заплатить 14 рублей – 7 рублей за полную сотню и 7 рублей за 23 рубля – неполную сотню. Некто попросил переслать ему зарплату в N рублей, взяв деньги за перевод из этой зарплаты.

Требуется написать программу, которая найдет, какую максимальную сумму некто сможет получить, и сколько денег будет стоить перевод.

Входные данные

Входной файл INPUT.TXT содержит одно натуральное число N ($8 \leq N \leq 6 \cdot 10^4$).

Выходные данные

Выходной файл OUTPUT.TXT должен содержать два натуральных числа – максимальную сумму и стоимость перевода. Числа разделить одним пробелом.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	10	3 7
2	108	100 7

Задача №317 Подарки Деда Мороза

(Время: 1 сек. Память: 16 Мб Сложность: 27%)

Ириска весит X грамм, мандарин – Y грамм, пряник – Z грамм.

Требуется написать программу, которая определит, сколько различных вариантов подарков весом ровно W грамм может сделать Дед Мороз.

Входные данные

В единственной строке входного файла INPUT.TXT содержится четыре целых числа X, Y, Z и

W ($1 \leq X, Y, Z \leq 100, 1 \leq W \leq 1000$).

Выходные данные

Выходной файл OUTPUT.TXT должен содержать одно целое число – количество вариантов подарков.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	10 25 15 40	3

Задача №318 Следующее число

(Время: 1 сек. Память: 16 Мб Сложность: 36%)

Задано натуральное число N.

Требуется написать программу, которая найдет следующее за ним число, в двоичном разложении которого столько же единиц, сколько в двоичном разложении числа N.

Входные данные

Входной файл INPUT.TXT содержит одно натуральное число N ($N \leq 2^{30}$).

Выходные данные

Выходной файл OUTPUT.TXT должен содержать ответ на задачу.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	1	2
2	2	4
3	3	5

Задача №319 Точки отрезка

(Время: 1 сек. Память: 16 Мб Сложность: 42%)

Концы отрезка на плоскости имеют целочисленные координаты.

Требуется написать программу, которая вычислит, сколько всего точек с целочисленными координатами принадлежат этому отрезку.

Входные данные

Входной файл INPUT.TXT содержит четыре числа – координаты концов отрезка (x1, y1) и (x2, y2). Каждая из координат не превышает по абсолютной величине значения 10^9 .

Выходные данные

Выходной файл OUTPUT.TXT должен содержать одно число – количество точек на заданном отрезке, имеющих целочисленные координаты.

Примеры

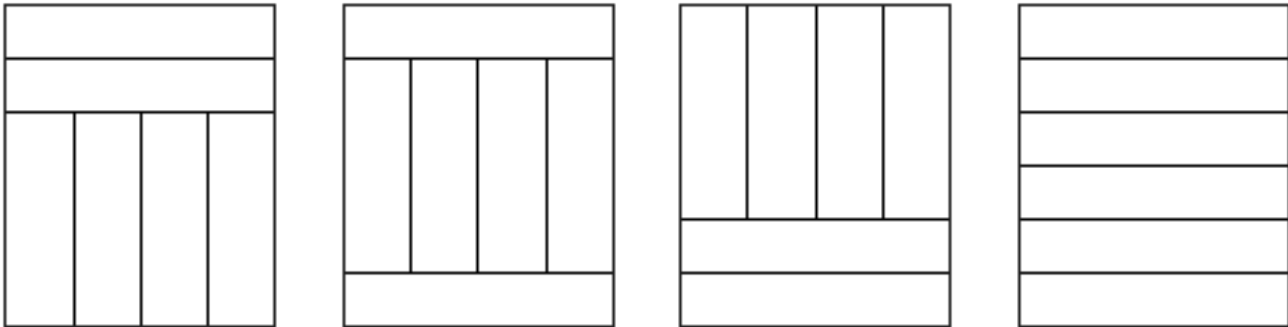
№	INPUT.TXT	OUTPUT.TXT
1	1 1 2 2	2
2	0 0 -2 -2	3
3	1 1 1 10	10

Задача №320 Коридор

(Время: 1 сек. Память: 16 Мб Сложность: 38%)

Прямоугольный коридор длиной N метров и шириной M метров решили застелить N прямоугольными плитками шириной 1 метр и длиной M метров, таким образом, чтобы не было не застеленной поверхности.

Требуется написать программу, которая найдет количество способов это сделать. Например, для коридора с размерами 6 на 4 существует четыре способа застелить плитками 1 на 4.



Входные данные

Входной файл INPUT.TXT содержит два целых числа – M (длина плитки и ширина коридора) и N (длина коридора). Для этих чисел верны неравенства $2 \leq M \leq N \leq 50$.

Выходные данные

Выходной файл OUTPUT.TXT должен содержать одно число – количество способов.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	4 6	4
2	2 2	2

Задача №321 Разные цифры

(Время: 1 сек. Память: 16 Мб Сложность: 32%)

Требуется написать программу, определяющую, в каких системах счисления с основаниями от 2 до 36 это число не содержит одинаковых цифр.

Входные данные

Входной файл INPUT.TXT содержит одно целое число N ($1 \leq N \leq 10^9$), записанное в десятичной системе счисления.

Выходные данные

Выходной файл OUTPUT.TXT должен содержать основания систем счисления в порядке возрастания, разделенные одним пробелом.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	100	11 12 13 14 15 16 17 18 20 21 22 23 25 26 27 28 29 30 31 32 33 34 35 36

Задача №322 Слово

(Время: 1 сек. Память: 16 Мб Сложность: 28%)

Числа Фибоначчи строятся следующим образом: первые два равны единице, а каждое следующее равно сумме двух предыдущих. Например, первые десять чисел Фибоначчи равны: 1, 1, 2, 3, 5, 8, 13, 21, 34, 55. В заданном тексте символы нумеруются слева направо, начиная с единицы.

Требуется написать программу, которая составит слово из символов, номера которых совпадают с числами Фибоначчи.

Входные данные

Входной файл INPUT.TXT содержит в единственной строке текст, состоящий из латинских строчных букв. В тексте не более 30000 символов.

Выходные данные

Выходной файл OUTPUT.TXT должен содержать слово из символов, номера которых совпадают с числами Фибоначчи. Символы слова идут в том же порядке, что и в заданном тексте.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	a	a
2	abc	abc
3	abcdefghij	abceh

Задача №323 Гипотеза Гольдбаха

(Время: 1 сек. Память: 16 Мб Сложность: 30%)

Известно, что любое чётное число, большее 2, представимо в виде суммы 2 простых чисел, причём таких разложений может быть несколько. Впервые гипотезу о существовании данного разложения сформулировал математик Х. Гольдбах.

Требуется написать программу, производящую согласно утверждению Гольдбаха, разложение заданного чётного числа. Из всех пар простых чисел, сумма которых равна заданному числу,

требуется найти пару, содержащую наименьшее простое число.

Входные данные

Входной файл INPUT.TXT содержит чётное число N ($4 \leq N \leq 998$).

Выходные данные

В выходной файл OUTPUT.TXT необходимо вывести два простых числа, сумма которых равна числу N . Первым выводится наименьшее число.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	6	3 3
2	992	73 919

Задача №324 Четырёхзначный палиндром

(Время: 1 сек. Память: 16 Мб Сложность: 10%)

Требуется написать программу, определяющую, является ли четырёхзначное натуральное число N палиндромом, т.е. числом, которое одинаково читается слева направо и справа налево.

Входные данные

Входной файл INPUT.TXT содержит натуральное число N ($1000 \leq N \leq 9999$).

Выходные данные

В выходной файл OUTPUT.TXT следует вывести слово «YES», если число N является палиндромом, или «NO» – если нет.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	6116	YES
2	1231	NO

Задача №325 Мы с конем вдвоем по полю пойдем

(Время: 1 сек. Память: 16 Мб Сложность: 31%)

Заданы две клетки шахматной доски. Требуется определить, возможно ли попасть из одной клетки в другую одним ходом шахматного коня, а если нет, то следует выяснить, возможно ли попасть с помощью двух ходов.

Входные данные

Входной файл INPUT.TXT содержит координаты двух клеток в общепринятом формате: каждая координата записывается как латинская строчная буква и цифра, координаты отделены друг от друга запятой и пробелом.

Выходные данные

Выходной файл OUTPUT.TXT должен содержать цифру «1», если возможно из одной клетки в другую попасть за 1 ход, либо цифру «2», если попасть можно за 2 хода, либо «NO», если одна клетка недостижима из другой ни за 1 ни за 2 хода.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	a1, h8	NO
2	a1, b3	1
3	a1, d4	2

Задача №326

Преобразование последовательности - 2

(Время: 1 сек. Память: 16 Мб Сложность: 29%)

Задана последовательность, содержащая n целых чисел. Необходимо найти число, которое встречается в этой последовательности наибольшее количество раз, а если таких чисел несколько, то найти минимальное из них, и после этого переместить все такие числа в конец заданной последовательности. Порядок расположения остальных чисел должен остаться без изменения.

Например, последовательность 1, 2, 3, 2, 3, 1, 2 после преобразования должна превратиться в последовательность 1, 3, 3, 1, 2, 2, 2.

Требуется написать программу, которая решает данную задачу.

Входные данные

Первая строка входного файла INPUT.TXT содержит число n — количество чисел во входной последовательности ($3 \leq n \leq 100$). Следующая строка содержит входную последовательность, состоящую из n целых чисел, не превышающих по модулю 100. Все числа в строке разделены пробелом.

Выходные данные

В выходной файл OUTPUT.TXT выводится последовательность чисел, которая получается в результате названного преобразования. Все числа в последовательности должны быть разделены пробелом.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	7 1 2 3 2 3 1 2	1 3 3 1 2 2 2

Задача №327

В одном шаге от счастья

(Время: 1 сек. Память: 16 Мб Сложность: 16%)

Вова купил билет в трамвае 13-го маршрута и сразу посчитал суммы первых трёх цифр и последних трёх цифр номера билета (номер у билета шестизначный). Оказалось, что суммы

отличаются ровно на единицу. «Я в одном шаге от счастья», — подумал Вова, — «или предыдущий или следующий билет точно счастливый». Прав ли он?

Входные данные

Входной файл INPUT.TXT содержит в первой строке число K – количество тестов. В следующих K строках записаны номера билетов. Количество тестов не больше 10. Номер состоит ровно из шести цифр, среди которых могут быть и нули. Гарантируется, что Вова умеет считать, то есть суммы первых трех цифр и последних трех цифр отличаются ровно на единицу.

Выходные данные

Выходной файл OUTPUT.TXT должен содержать K строк, в каждой из которых для соответствующего теста следует указать "Yes", если Вова прав, и "No", если нет.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	3 715068 445219 012200	Yes No Yes

Примечание

Трамвайный билет называется счастливым, если сумма его первых трёх цифр равна сумме его последних трёх цифр.

Задача №328 Точки на костях Домино

(Время: 1 сек. Память: 16 Мб Сложность: 25%)

Для того, чтобы заработать огромный капитал, новым русским необходимо иметь неординарное мышление. Конечно, при такой сложной работе, должны так же присутствовать какие то особенные механизмы для отдыха и развлечений. В этих целях в казино был придуман специальный набор домино для новых русских. Обычные кости домино представляют собой набор из различных комбинаций сочетаний двух плиток, на каждой из которых отображается от 0 до 6 точек. А этот набор представляет собой подобные сочетания плиток, но количество точек на каждой может быть от нуля до заданного значения, которое зависит от интеллектуального уровня игроков. В таком наборе костей присутствуют всевозможные сочетания плиток, но при этом ни одна из костей не повторяется (даже такие комбинации как 2-5 и 5-2 считаются одинаковыми).

Для изготовления данного набора костей перед изготовителем встала проблема вычисления суммарного количества точек на всех костях домино. Это связано с тем, что домино для новых русских украшается бриллиантами, которые представляют собой точки на плитках и при изготовлении необходимо оценить стоимость.

Помогите написать программу, которая решит эту задачу.

Входные данные

Входной файл INPUT.TXT содержит одно натуральное число N – максимальное количество точек на одной плитке домино. ($N \leq 10000$)

Выходные данные

В выходной файл OUTPUT.TXT выведите количество бриллиантовых камней, которые необходимо изготовить для заданного набора костей.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	2	12

Задача №329 Лесенка-2

(Время: 1 сек. Память: 16 Мб Сложность: 37%)

Вова стоит перед лесенкой из N ступеней. На каждой из ступеней написаны произвольные целые числа. Первым шагом Вова может перейти на первую ступень или, перепрыгнув через первую, сразу оказаться на второй. Также он поступает и дальше, пока не достигнет N-ой ступени. Посчитаем сумму всех чисел, написанных на ступенях через которые прошел Вова.

Требуется написать программу, которая определит оптимальный маршрут Вовы, при котором, шагая, он получит наибольшую сумму.

Входные данные

Входной файл INPUT.TXT содержит в первой строке натуральное число N – количество ступеней лестницы. Во второй строке через пробел заданы числа, написанные на ступенях лестницы, начиная с первой. Количество ступеней не превышает 1000, числа, написанные на ступенях, не превосходят по модулю 1000.

Выходные данные

Выходной файл OUTPUT.TXT должен содержать в первой строке наибольшее значение суммы. Во второй строке должны быть записаны через пробел номера ступеней по возрастанию, по которым должен шагать Вова.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	3 1 2 1	4 1 2 3
2	3 1 -1 1	2 1 3

Задача №330 Телепортация

(Время: 1 сек. Память: 16 Мб Сложность: 30%)

Вова попал на космическом корабле на бесконечную плоскую планету в точку с координатами (x_1, y_1) . Вова, управляя кораблем, имеет возможность за одну секунду телепортироваться из точки (x, y) в одну из точек $(x+C, y+C)$, $(x+C, y-C)$, $(x-C, y+C)$, $(x-C, y-C)$, где C - произвольное натуральное число, которое можно изменять на каждом ходу.

Требуется написать программу, которая определит, через какое минимальное время Вове удастся достичь точки (x_2, y_2) .

Входные данные

Входной файл INPUT.TXT содержит в первой строке числа x_1, y_1 , во второй – x_2, y_2 . Все числа целые от нуля до 10^6 . Точки (x_1, y_1) и (x_2, y_2) не совпадают.

Выходные данные

Выходной файл OUTPUT.TXT должен содержать одно число – минимальное время телепортации. Если такая телепортация невозможна, то вывести 0.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	0 0 0 2	2

Задача №331 Время прибытия

(Время: 1 сек. Память: 16 Мб Сложность: 26%)

Задано время отправления поезда и время в пути до конечной станции. Требуется написать программу, которая найдет время прибытия этого поезда (возможно, в другие сутки).

Входные данные

Входной файл INPUT.TXT содержит две строки. В первой строке задано время отправления, а во второй строке – время в пути. Время отправления задается в формате «НН:ММ», где НН время в часах, которое принимает значение от 00 до 23, ММ – время в минутах, которое принимает значение от 00 до 59. Время в пути задается двумя неотрицательными целыми числами – количество часов и количество минут. Числа разделяются одним пробелом. Количество часов не превышает 120, минут – 59.

Выходные данные

Выходной файл OUTPUT.TXT должен содержать одну строку – время прибытия поезда на конечную станцию. Формат вывода этого времени совпадает с форматом ввода времени отправления.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	00:00 10 10	10:10
2	01:02 4 6	05:08
3	11:00 22 0	09:00

Задача №332

Минимальная стоимость проезда

(Время: 1 сек. Память: 16 Мб Сложность: 40%)

На прямой ветке железной дороги расположено несколько станций. Задана стоимость проезда между любыми двумя станциями.

Требуется написать программу нахождения минимальной стоимости проезда между крайними станциями. Двигаться по железной дороге можно только в одном направлении (от станции с меньшим номером до станции с большим номером.).

Входные данные

Входной файл INPUT.TXT содержит в первой строке натуральное число N , не большее 250. Всего на дороге расположено $N+1$ станций, пронумерованных от 0 до N . В следующих строках записано $N(N+1)/2$ чисел, задающих стоимости проезда между станциями: сначала стоимость проезда от станции 0 до станций 1, 2, 3, ..., N , затем от станции 1 до станций 2, 3, ..., N , ..., от станции $N-1$ до станции N . Все стоимости проезда – неотрицательные целые числа, не превосходящие 10000.

Выходные данные

Выходной файл OUTPUT.TXT должен содержать одно число – минимальную стоимость проезда от станции 0 до станции N с возможными пересадками.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	3 7 10 20 4 8 2	12

Пояснение

В приведенном примере всего 4 станции с номерами 0, 1, 2, 3. Оптимальный маршрут проходит через станции 0, 2 и 3. Его стоимость равна $10+2=12$.

Задача №333

Общие цифры

(Время: 1 сек. Память: 16 Мб Сложность: 28%)

Заданы натуральные числа A , B , C . Требуется написать программу, которая найдет общие цифры в этих числах.

Входные данные

Входной файл INPUT.TXT содержит три натуральных числа A , B , C ($1 \leq A, B, C \leq 10^{80}$). Числа разделены одним пробелом.

Выходные данные

Выходной файл OUTPUT.TXT должен содержать в первой строке количество общих цифр, а

во второй строке в порядке возрастания через один пробел общие цифры.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	1 2 3	0
2	12 13 14	1 1
3	1234 2345 3456	2 3 4

Задача №334 Китайские часы

(Время: 1 сек. Память: 16 Мб Сложность: 41%)

Русский бизнесмен Иван Петров закупил в Китае большую партию наручных часов, чтобы продать их на родине за полцены (т.е. в 5 раз дороже, чем они стоили в Китае). Иван столкнулся с проблемой: китайские часы оказались некачественными. Мало того, что часы работали на протяжении всего нескольких часов, пока их не стукнешь, так еще и время подводить неудобно: вращать можно не минутную, а только секундную стрелку, причем, что самое ужасное, только в одну сторону в направлении увеличения времени. Например, для того, чтобы подвести часы на секунду назад, необходимо было сделать более 700 полных оборотов секундной стрелки, на что Иван бы потратил более 10 минут.

Чтобы продать эти часы оптом Ивану необходимо на момент сделки создать видимость того, что часы исправны. Для этого он собирается остановить все часы, установить их на одно и то же время. А перед сделкой ударить по чемодану с часами, чтобы они все дружно пошли.

Помогите Ивану выяснить: какое время на часах лучше установить для того, чтобы Иван потратил как можно меньше времени для того, чтобы подвести все часы.

Входные данные

В первой строке входного файла INPUT.TXT содержится натуральное число N – количество часов ($N \leq 50000$). В последующих N строках располагаются показания всех часов в формате h:mm:ss, где h – показывает который час, mm – минуты, ss - секунды ($1 \leq h \leq 12, 0 \leq mm \leq 59, 0 \leq ss \leq 59$).

Выходные данные

Выходной файл OUTPUT.TXT должен содержать время, которое нужно установить на всех часах, в формате, указанном выше. В случае неоднозначного ответа выведите наименьшее время.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	3 8:19:16 2:05:11 12:50:07	2:05:11

Задача №335

Трипростые числа

(Время: 1 сек. Память: 16 Мб Сложность: 40%)

Будем называть натуральное число *трипростым*, если в нем любые подряд идущие 3 цифры образуют трехзначное простое число.

Требуется найти количество N-значных трипростых чисел.

Входные данные

Входной файл INPUT.TXT содержит натуральное число N ($3 \leq N \leq 10000$).

Выходные данные

Выходной файл OUTPUT.TXT должен содержать количество N-значных трипростых чисел, которое следует вывести по модулю 10^9+9 .

Пример

№	INPUT.TXT	OUTPUT.TXT
1	4	204

Задача №336

Лифт

(Время: 1 сек. Память: 16 Мб Сложность: 20%)

В доме Вилли установили скоростной лифт новой экспериментальной модели. В этом лифте кнопки с номерами этажей заменены двумя другими кнопками. При нажатии на первую кнопку лифт поднимается на один этаж вверх, а при нажатии на вторую – опускается на один этаж вниз.

Младшему брату Вилли Дилли очень нравится кататься на новом лифте. Он катается на нём до тех пор, пока не побывает на каждом из этажей хотя бы по одному разу. После этого Дилли довольный возвращается домой.

Зная порядок, в котором Дилли нажимал на кнопки лифта, попробуйте определить общее количество этажей в доме Вилли и Дилли.

Входные данные

Первая строка входного файла INPUT.TXT содержит последовательность нажатий на кнопки лифта. Символ «1» означает, что была нажата первая кнопка, а символ «2» – что была нажата вторая кнопка. Символы «1» и «2» не разделены пробелами. Количество нажатий не превосходит 100. Гарантируется, что лифт никогда не опускался ниже первого и не поднимался выше последнего этажа.

Выходные данные

В выходной файл OUTPUT.TXT следует вывести одно число – количество этажей в доме Вилли и Дилли.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	11	3

2	21212	2
3	1221221221221	6

Задача №337

Лампочки

(Время: 2 сек. Память: 16 Мб Сложность: 94%)

Имеется ряд из N лампочек, которые пронумерованы от 1 до N. Изначально ни одна из лампочек не горит. Далее происходит K последовательных линейных инверсий этого ряда ламп. Под линейной инверсией понимается инверсия каждой P-й лампочки в ряде. Например, если P=3, то произойдет инверсия 3й, 6й, 9й и т.д. лампочек.

Требуется определить: сколько горящих лампочек останется после реализации всех заданных линейных инверсий?

Входные данные

В первой строке входного файла INPUT.TXT заданы числа N и K – число лампочек и число линейных инверсий. Вторая строка состоит из K целых чисел Pi, задающих период данных инверсий. (1 <= N <= 10⁹, 1<=K<=100, 1 <= Pi <= 50)

Выходные данные

В выходной файл OUTPUT.TXT следует вывести ответ на задачу.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	20 3 2 3 8	8
2	172 10 19 2 7 13 40 23 16 1 45 9	99

Задача №338

Лоскутки

(Время: 1 сек. Память: 16 Мб Сложность: 50%)

Вася Пупкин взял листочек в клетку и начал его резать по определённым линиям. На запасном листке такого же размера он закрасил клетки, по которым проходили линии. Василий Васильевич так увлёкся этим занятием, что запутался, сколько частей от листа у него осталось. Ваша задача найти это число.

Входные данные

Во входном файле INPUT.TXT в первой строке записаны N и M (0 < N, M <= 100) – размерность матрицы. Далее записана матрица из N строк, каждая из которых содержит M нулей и единиц. 0 обозначает не закрашенную клетку и 1 – закрашенную (линию разреза).

Выходные данные

В выходной файл OUTPUT.TXT следует вывести количество оставшихся частей листа.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	4 4 0 0 1 0 0 0 1 0 0 1 0 0 1 0 0 1	2

Задача №339 Мероприятие

(Время: 1 сек. Память: 16 Мб Сложность: 31%)

Мише поручили решить следующую задачу: имеется дата начала и конца некоторого мероприятия; требуется определить его длительность. Он написал программу и попросил Машу её проверить.

Через некоторое время пришла Маша и расстроила Мишу: «Твоя программа работает неправильно. По-моему, ты забыл, что года бывают високосными».

У Миши очень мало свободного времени и он не успевает исправить свою программу. Помогите ему.

Год является високосным, тогда и только тогда, когда выполнено одно из следующих условий:

- год делится на 4, но не делится на 100;
- год делится на 400.

Например, года 400, 404, 496, 504, 2000, 2004 являются високосными, а года 100, 200, 300, 503, 1000, 2001, 2005 - нет.

Входные данные

Первая строка входного файла INPUT.TXT содержит дату начала мероприятия. Вторая строка входного файла содержит дату конца мероприятия. Гарантируется, что первая дата меньше второй. Даты заданы в формате DD.MM.YYYY.

Выходные данные

В выходной файл OUTPUT.TXT выведите единственное число - длительность мероприятия (в днях).

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	01.09.2005 06.10.2005	36
2	01.09.0005 01.09.0006	366
3	01.02.2004 01.03.2004	30

Задача №340

Коробки

(Время: 1 сек. Память: 16 Мб Сложность: 19%)

На столе лежат коробка размера $A_1 \times B_1 \times C_1$ и коробка размера $A_2 \times B_2 \times C_2$. Выясните можно ли одну из этих коробок положить в другую, если разрешены повороты коробок вокруг любого ребра на угол 90 градусов.

Входные данные

Первая строка входного файла содержит три целых числа A_1 , B_1 и C_1 . Вторая строка входного файла содержит три целых числа A_2 , B_2 и C_2 . Все числа положительны и не превосходят 1000.

Выходные данные

Если коробки одинаковы, выведите "Boxes are equal". Если первая коробка может быть положена во вторую, выведите "The first box is smaller than the second one". Если вторая коробка может быть положена в первую, выведите "The first box is larger than the second one". Иначе, выведите "Boxes are incomparable".

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	1 2 3 3 2 1	Boxes are equal
2	2 2 3 3 2 1	The first box is larger than the second one
3	2 2 3 3 2 3	The first box is smaller than the second one
4	3 4 5 2 4 6	Boxes are incomparable

Задача №341

Числовая последовательность

(Время: 1 сек. Память: 16 Мб Сложность: 35%)

Дима недавно поступил на работу в научно-исследовательский институт «Числовые Последовательности». Как следует из названия этого института, основным направлением его работы является проведение различных исследований в области числовых последовательностей.

Недавно руководитель отдела, где начал работать Дима, при решении одной из проблем столкнулся с весьма интересной последовательностью чисел $a_1, a_2, \dots, a_n, \dots$, которая определяется следующим образом: $a_1 = 0$ и каждое последующее число a_i ($1 < i \leq n$) определяется как наименьшее большее натуральное число, десятичная запись которого не содержит цифр, представленных в десятичной записи a_{i-1} .

Требуется написать программу, которая по значению числа n вычисляет величину a_n .

Входные данные

Входной файл INPUT.TXT содержит натуральное число N ($N \leq 500$).

Выходные данные

В выходной файл OUTPUT.TXT выведите искомое число a_N .

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	1	0
2	28	911

Задача №342 Вписанная окружность

(Время: 1 сек. Память: 16 Мб Сложность: 68%)

Очень интересными объектами, которые изучаются в планиметрии, являются вписанные и описанные окружности. Известно, например, что вокруг любого треугольника можно описать окружность и в любой треугольник можно вписать окружность. А что будет, если вместо треугольника задан выпуклый многоугольник?

Требуется написать программу, которая определяет, можно ли в заданный выпуклый многоугольник вписать окружность, и, если это можно сделать, то вычисляет координаты ее центра и радиус.

Входные данные

Первая строка входного файла INPUT.TXT содержит количество вершин многоугольника n ($3 \leq n \leq 8$). Последующие n строк содержат координаты вершин многоугольника в порядке обхода против часовой стрелки, каждая i -ая из них содержит два целых числа: x_i и y_i , значения которых не превосходят 1000 по абсолютной величине.

Выходные данные

В первой строке выходного файла OUTPUT.TXT необходимо вывести YES, если окружность, вписанная в заданный многоугольник, существует, в противном случае следует вывести слово NO . В случае положительного ответа во второй строке следует указать координаты центра окружности и ее радиус. Числа следует выводить с тремя знаками после точки (даже если там нули), округляя их до 10^{-3} по математическим правилам. Число «0.000» следует выводить без знака «-».

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	4 0 0 1 0 1 1 0 1	YES 0.500 0.500 0.500
2	4 0 0 1 0 1 2 0 2	NO

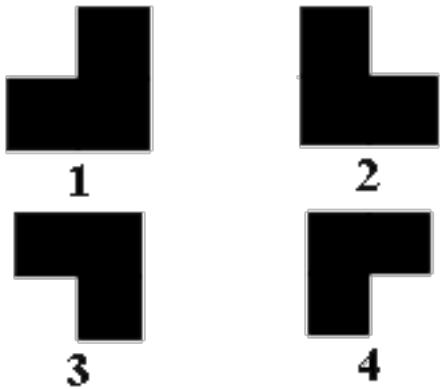
Задача №343

Укладка плиток

(Время: 1 сек. Память: 16 Мб Сложность: 35%)

Вы являетесь одним из разработчиков нового архитектурного пакета прикладных программ «CadArch». Одной из его функций является проектирование укладки половых плиток. В настоящее время вы занимаетесь программной реализацией модуля, который отвечает за укладку плиток в прямоугольных помещениях.

Для простоты будем считать, что пол помещения представляет собой прямоугольник размером n на m метров, разбитый на $m \cdot n$ квадратиков со стороной по 1 метру. Кроме этого, будем считать, что имеется четыре типа плиток, показанные в таблице. Каждая из плиток представляет собой квадрат размером 2 на 2 метра, из которого вырезан один квадратик размером 1 на 1 метр.



Проектируемый модуль должен работать следующим образом. На вход модуля подается набор команд, каждая из которых обозначает, в какое место и какого типа плитку необходимо положить. Команда обрабатывается следующим образом: если ни один из квадратиков, который должна занимать текущая плитка, не занят и плитка полностью помещается внутри прямоугольника, то плитка размещается в указанном месте, в противном случае – нет.

Требуется написать программу, которая определяет, какая площадь в соответствии с заданным набором команд будет покрыта плитками.

Входные данные

Первая строка входного файла INPUT.TXT содержит два числа n и m — длина и ширина пола помещения ($1 \leq m, n \leq 50$). Вторая строка содержит число k — количество команд, которые необходимо обработать. Каждая из последующих k строк содержит описание одной команды из набора команд. Описание команды состоит из трех чисел. Первое число определяет тип плитки (число от 1 до 4), а два других - координаты левого верхнего угла квадрата (y, x) размером 2 на 2, в который вписана соответствующая плитка ($0 \leq x, y, k \leq 1000$).

Выходные данные

В выходной файл OUTPUT.TXT необходимо вывести одно число, определяющее площадь, покрытую плитками после выполнения заданной во входном файле последовательности команд.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	4 4 4 4 1 1 2 2 2 3 1 1 1 3 3	9

Задача №344 Ближайшие точки

(Время: 1 сек. Память: 16 Мб Сложность: 38%)

Антон в школе начал изучать математику. Его внимание привлекло новое для него понятие

числовой прямой. Антон быстро научился вычислять расстояния между двумя точками на этой прямой, задавать отрезки и интервалы на ней.

Готовясь к контрольной работе, Антон столкнулся со следующей задачей: «На числовой прямой задано n точек. Необходимо найти среди них две ближайшие». Расстояние между двумя точками числовой прямой x и y равно $|x - y|$.

Требуется написать программу, которая поможет Антону решить поставленную задачу.

Входные данные

Первая строка входного файла INPUT.TXT содержит количество точек n ($2 \leq n \leq 10^5$). Вторая строка содержит n различных целых чисел x_i – координаты заданных точек числовой прямой. Числа в строке разделены пробелом. Значения всех координат x_i не превосходят 10^9 по абсолютной величине.

Выходные данные

В первой строке выходного файла OUTPUT.TXT необходимо вывести минимальное расстояние между двумя точками, заданными во входном файле. Во второй строке выходного файла необходимо вывести номера точек, которым соответствует найденное расстояние. Точки нумеруются натуральными числами от 1 до n в том порядке, в котором они заданы во входном файле. Если ответов несколько, выведите тот из них, в котором точки расположены левее других на числовой прямой. Первым выводится номер левой точки, далее через пробел – номер правой точки.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	5 10 3 6 2 5	1 4 2

Задача №345 Рекурсия

(Время: 1 сек. Память: 16 Мб Сложность: 46%)

Одним из важных понятий, используемых в теории алгоритмов, является рекурсия. Неформально ее можно определить как использование в описании объекта самого себя. Если речь идет о процедуре, то в процессе исполнения эта процедура напрямую или косвенно (через другие процедуры) вызывает сама себя.

Рекурсия является очень «мощным» методом построения алгоритмов, но таит в себе некоторые опасности. Например, неаккуратно написанная рекурсивная процедура может войти в бесконечную рекурсию, то есть, никогда не закончить свое выполнение (на самом деле, выполнение закончится с переполнением стека).

Поскольку рекурсия может быть косвенной (процедура вызывает сама себя через другие процедуры), то задача определения того факта, является ли данная процедура рекурсивной, достаточно сложна. Попробуем решить более простую задачу.

Рассмотрим программу, состоящую из n процедур P_1, P_2, \dots, P_n . Пусть для каждой процедуры известны процедуры, которые она может вызывать. Процедура P называется потенциально рекурсивной, если существует такая последовательность процедур Q_0, Q_1, \dots, Q_k , что $Q_0 = Q_k = P$ и для $i = 1 \dots k$ процедура Q_{i-1} может вызвать процедуру Q_i . В этом случае задача будет заключаться в

определении для каждой из заданных процедур, является ли она потенциально рекурсивной.

Требуется написать программу, которая позволит решить названную задачу.

Входные данные

Первая строка входного файла INPUT.TXT содержит целое число n — количество процедур в программе ($1 \leq n \leq 100$). Далее следуют n блоков, описывающих процедуры. После каждого блока следует строка, которая содержит 5 символов «*».

Описание процедуры начинается со строки, содержащий ее идентификатор, состоящий только из маленьких букв латинского алфавита и цифр. Идентификатор непуст, и его длина не превосходит 100 символов. Далее идет строка, содержащая число k ($k \leq n$) — количество процедур, которые могут быть вызваны описываемой процедурой. Последующие k строк содержат идентификаторы этих процедур — по одному идентификатору на строке.

Различные процедуры имеют различные идентификаторы. При этом ни одна процедура не может вызвать процедуру, которая не описана во входном файле.

Выходные данные

В выходной файл OUTPUT.TXT для каждой процедуры, присутствующей во входных данных, необходимо вывести слово YES, если она является потенциально рекурсивной, и слово NO — в противном случае, в том же порядке, в каком они перечислены во входных данных.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	3	YES
	p1	YES
	2	NO
	p1	
	p2	

	p2	
	1	
	p1	

	p3	
	1	
	p1	

Задача №346 Сумма двух чисел

(Время: 1 сек. Память: 16 Мб Сложность: 46%)

Заданы три числа: a , b , c . Необходимо выяснить, можно ли так переставить цифры в числах a и b , чтобы в сумме получилось c .

Входные данные

Входной файл INPUT.TXT содержит три целых числа: a , b , c ($0 < a, b, c < 10^9$). Числа разделены пробелом.

Выходные данные

В выходной файл OUTPUT.TXT следует вывести YES, если искомая перестановка цифр возможна, в противном случае необходимо вывести NO. При положительном ответе во второй строке следует вывести число x, получаемое перестановкой цифр числа a, и число y, получаемое перестановкой цифр числа b, сумма которых равна c. Числа x и y при выводе не должны содержать ведущих нулей. Числа в строке разделены пробелом. Если решений несколько, то следует вывести ту пару, в которой число x минимально.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	12 31 25	YES 12 13
2	12 31 26	NO
3	101 2 13	YES 11 2

Задача №347 Покер

(Время: 1 сек. Память: 16 Мб Сложность: 33%)

Имеется 5 целых чисел. Среди них:

- если одинаковы 5, то вывести "Impossible", иначе
- если одинаковы 4, то вывести "Four of a Kind", иначе
- если одинаковы 3 и 2, то вывести "Full House", иначе
- если есть 5 последовательных, то вывести "Straight", иначе
- если одинаковы 3, то вывести "Three of a Kind", иначе
- если одинаковы 2 и 2, то вывести "Two Pairs", иначе
- если одинаковы 2, то вывести "One Pair", иначе
- вывести "Nothing".

Входные данные

Входной файл INPUT.TXT содержит 5 целых чисел от 1 до 13, разделенных пробелом.

Выходные данные

В выходной файл OUTPUT.TXT следует вывести результат анализа.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	1 3 9 3 2	One Pair
2	1 5 5 4 4	Two Pairs
3	1 5 2 4 3	Straight
4	10 11 12 13 1	Nothing

Задача №348

Пересечение отрезков

(Время: 1 сек. Память: 16 Мб Сложность: 49%)

Два отрезка на плоскости заданы целочисленными координатами своих концов в декартовой системе координат. Требуется определить, существует ли у них общая точка.

Входные данные

Входной файл INPUT.TXT содержит координаты четырех точек, задающих отрезки. В первой строке содержатся координаты первого конца первого отрезка , во второй - второго конца первого отрезка, в третьей и четвёртой - координаты концов второго отрезка. Все координаты – целые числа, не превосходящие 10000 по абсолютной величине.

Выходные данные

В выходной файл OUTPUT.TXT следует вывести слово "Yes", если общая точка есть, или слово "No" - в противном случае.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	0 0 1 1 0 1 1 0	Yes
2	0 0 1 0 0 1 1 1	No

Задача №349 Простые числа

(Время: 1 сек. Память: 16 Мб Сложность: 28%)

Необходимо вывести все простые числа от М до N включительно.

Входные данные

Входной файл INPUT.TXT содержит два натуральных числа М и N, разделенных пробелом ($2 \leq M \leq N \leq 10^6$)

Выходные данные

В выходной файл OUTPUT.TXT выведите в одной строке через пробел все простые числа от М до N в порядке возрастания. Если таковых чисел нет, то следует вывести «Absent».

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	2 5	2 3 5
2	4 4	Absent

Задача №350

Перестановки

(Время: 1 сек. Память: 16 Мб Сложность: 44%)

Дана строка, состоящая из N попарно различных символов. Требуется вывести все перестановки символов данной строки в алфавитном порядке.

Входные данные

Входной файл INPUT.TXT содержит строку, состоящую не более чем из N символов. ($1 \leq N \leq 8$)

Выходные данные

В выходной файл OUTPUT.TXT выведите все элементы искомой последовательности по одному в каждой строке.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	AB	AB BA
2	123	123 132 213 231 312 321

Задача №351

Прыжки по буквам

(Время: 1 сек. Память: 16 Мб Сложность: 56%)

Дана цепочка из N символов, состоящая из прописных латинских букв. Необходимо пройти с первого символа цепочки до последнего символа, прыгая не более чем на K символов. Стоимость прыжка, при котором символ не меняется, равна 0, а стоимость прыжка на другой символ равна 1.

Требуется написать программу, которая вычислит наименьшую стоимость перехода с первого на последний символ.

Входные данные

Входной файл INPUT.TXT содержит в первой строке два целых числа: длина цепочки N ($2 \leq N \leq 10^5$) и максимальная длина прыжка K ($1 \leq K < N$). Во второй строке содержится цепочка из N латинских букв.

Выходные данные

Выходной файл OUTPUT.TXT должен содержать одно число – минимальную стоимость перехода.

Пример

--	--	--

№	INPUT.TXT	OUTPUT.TXT
1	10 2 АВАВВСАСВС	2

Задача №352

Дробь

(Время: 1 сек. Память: 16 Мб Сложность: 39%)

Вася учится в третьем классе и сейчас он проходит тему «Простые дроби с натуральными числителем и знаменателем». Оказывается, что дробь называется правильной, если ее числитель меньше знаменателя, и несократимой, если числитель и знаменатель являются взаимно простыми. Вася очень любит математику и поэтому дома он решает много задач. В данный момент Вася ищет наибольшую правильную несократимую дробь, у которой сумма числителя и знаменателя равна N.

Требуется написать программу, которая поможет Васе решить эту задачу.

Входные данные

Входной файл INPUT.TXT содержит одно целое число N ($3 \leq N \leq 2 \cdot 10^9$).

Выходные данные

Выходной файл OUTPUT.TXT должен содержать два числа – числитель и знаменатель найденной дроби, разделенные пробелом.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	3	1 2
2	10	3 7

Задача №353

Треугольники

(Время: 1 сек. Память: 16 Мб Сложность: 41%)

Дан набор из нескольких отрезков. Необходимо составить треугольник наибольшей площади, используя в качестве сторон три отрезка из заданных.

Требуется написать программу, которая найдет наибольшую площадь треугольника.

Входные данные

Входной файл INPUT.TXT содержит в первой строке одно целое число N ($3 \leq N \leq 1000$) – количество отрезков. Во второй строке содержатся N целых чисел от 1 до 1000 – длины отрезков. Числа разделены пробелом.

Выходные данные

Выходной файл OUTPUT.TXT должен содержать одно число с тремя десятичными знаками после запятой – наибольшую площадь треугольника из заданных отрезков. Если из заданных отрезков нельзя построить ни одного треугольника, то вывести в файл 0.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	5 2 4 8 16 7	13.998
2	3 3 4 5	6.000
3	3 1 2 5	0

Задача №354

Разложение на простые множители

(Время: 1 сек. Память: 16 Мб Сложность: 27%)

Требуется вывести представление целого числа N в виде произведения простых чисел.

Входные данные

Входной файл INPUT.TXT содержит натуральное число N ($2 \leq N \leq 2^{31}-1$).

Выходные данные

В выходной файл OUTPUT.TXT выведите список простых множителей числа N в порядке неубывания, разделенных знаком «*».

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	5	5
2	30	2*3*5

Задача №355

Перестановки - 2

(Время: 1 сек. Память: 16 Мб Сложность: 46%)

Дана строка, состоящая из N символов. Требуется вывести все перестановки символов данной строки в лексикографическом порядке без повторений.

Входные данные

Входной файл INPUT.TXT содержит строку из N символов ($1 \leq N \leq 8$).

Выходные данные

В выходной файл OUTPUT.TXT выведите все элементы искомой последовательности по одному в каждой строке.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	AB	AB BA

	122	122
2		212
		221

Задача №356

Копилка

(Время: 1 сек. Память: 16 Мб Сложность: 49%)

Задан вес E пустой копилки и вес F копилки с монетами. В копилке могут находиться монеты N видов, для каждого вида известна ценность P_i и вес W_i одной монеты. Найти минимальную и максимальную суммы денег, которые могут находиться в копилке.

Входные данные

В первой строке входного файла INPUT.TXT находятся числа E и F , во второй - число N , в следующих N строках - по два числа, P_i и W_i . ($1 \leq E \leq F \leq 10000$, $1 \leq N \leq 500$, $1 \leq P_i \leq 50000$, $1 \leq W_i \leq 10000$).

Выходные данные

В выходной файл OUTPUT.TXT выведите два числа через пробел - минимальную и максимальную суммы. Если копилка не может иметь точно заданный вес при условии, что она наполнена монетами заданных видов, следует вывести "This is impossible".

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	1000 1100 2 1 1 5 2	100 250
2	1000 1010 2 6 3 2 2	10 16
3	1000 2000 1 10 3	This is impossible.

Задача №357

Делимость на 11

(Время: 1 сек. Память: 16 Мб Сложность: 22%)

Для делимости числа на 11 необходимо, чтобы разность между суммой цифр, стоящих на четных местах, и суммой цифр, стоящих на нечетных местах, делилась на 11.

Требуется написать программу, которая проверит делимость заданного числа на 11.

Входные данные

Входной файл INPUT.TXT содержит одно натуральное число N , делимость которого надо проверить ($1 \leq N \leq 10^{10000}$).

Выходные данные

В выходной файл OUTPUT.TXT выведите “YES”, если число делится на 11, или “NO” иначе.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	121	YES
2	1211	NO

Задача №358 Забор в парке

(Время: 1 сек. Память: 16 Мб Сложность: 42%)

В бесконечном парке деревья образуют квадратную решетку с шагом один метр. Часть парка было решено оградить забором, который представляет собой треугольник с заданными координатами вершин. Деревья, которые в точности попадают на вершины или стороны треугольника, придется срубить.

Требуется написать программу, которая найдет количество таких деревьев.

Входные данные

Входной файл INPUT.TXT содержит шесть целых чисел – координаты вершин треугольника (абсцисса, ордината). Все числа по абсолютной величине не превышают 10^9 и разделены пробелами.

Выходные данные

Выходной файл OUTPUT.TXT должен содержать одно число – количество деревьев.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	0 0 2 0 0 2	6
2	0 0 -1 -1 -1 0	3

Задача №359 Змейка - 2

(Время: 1 сек. Память: 16 Мб Сложность: 38%)

Мальчик Вася на уроке математики, вместо того, чтобы слушать учителя, рисовал числа в тетрадке в клеточку. Да не просто так рисовал, а определенным образом. Сначала он поставил в клетку число 1. Затем справа от нее нарисовал число 2. Затем снизу от числа 2 написал число 3. Затем перешёл на клетку правее и продолжил увлекательное занятие двигаясь по столбцу вверх, пока число в этом столбце не стало выше самого верхнего числа в предыдущем столбце. Затем он перешёл на клетку правее и опять таки продолжил рисование чисел, начиная с 7, но только уже сверху вниз, пока не нарисовал число, которое оказалось на одну клетку ниже самого нижнего числа в предыдущем столбце. И так далее. Вася не любил числа, заканчивающиеся нулем, и пропускал их при рисовании змейки. Первые его шесть заполненных столбцов мы скопировали из его тетрадки и привели здесь на рисунке. Так как Вася очень любопытный, то он очень хочет узнать, какое же число будет у него стоять в N-ом столбце в той строке, где стоит число 1. Первые 6 таких чисел в этой строке видны на рисунке: 1, 2, 5, 8, 14, 19.

				16	17
		6	7	15	18
1	2	5	8	14	19
	3	4	9	13	21
			11	12	22
					23

Требуется написать программу, которая поможет Васе.

Входные данные

Входной файл INPUT.TXT содержит одно число N ($1 \leq N \leq 10^6$) – номер столбца.

Выходные данные

Выходной файл OUTPUT.TXT должен содержать N-ое число в строке, где стоит число 1.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	3	5
2	5	14

Задача №360 Максимальная тройка

(Время: 2 сек. Память: 16 Мб Сложность: 33%)

В данной двумерной целочисленной таблице размером $N \times N$ требуется найти три элемента, сумма которых максимальна. При этом первый элемент должен быть соседним по горизонтали или вертикали со вторым, а второй — с третьим.

Входные данные

Входной файл INPUT.TXT содержит в первой строке число N ($1 < N \leq 2000$). В следующих N строках записано по N чисел – элементы таблицы. Элементы матрицы по абсолютной величине не превышают 100.

Выходные данные

Выходной файл OUTPUT.TXT должен содержать единственное число — максимальную сумму.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	3 1 1 1 2 2 1 2 1 0	6

Задача №361

Подстроки из одинаковых букв

(Время: 1 сек. Память: 16 Мб Сложность: 52%)

В заданной строке, состоящей из малых латинских букв, необходимо найти пару самых длинных подстрок, состоящих из одних и тех же букв (возможно, в разном порядке). Например, в строке twotwow это будут подстроки wotwo и otwow.

Входные данные

Входной файл INPUT.TXT содержит исходную строку, длиной от 1 до 100 символов.

Выходные данные

Выходной файл OUTPUT.TXT должен содержать единственное число – длину подстрок в максимальной паре, или 0, если таких подстрок в строке нет.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	abcde	0
2	abcdea	5

Задача №362

Открытка и конверт

(Время: 1 сек. Память: 16 Мб Сложность: 50%)

Даны размеры прямоугольных открытки и конверта. Требуется определить, поместится ли открытка в конверте.

Входные данные

Входной файл INPUT.TXT содержит в первой строке размеры открытки, во второй строке заданы размеры конверта. Все размеры – натуральные числа, не превосходящие 100.

Выходные данные

В выходной файл OUTPUT.TXT выведите «Possible», если открытку можно разместить в конверте и «Impossible» в противном случае.

Примеры

№	INPUT.TXT	OUTPUT.TXT

1	1 10 9 9	Possible
2	10 15 15 10	Possible
3	3 4 3 3	Impossible

Задача №363

Длинное произведение

(Время: 1 сек. Память: 16 Мб Сложность: 46%)

Даны целые неотрицательные числа М и N. Требуется найти произведение этих чисел.

Входные данные

Входной файл INPUT.TXT содержит в первой строке число М, а во второй строке – число N. (0 <= М, N <= 10²⁵⁰⁰)

Выходные данные

В выходной файл OUTPUT.TXT выведите произведение чисел М и N.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	5 7	35
2	4134937827592 784	3241791256832128
3	9876543210 1023456789	10108215200126352690

Задача №364

Совершенные числа

(Время: 1 сек. Память: 16 Мб Сложность: 51%)

Число называется совершенным, если оно равно сумме всех своих делителей, меньших его самого. Требуется найти все совершенные числа от М до N.

Входные данные

Входной файл INPUT.TXT содержит числа М и N, разделенные пробелом. (1 <= М <= N <= 5*10¹⁸)

Выходные данные

В выходной файл OUTPUT.TXT выведите по одному числу в строке в порядке возрастания все совершенные числа, находящихся на отрезке [М, N]. В том случае, когда таких чисел нет следует вывести «Absent».

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	6 6	6
2	4 5	Absent
3	5 30	6 28

Задача №365

Разложение на слагаемые

(Время: 1 сек. Память: 16 Мб Сложность: 54%)

Требуется вывести все различные представления натурального числа N в виде суммы натуральных чисел. Представления, отличающиеся друг от друга порядком слагаемых, не являются различными.

Входные данные

Входной файл INPUT.TXT содержит целое число N ($2 \leq N \leq 40$).

Выходные данные

В выходной файл OUTPUT.TXT выведите все различные представления числа N без повторов в виде суммы по одному на отдельной строке. Как слагаемые, так и сами суммы могут следовать в произвольном порядке.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	4	1+1+1+1 1+2+1 1+3 2+2
2	5	1+1+1+1+1 1+1+1+2 1+1+3 1+2+2 2+3 1+4

Задача №366

Выражение

(Время: 1 сек. Память: 16 Мб Сложность: 56%)

Даны N целых чисел X_1, X_2, \dots, X_N . Требуется расставить между ними знаки «+» и «-» так, чтобы значение получившегося выражения было равно заданному целому S.

Входные данные

Входной файл INPUT.TXT в первой строке содержит числа N и S. В следующей строке располагается N чисел, разделенных пробелом. Ограничения: $2 \leq N \leq 24, 0 \leq X_i \leq 5 \cdot 10^7, -10^9 \leq S \leq 10^9$.

Выходные данные

В выходной файл OUTPUT.TXT выведите «No solution», если такой результат получить невозможно, иначе выведите получившееся равенство. Если решение не единственное, выведите любое.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	3 10 15 25 30	15+25-30=10
2	2 100 10 10	No solution

Задача №367 Степень - 2

(Время: 1 сек. Память: 16 Мб Сложность: 45%)

Для натуральных чисел A и B требуется вычислить значение A^B .

Входные данные

Входной файл INPUT.TXT в первой строке содержит числа A и B, разделенные пробелом. ($1 \leq A \leq 9, 1 \leq B \leq 10^4$)

Выходные данные

В выходной файл OUTPUT.TXT выведите одно число – результат возведения в степень, без лидирующих нулей.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	3 20	3486784401
2	2 16	65536
3	5 50	88817841970012523233890533447265625

Задача №368 Маршрут

(Время: 1 сек. Память: 16 Мб Сложность: 38%)

В таблице из N строк и N столбцов клетки заполнены цифрами от 0 до 9. Требуется найти такой путь из клетки (1, 1) в клетку (N, N), чтобы сумма цифр в клетках, через которые он пролегает, была минимальной; из любой клетки ходить можно только вниз или вправо.

Входные данные

В первой строке входного файла INPUT.TXT находится число N. В следующих N строках содержатся по N цифр без пробелов. ($2 \leq N \leq 250$)

Выходные данные

В выходной файл OUTPUT.TXT выведите N строк по N символов. Символ «#» (решетка) показывает, что маршрут проходит через эту клетку, а «.» (точка) - что не проходит. Если путей с минимальной суммой цифр несколько, можно вывести любой.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	3 943 216 091	#.. ### ..#

Задача №369 Гангстеры

(Время: 1 сек. Память: 16 Мб Сложность: 44%)

N гангстеров собираются в ресторан. i-й гангстер приходит в момент времени T_i и имеет богатство P_i . Дверь ресторана имеет K+1 степень открытости, они обозначаются целыми числами из интервала $[0, K]$. Степень открытости двери может изменяться на единицу в единицу времени, то есть дверь может открыться на единицу, закрыться на единицу или остаться в том же состоянии. В начальный момент времени дверь закрыта (степень открытости 0). i-й гангстер заходит в ресторан, только если дверь открыта специально для него, то есть когда степень открытости двери соответствует его полноте S_i . Если в момент, когда гангстер подходит к ресторану, степень открытости двери не соответствует его полноте, он уходит и больше не возвращается. Ресторан работает в интервале времени $[0, T]$.

Требуется собрать гангстеров с максимальным суммарным богатством в ресторане, открывая и закрывая дверь соответствующим образом.

Входные данные

В первой строке входного файла INPUT.TXT находятся числа N, K, T, во второй - T_1, T_2, \dots, T_N , в третьей - P_1, P_2, \dots, P_N . в четвёртой - S_1, S_2, \dots, S_N . Числа в строках разделены пробелами. Ограничения: $1 \leq N \leq 100$, $1 \leq K \leq 100$, $1 \leq T \leq 30\,000$, $0 \leq T_i \leq T$, $1 \leq P_i \leq 300$, $1 \leq S_i \leq K$.

Выходные данные

В выходной файл OUTPUT.TXT выведите одно число - максимальное суммарное богатство гангстеров, попавших в ресторан. Если зайти не удалось никому, вывести 0.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	4 10 20 10 16 8 16 10 11 15 1 10 7 1 8	26
2	2 17 100 5 0 50 33 6 1	0

Задача №370

Площадь многоугольника

(Время: 1 сек. Память: 16 Мб Сложность: 48%)

Многоугольник на плоскости задан целочисленными координатами своих N вершин в декартовой системе координат. Требуется найти площадь многоугольника. Стороны многоугольника не соприкасаются (за исключением соседних - в вершинах) и не пересекаются.

Входные данные

В первой строке входного файла INPUT.TXT находится число N . В следующих N строках находятся пары чисел (X_i, Y_i) - координаты точек. Если соединить точки в данном порядке, а также первую и последнюю точки, получится заданный многоугольник. ($3 \leq N \leq 50\,000$, $-20\,000 \leq X_i, Y_i \leq 20\,000$)

Выходные данные

В выходной файл OUTPUT.TXT выведите одно число - площадь многоугольника. Его следует округлить до ближайшего числа с одной цифрой после запятой.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	4 5 0 0 5 -5 0 0 -5	50.0
2	4 0 4 0 0 3 0 1 1	3.5

Задача №371

Дружественные числа

(Время: 1 сек. Память: 16 Мб Сложность: 58%)

Два различных натуральных числа называются дружественными, если первое из них равно сумме делителей второго числа, за исключением самого второго числа, а второе равно сумме делителей первого числа, за исключением самого первого числа. Требуется найти все пары дружественных чисел, оба из которых принадлежат промежутку от M до N .

Входные данные

Входной файл INPUT.TXT в первой строке содержит натуральные числа M и N , разделенные пробелом, не превышающие 10^6 .

Выходные данные

В выходной файл OUTPUT.TXT выведите в каждой строке по паре чисел через пробел. Первое

число пары должно быть меньше второго. Строки должны быть отсортированы в порядке возрастания первого числа пары. Если пар дружественных чисел в промежутке нет, то следует вывести "Absent".

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	200 300	220 284
2	200 250	Absent
3	185000 205000	185368 203432 196724 202444

Комментарий к примеру №1

220=1+2+4+71+142 (все делители числа 284)
284=1+2+4+5+10+11+20+22+44+55+110 (все делители числа 220)

Задача №372 Скобки - 2

(Время: 1 сек. Память: 16 Мб Сложность: 52%)

Вывести все правильные скобочные выражения длиной N, состоящие из круглых и квадратных скобок.

Входные данные

Входной файл INPUT.TXT содержит единственное четное натуральное число N, не превышающее 14.

Выходные данные

В выходной файл OUTPUT.TXT выведите всевозможные правильные скобочатые выражения по одному в каждой строке в произвольном порядке.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	2	() []
2	4	() () () [] (()) ([]) [] () [] [] [()] [[]]

Задача №373 Маршрут - 2

(Время: 1 сек. Память: 16 Мб Сложность: 47%)

Дана матрица $N \times N$, заполненная положительными числами. Путь по матрице начинается в левом верхнем углу. За один ход можно пройти в соседнюю по вертикали или горизонтали клетку (если она существует). Нельзя ходить по диагонали, нельзя оставаться на месте.

Требуется найти максимальную сумму чисел, стоящих в клетках по пути длиной K (клетку можно посещать несколько раз).

Входные данные

Входной файл INPUT.TXT в первой строке содержит разделенные пробелом числа N и K . Затем идут N строк по N чисел в каждой – данные таблицы. Элементы матрицы – целые числа от 1 до 9999, $2 \leq N \leq 100$, $1 \leq K \leq 2000$.

Выходные данные

В выходной файл OUTPUT.TXT выведите одно число – максимальную сумму.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	5 7 1 1 1 1 1 1 1 3 1 9 1 1 6 1 1 1 1 3 1 1 1 1 1 1 1	21

Задача №374 Выпуклая оболочка - 2

(Время: 1 сек. Память: 16 Мб Сложность: 55%)

На плоскости заданы N точек своими декартовыми координатами. Найти минимальный периметр многоугольника, содержащего все эти точки. Гарантируется, что искомый многоугольник имеет ненулевую площадь.

Входные данные

Входной файл INPUT.TXT в первой строке содержит число N , далее - N строк с парами координат (x_i, y_i) . Ограничения: $3 \leq N \leq 1000$, $-10\,000 \leq x_i, y_i \leq 10\,000$, все числа целые, все точки различны.

Выходные данные

В выходной файл OUTPUT.TXT выведите одно вещественное число - длину периметра полученного многоугольника с точностью не менее 10^{-2} .

Пример

№	INPUT.TXT	OUTPUT.TXT
1	5 1 0 0 1 -1 0	5.66

0	-1
0	0

Задача №375

Системы счисления

(Время: 1 сек. Память: 16 Мб Сложность: 53%)

Дано целое неотрицательное число в m-ричной системе счисления. Требуется вывести это число в k-ричной системе счисления.

Входные данные

Входной файл INPUT.TXT в первой строке содержит два числа m и k (в десятичной системе счисления), во второй строке - число для перевода. Ограничения: $2 \leq m, k \leq 36$, для представления цифр 10...35 используются прописные латинские буквы A...Z соответственно, число разрядов исходного числа не превышает 1000.

Выходные данные

В выходной файл OUTPUT.TXT выведите искомое число без лидирующих нулей.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	10 36 29234652	HELLO
2	2 10 1101	13

Задача №376

День рождения - 2

(Время: 1 сек. Память: 16 Мб Сложность: 32%)

Заданы день и месяц рождения, а также текущие день, месяц и год. Определить, сколько дней осталось до дня рождения.

Примечание. Високосные годы - это те, номер которых делится на 400, а также те, номер которых делится на 4, но не делится на 100.

Входные данные

В первой строке входного файла INPUT.TXT находятся, разделённые пробелами, день и месяц рождения, во второй - разделённые пробелами текущие день, месяц и год. Ограничения: год от 1920 до 3000, месяц от 1 до 12, день от 1 до числа дней в месяце.

Выходные данные

В выходной файл OUTPUT.TXT выведите число дней, оставшихся до дня рождения.

Примеры

№	INPUT.TXT	OUTPUT.TXT

1	19 04 19 04 2002	0
2	05 05 19 04 2002	16
3	29 02 28 02 2001	1096

Задача №377

Закраска прямой

(Время: 1 сек. Память: 16 Мб Сложность: 48%)

На числовой прямой окрасили N отрезков. Известны координаты левого и правого концов каждого отрезка (Li и Ri). Найти длину окрашенной части числовой прямой.

Входные данные

Входной файл INPUT.TXT в первой строке содержит число N, в следующих N строках - пары Li и Ri. Ограничения: все числа целые, не превышающие 10⁹ по абсолютной величине, 1 <= N <= 15 000.

Выходные данные

В выходной файл OUTPUT.TXT выведите длину окрашенной части прямой.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	2 1 3 2 4	3
2	1 10 10	0

Задача №378

Суммы

(Время: 1 сек. Память: 16 Мб Сложность: 62%)

Дано N целых чисел A₁, A₂, ..., A_N. Требуется найти количество различных значений сумм вида k₁A₁ + k₂A₂ + ... + k_NA_N.

Входные данные

Входной файл INPUT.TXT в первой строке содержит число N, во второй - A₁, A₂, ..., A_N через пробел. Ограничения: все числа целые, 1 <= N <= 500, 0 <= A_i <=100, 0 <= k_i <= 1.

Выходные данные

В выходной файл OUTPUT.TXT выведите количество различных значений сумм.

Примеры

--	--	--

№	INPUT.TXT	OUTPUT.TXT
1	3 1 1 2	5
2	3 1 3 2	7
3	5 49 100 98 49 0	10

Задача №379

Игра с датой

(Время: 1 сек. Память: 16 Мб Сложность: 54%)

Играют двое. Задаётся какая-то дата 2008 года. Каждый игрок на своём ходе называет более позднюю дату, увеличивая на 1 или 2 либо день в месяце, либо месяц, но не то и другое сразу. При этом сочетание дня и месяца должно оставаться датой. Игрок, назвавший 31 декабря, проигрывает. Оба играют наилучшим образом. Исходя из заданной даты вывести, кто выиграет.

Входные данные

В первой строке входного файла INPUT.TXT находятся числа, обозначающие день и месяц. Месяц указывается от 1 до 12, день от 1 до числа дней в месяце, дата «31 декабря» отсутствует во входных данных.

Выходные данные

В выходной файл OUTPUT.TXT выведите 1, если выигрывает первый (начинающий) игрок, или 2 – в противном случае.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	30 12	2
2	29 12	1
3	29 11	2

Задача №380

Площадь прямоугольников

(Время: 1 сек. Память: 16 Мб Сложность: 50%)

Дано N прямоугольников со сторонами, параллельными осям координат. Требуется определить площадь фигуры, образованной объединением данных прямоугольников.

Входные данные

В первой строке входного файла INPUT.TXT находится число прямоугольников - N. Затем идут N строк, содержащих по 4 числа: x1, y1, x2, y2 - координаты двух противоположных углов прямоугольника. Все координаты – целые числа, не превосходящие по абсолютной величине 10 000. (1 <= N <= 100)

Выходные данные

В выходной файл OUTPUT.TXT выведите одно целое число – площадь фигуры.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	1 1 1 7 7	36
2	2 1 1 3 3 2 2 4 4	7

Задача №381 Lines

(Время: 1 сек. Память: 16 Мб Сложность: 44%)

В таблице из N строк и N столбцов некоторые клетки заняты шариками, другие свободны. Выбран шарик, который нужно переместить, и место, куда его нужно переместить. Выбранный шарик за один шаг перемещается в соседнюю по горизонтали или вертикали свободную клетку. Требуется выяснить, возможно ли переместить шарик из начальной клетки в заданную, и, если возможно, то найти путь из наименьшего количества шагов.

Входные данные

В первой строке входного файла INPUT.TXT находится число N, в следующих N строках - по N символов. Символом точки обозначена свободная клетка, латинской заглавной O - шарик, @ - исходное положение шарика, который должен двигаться, латинской заглавной X - конечное положение шарика. (2 <= N <= 50)

Выходные данные

В выходной файл OUTPUT.TXT выведите в первой строке Yes, если движение возможно, или No, если нет. Если движение возможно, то далее следует вывести N строк по N символов - как и на вводе, но букву X, а также все точки по пути следует заменить плюсами. Если решений несколько, выведите любое.

Примеры

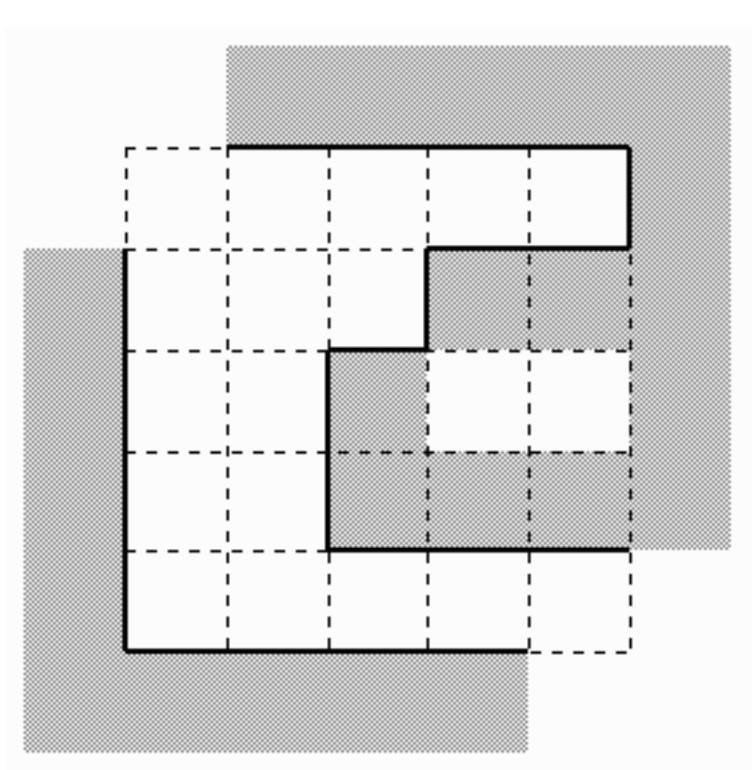
№	INPUT.TXT	OUTPUT.TXT
1	5X .O.O.O.O O.O.O.O. @.....	Yes +++++ +O.O.O.O +++++ O.O.O.O+ @++++
2	5 ..X.. O.O.O.O.O@..	No

Задача №382

Покраска лабиринта

(Время: 1 сек. Память: 16 Мб Сложность: 46%)

Лабиринт представляет собой квадрат, состоящий из $N \times N$ сегментов. Каждый из сегментов может быть либо пустым, либо заполненным монолитной каменной стеной. Гарантируется, что левый верхний и правый нижний сегменты пусты. Лабиринт обнесён сверху, снизу, слева и справа стенами, оставляющими свободными только левый верхний и правый нижний углы. Директор лабиринта решил покрасить стены лабиринта, видимые изнутри (см. рисунок). Помогите ему рассчитать количество краски, необходимой для этого.



Входные данные

В первой строке входного файла INPUT.TXT находится число N , затем идут N строк по N символов: точка обозначает пустой сегмент, решётка - сегмент со стеной. Размер сегментов – 5×5 метров, высота стен – 5 метров. ($3 \leq N \leq 50$)

Выходные данные

В выходной файл OUTPUT.TXT выведите одно число - площадь видимой части внутренних стен лабиринта в квадратных метрах.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	5## ..#.. ..###	550

Задача №383

Красивые числа - 2

(Время: 1 сек. Память: 16 Мб Сложность: 26%)

Будем называть число красивым, если сумма его цифр в десятичной системе счисления делится на количество цифр в нем (в десятичной системе счисления).

Необходимо найти N -ое в порядке возрастания красивое число.

Входные данные

Входной файл INPUT.TXT содержит целое число N . ($1 \leq N \leq 100\,000$)

Выходные данные

В выходной файл OUTPUT.TXT выведите ответ на задачу.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	1	1
2	15	20

Задача №384 Числа Фибоначчи - 3

(Время: 1 сек. Память: 16 Мб Сложность: 52%)

Последовательностью Фибоначчи называется последовательность чисел $F_0 = 0, F_1 = 1, \dots, F_k = F_{k-1} + F_{k-2}$ ($k > 1$).

Требуется найти наибольший общий делитель двух чисел Фибоначчи.

Входные данные

Во входном файле INPUT.TXT записаны два целых числа i и j ($1 \leq i, j \leq 10^6$).

Выходные данные

В выходной файл OUTPUT.TXT выведите остаток от деления НОД чисел F_i и F_j на 10^9 .

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	5 10	5
2	2 4	1

Задача №385 Развлечения с измерителем

(Время: 1 сек. Память: 16 Мб Сложность: 33%)

Дима обнаружил у папы на столе специальный чертежный прибор, похожий на циркуль-измеритель. Измеритель отличается от обычного циркуля тем, что в обеих его ножках находятся иголки (у обычного циркуля в одной ножке находится иголка, а в другой – грифель).

Кроме измерителя Дима нашел на столе клетчатый лист бумаги, в углах некоторых клеток которого были нарисованы точки. Так как измеритель служит для измерения расстояний, то Дима решил измерить все попарные расстояния между всеми точками на листе бумаги.

Ваша задача - написать программу, которая по координатам точек определит, сколько различных расстояний встречается среди расстояний, которые измерил Дима.

Входные данные

Первая строка входного файла INPUT.TXT содержит число n – количество точек ($2 \leq n \leq 50$). Следующие n строк содержат по два целых числа – координаты точек. Координаты не превышают 10^4 по абсолютной величине.

Выходные данные

На первой строке выходного файла OUTPUT.TXT выведите k – количество различных расстояний, которые измерил Дима. Следующие k строк должны содержать по одному вещественному числу – сами расстояния. Расстояния должны быть выведены в возрастающем порядке. Каждое число должно быть выведено с точностью не менее чем 10^{-9} .

Пример

№	INPUT.TXT	OUTPUT.TXT
1	4 0 0 1 1 1 0 0 1	2 1.0 1.414213562373

Задача №386 Генерация тестов

(Время: 1 сек. Память: 16 Мб Сложность: 37%)

При подготовке задач для олимпиад по информатике и программированию часто возникает необходимость подготовки тестов. Поскольку зачастую количество тестов достаточно велико, и каждый из них может содержать множество данных, то генерацию тестов разумно автоматизировать.

В геометрических задачах часто требуется сгенерировать n точек на плоскости так, чтобы никакие три из них не лежали на одной прямой. В этом и состоит ваша задача. Напишите программу, которая по числу N построит множество из N точек, обладающее указанным свойством.

Входные данные

Входной файл INPUT.TXT содержит целое число N ($1 \leq N \leq 300$).

Выходные данные

Если искомое множество точек можно построить, то выведите в выходной файл OUTPUT.TXT в первой строке слово YES, а далее N строк, каждая из которых должна содержать два числа – координаты соответствующей точки. Среди точек не должно быть совпадающих. Все координаты должны быть целыми числами, не превосходящими 10000 по абсолютному значению. Если искомое множество точек нельзя построить, выведите в выходной файл строку NO.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	1	YES 0 0
2	4	YES 0 0 0 1 1 0 1 1

Задача №387

Левая рекурсия

(Время: 1 сек. Память: 16 Мб Сложность: 20%)

В теории формальных грамматик и автоматов (ТФГиА) важную роль играют так называемые *контекстно-свободные грамматики* (КС-грамматики). КС-грамматикой будем называть четверку, состоящую из множества N нетерминальных символов, множества T терминальных символов, множества P правил (продукций) и начального символа S , принадлежащего множеству N .

Каждая продукция p из P имеет форму $A \rightarrow a$, где A нетерминальный символ (A из N), а a – строка, состоящая из терминальных и нетерминальных символов. Процесс вывода слова начинается со строки, содержащей только начальный символ S . После этого на каждом шаге один из нетерминальных символов, входящих в текущую строку, заменяется на правую часть одной из продукций, в которой он является левой частью. Если после такой операции получается строка, содержащая только терминальные символы, то процесс вывода заканчивается.

Во многих теоретических задачах удобно рассматривать так называемые *нормальные формы* грамматик. Процесс приведения грамматики к нормальной форме часто начинается с устранения левой рекурсии. В этой задаче мы будем рассматривать только ее частный случай, называемый непосредственной левой рекурсией. Говорят, что правило вывода $A \rightarrow R$ содержит непосредственную левую рекурсию, если первым символом строки R является A .

Задана КС-грамматика. Требуется найти количество правил, содержащих непосредственную левую рекурсию.

Входные данные

Первая строка входного файла INPUT.TXT содержит количество n ($1 \leq n \leq 1000$) правил в грамматике. Каждая из последующих n строк содержит по одному правилу. Нетерминальные символы обозначаются заглавными буквами латинского алфавита, терминальные – строчными. Левая часть продукции отделяется от правой символами \rightarrow . Правая часть продукции имеет длину от 1 до 30 символов.

Выходные данные

В выходной файл OUTPUT.TXT выведите ответ на задачу.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	3 S→Sabc S→A A→AA	2

Задача №388

Седловые точки

(Время: 1 сек. Память: 16 Мб Сложность: 28%)

Задана матрица, содержащая N строк и M столбцов. Седловой точкой этой матрицы назовем элемент, который одновременно является минимумом в своей строке и максимумом в своем столбце.

Найдите количество седловых точек заданной матрицы.

Входные данные

Входной файл INPUT.TXT в первой строке содержит целые числа N и M ($1 \leq N, M \leq 750$). Далее следуют N строк по M чисел в каждой. Элементы матрицы - целые числа, не превосходящие 1000 по абсолютной величине.

Выходные данные

В выходной файл OUTPUT.TXT выведите ответ на задачу.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	2 2 0 0 0 0	4
2	2 2 1 2 3 4	1

Задача №389 К коду Грея

(Время: 1 сек. Память: 16 Мб Сложность: 50%)

Рассмотрим циклическую последовательность попарно различных чисел $\{a_0, a_1, \dots, a_{2^n-1}\}$, $0 \leq a_i \leq 2^n-1$. Назовем эту последовательность кодом Грея, если любой a_i отличается от левого соседа a_{i-1} и правого соседа a_{i+1} только в одной цифре в двоичной записи этих чисел. Для a_0 левым соседом считается a_{2^n-1} , а для a_{2^n-1} правым соседом считается a_0 .

Вася хочет запрограммировать игру-головоломку, которая будет позволять пользователю менять местами два любых числа a_i и a_j . Задача игрока – получить код Грея. Модуль, отвечающий за перестановку чисел, Вася берет на себя. А вот Ваша задача – написать программу, которая будет определять после каждой перестановки – является ли последовательность кодом Грея.

Входные данные

В первой строке входного файла INPUT.TXT содержится число n. В следующей строке перечислены попарно различные числа a_i . В третьей строке записано число m – количество перестановок, сделанных пользователем. В следующих m строках перечислены числа (i, j) – индексы переставляемых элементов. Ограничения: $1 \leq n \leq 16$; $1 \leq m \leq 10^5$; $i \neq j$, $0 \leq i, j < 2^n$.

Выходные данные

В выходной файл OUTPUT.TXT запишите m строчек – в i-той строке запишите «Yes», если после i-той перестановки последовательность стала кодом Грея и «No» в противном случае.

Пример

№	INPUT.TXT	OUTPUT.TXT
	2 0 1 3 2	No Yes

1	2
	1 2
	2 1

Задача №390

Треугольная область

(Время: 1 сек. Память: 16 Мб Сложность: 41%)

Одним из субъектов Флатландии является Треугольная область. Как следует из ее названия, она имеет форму треугольника, вершины которого находятся в точках с координатами (x_1,y_1) , (x_2,y_2) , (x_3,y_3) . Административный центр этой области находится в точке с координатами (x_c,y_c) , которая лежит строго внутри указанного треугольника.

Для оценки транспортного и логистического потенциала области, ее руководству понадобилось узнать расстояние от административного центра области до ее границы. Напишите программу, которая вычислит это расстояние.

Входные данные

Первая строка входного файла INPUT.TXT содержит шесть целых чисел – $x_1, y_1, x_2, y_2, x_3, y_3$. Вторая строка входного файла содержит два целых числа – x_c и y_c . Все числа во входном файле не превосходят 10000 по абсолютной величине. В заданном треугольнике нет тупых углов.

Выходные данные

В выходной файл OUTPUT.TXT выведите искомое расстояние до границы с точностью не менее 10^{-6} .

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	0 0 1 2 2 0 1 1	0.4472135
2	0 0 0 3 3 0 1 1	0.7071067

Задача №391

Взлом хеш-функции

(Время: 1 сек. Память: 16 Мб Сложность: 35%)

В некоторых задачах защиты информации используются так называемые хеш-функции. Одним из важнейших классов хеш-функций являются так называемые полиномиальные хеш-функции.

Пусть дана строка $S = s_1s_2 \dots s_l$, состоящая из цифр от 0 до 9. Тогда значение полиномиальной хеш-функции $p(S, x, m)$ вычисляется следующим образом:

$$p(S, x, m) = \left(\sum_{i=1}^l s_i x^{i-1}\right) \bmod m$$

($a \bmod b$ обозначает остаток от деления числа a на число b). Например, пусть $S = 0123$, тогда $p(S, 2, 5) = (0 \cdot 1 + 1 \cdot 2 + 2 \cdot 4 + 3 \cdot 8) \bmod 5 = 4$.

Одним из способов применения хеш-функций является хранение паролей. Часто бывает так, что пароли приходится хранить в незащищенной таблице базы данных, поэтому вместо них хранят хеш-функции от них. При проверке пароля вычисляется хеш-функция от введенной строки и сравнивается со значением, хранящимся в таблице.

Ваша задача состоит в том, чтобы по заданным числам x , m , L и v найти строку S из цифр от 0 до 9 длины L , значение полиномиальной хеш-функции $p(S, x, m)$ равно v .

Входные данные

Входной файл INPUT.TXT содержит четыре целых числа: x (x – простое число, $5 \leq x \leq 100$), m (m является степенью двойки, $1 \leq m \leq 256$), L ($10 \leq L \leq 100$) и v ($0 \leq v \leq m-1$).

Выходные данные

В выходной файл OUTPUT.TXT выведите искомую строку или NO SOLUTION, если такой строки не существует. Если решений несколько, выведите любое.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	5 16 10 9	0422207956

Задача №392 Сдвиг перестановки

(Время: 1 сек. Память: 16 Мб Сложность: 24%)

Перестановкой порядка n называется последовательность из попарно различных целых положительных чисел p_1, p_2, \dots, p_n , где каждое $1 \leq p_i \leq n$. Будем говорить, что перестановка q_1, q_2, \dots, q_n лексикографически меньше перестановки p_1, p_2, \dots, p_n , если существует такое i , что $q_i < p_i$, а для любого $j < i$ $p_j = q_j$.

Циклическим сдвигом на k перестановки p_1, p_2, \dots, p_n называется последовательность, $p_{k+1}, p_{k+2}, \dots, p_n, p_1, \dots, p_k$. Отметим, что любой циклический сдвиг перестановки также является перестановкой.

Ваша задача состоит в том, чтобы найти наименьший лексикографически циклический сдвиг заданной перестановки.

Входные данные

Первая строка входного файла INPUT.TXT содержит порядок n ($1 \leq n \leq 10^5$) заданной перестановки. Вторая строка содержит числа p_1, p_2, \dots, p_n , отделенные друг от друга пробелами.

Выходные данные

В выходной файл OUTPUT.TXT выведите перестановку, являющуюся наименьшим лексикографически циклическим сдвигом перестановки, заданной во входном файле.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	3	1 3 2

Задача №393
Плейлист

(Время: 1 сек. Память: 16 Мб Сложность: 48%)

Антон – большой любитель музыки. Но в разные моменты ему нравится разная музыка. Однажды он решил составить рейтинг своих песен, причем считать его так – если песня дослушана до конца, то ее рейтинг увеличивается на единицу, если же он так и не дослушал ее, переключив на следующую, то уменьшается на единицу. Отметим, что вследствие этого рейтинг может стать отрицательным.

Он попросил Вас написать ему программу, которая подсчитывала бы такой рейтинг. Для этого он записал в каком порядке прослушивались песни и в какие моменты времени он переключал песню на следующую. Изначально рейтинги всех песен равны нулю.

Входные данные

Первая строка входного файла INPUT.TXT содержит одно число n ($1 \leq n \leq 1000$) – количество песен у Антона. В следующих n строках следуют описания песен – название песни, состоящее не более, чем из 50 маленьких латинских букв, и длина песни в секундах, разделенные пробелом. Каждая песня длится положительное число секунд и не более 30 минут. $(n + 2)$ -я строка содержит два числа m, k ($0 \leq k \leq m \leq 1000$) – количество прослушанных песен и количество переключений на следующую. В следующих m строках следуют названия прослушанных песен в порядке их прослушивания (все эти песни из списка). И, наконец, в последней строке – k целых неотрицательных чисел – последовательные (в неубывающем порядке) времена переключения на следующую песню (гарантируется, что она всегда существует), отсчитываемые в секундах от начала прослушивания первой. Считается, что если Антон переключил песню в момент окончания какой-нибудь, то он пропускает уже следующую.

Выходные данные

В выходной файл OUTPUT.TXT выведите n чисел – рейтинги песен в том порядке, в котором они даны во входном файле.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	2 songa 10 songb 20 5 2 songa songb songb songa songa 40 55	1 0

Задача №394
Апельсины

(Время: 1 сек. Память: 16 Мб Сложность: 31%)

Катя решила пригласить к себе в гости n друзей. Так как ее друзья очень любят фрукты, то в качестве угощения для них она купила m одинаковых апельсинов.

Она хочет разрезать каждый апельсин на одинаковое число равных долек так, чтобы их можно было распределить между гостями (сама Катя апельсины есть не будет), и всем гостям досталось поровну долек.

Напишите программу, которая вычисляет минимальное количество долек, на которое необходимо разрезать каждый апельсин, чтобы были выполнены указанные выше условия.

Входные данные

Входной файл INPUT.TXT содержит два положительных целых числа n и m ($1 \leq n, m \leq 10^9$).

Выходные данные

В выходной файл OUTPUT.TXT выведите ответ на задачу.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	2 5	2
2	2 4	1

Задача №395 Произведение цифр - 2

(Время: 1 сек. Память: 16 Мб Сложность: 28%)

Найдите количество чисел из отрезка $[l, r]$, которые делятся на произведение своих цифр.

Входные данные

Входной файл INPUT.TXT содержит два целых числа l и r. ($1 \leq l \leq r \leq 10^9, |r-l| \leq 10^5$)

Выходные данные

В выходной файл OUTPUT.TXT выведите ответ на задачу.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	1 12	11

Задача №396 Точки и отрезки

(Время: 2 сек. Память: 16 Мб Сложность: 62%)

Дано N отрезков на числовой прямой и M точек на этой же прямой. Для каждой из данных точек определите, скольким отрезкам она принадлежит. Точка x считается принадлежащей отрезку с концами a и b, если выполняется двойное неравенство $\min(a, b) \leq x \leq \max(a, b)$.

Входные данные

Первая строка входного файла INPUT.TXT содержит два целых числа N – число отрезков и M – число точек ($1 \leq N, M \leq 10^5$). В следующих N строках по два целых числа a_i и b_i – координаты концов соответствующего отрезка. В последней строке M целых чисел – координаты точек. Все числа во входном файле не превосходят по модулю 10^9 .

Выходные данные

В выходной файл OUTPUT.TXT выведите M чисел – для каждой точки количество отрезков, в которых она содержится.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	3 2 0 5 -3 2 7 10 1 6	2 0
2	1 3 -10 10 -100 100 0	0 0 1

Задача №397 Качество строки

(Время: 1 сек. Память: 16 Мб Сложность: 52%)

Назовем *качеством строки* разность между максимальным и минимальным номерами в алфавите букв, входящих в строку. Например, качество строки ab равно $2 - 1 = 1$, а строки $abcz$ равно $26 - 1 = 25$.

Дана строка S . Необходимо найти непустую подстроку этой строки, обладающую максимальным качеством, а из всех таких – минимальную по длине.

Входные данные

Входной файл INPUT.TXT содержит непустую строку S , состоящую из строчных букв латинского алфавита. Ее длина не превосходит $2 \cdot 10^5$ символов.

Выходные данные

В выходной файл OUTPUT.TXT выведите искомую подстроку. Если вариантов ответа несколько, выведите любой.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	aba	ab
2	zzz	z

Задача №398 Сумма - 2

(Время: 1 сек. Память: 16 Мб Сложность: 25%)

Задано натуральное число x . Найдите число способов представить его в виде суммы четырех натуральных чисел: $x = a + b + c + d$, где $a \leq b \leq c \leq d$.

Входные данные

Входной файл INPUT.TXT содержит целое число x ($1 \leq x \leq 1500$).

Выходные данные

В выходной файл OUTPUT.TXT выведите ответ на задачу.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	3	0
2	5	1

Задача №399 Жук

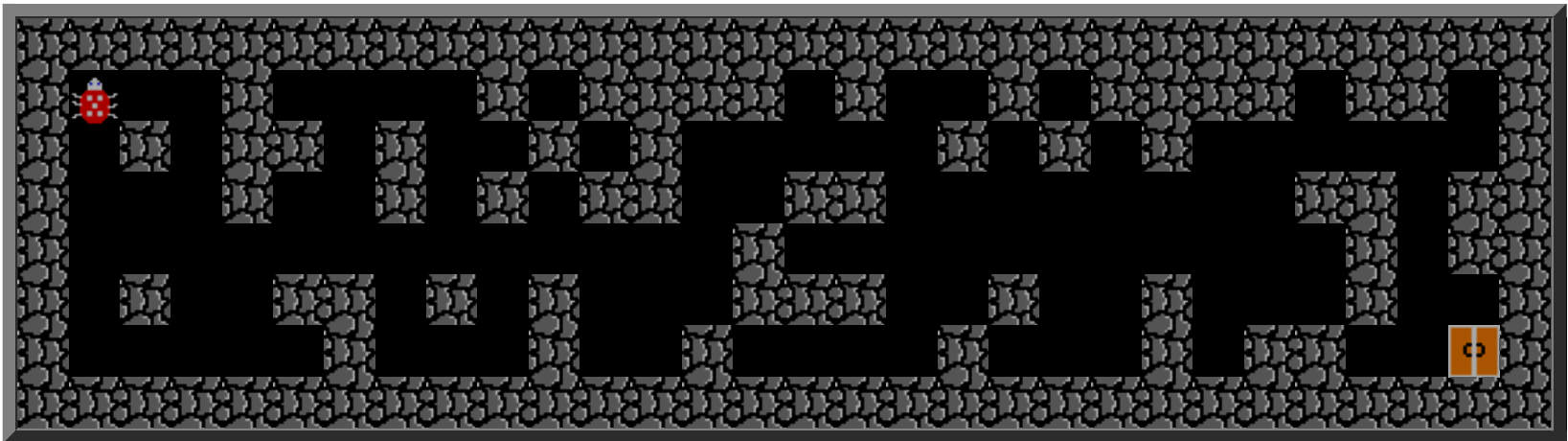
(Время: 1 сек. Память: 16 Мб Сложность: 30%)

Петя нашел в Интернете по адресу <http://buglab.ru> игру-головоломку "Жук", в которой от участников требуется построить для жука лабиринт таким образом, чтобы жук как можно дольше искал выход.

Жук всегда начинает свое движение с левого верхнего угла, а выход всегда находится в правом нижнем. Жук движется не оптимально, а следующим образом: он идет туда, где еще не был, либо был там реже. Т.е. проходя каждую клетку лабиринта, жук запоминает: сколько раз он был в этой клетке и при обдумывании направления своего движения в какой то конкретный момент он смотрит: сколько раз он был в клетке снизу, сколько справа, сколько слева и сколько сверху и движется туда, где он был меньше раз. Если таких направлений несколько и одно из них совпадает с текущим направлением движения, то он не меняет направления, иначе он движется согласно следующим приоритетам: вниз, направо, вверх, налево. Т.е. если минимальное число посещений сразу справа и слева (а двигался он при этом вверх или вниз), то жук идет направо, т.к. у "направо" приоритет выше. Следует заметить, что двигаясь по данному алгоритму жук всегда достигнет выхода в том случае, когда выход существует.

Изучив алгоритм движения жука Петя хочет написать программу, которая по заданному лабиринту определит количество перемещений жука прежде, чем он достигнет выхода. Помогите Пете с реализацией данной программы!

Конструктор лабиринта





Ходы: "0"

Входные данные

Входной файл INPUT.TXT в первой строке содержит разделенные пробелом целые числа N и M - количество строк и столбцов в лабиринте ($4 \leq N$, $M \leq 100$). Далее следует N строк, содержащих данные лабиринта построчно. Каждая строка содержит M символов - клетки лабиринта текущей строки, где символ "@" обозначает присутствие стены, а символ пробела - пустое пространство. Гарантируется, что граница лабиринта окружена стеной. Предполагается, что жук начинает свое движение из координаты (2, 2) и заканчивает в координате (M-1, N-1), подразумевается, что в этих координатах нет стен.

Выходные данные

В выходной файл OUTPUT.TXT выведите количество движений жука, если спасительный маршрут для жука существует, и -1 в противном случае.

Примеры

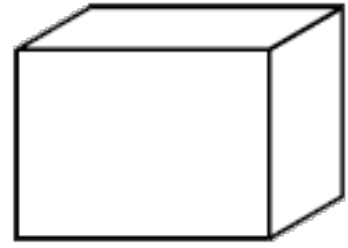
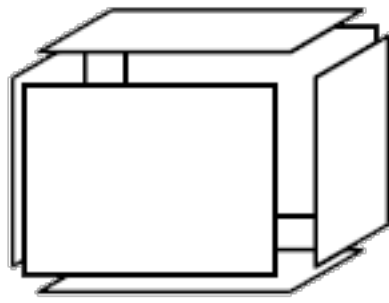
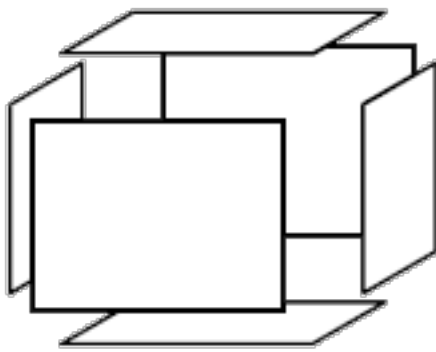
1	INPUT.TXT	OUTPUT.TXT
1	<pre> 6 6 @@@@@@ @ @ @ @ @ @ @@ @ @ @ @@@@@@ </pre>	20
2	<pre> 8 30 @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@ @ @ @ @@@@ @ @ @@@@ @@@ @ @ @ @@ @ @ @ @ @ @ @ @ @ @ @ @@ @@ @@ @@ @ @ @ @ @ @@ @ @ @@ @ @ @@@@ @ @ @ @ @ @ @ @ @ @ @@ @ @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@ </pre>	630
3	<pre> 4 4 @@@@ @ @@ @@ @ @@@@ </pre>	-1

Задача №400

Коробка

(Время: 1 сек. Память: 16 Мб Сложность: 37%)

Иван работает на заводе, который производит тяжелую технику. Его работа очень проста – он собирает коробки и упаковывает в них технику для заказчиков. Каждая такая коробка представляет собой параллелепипед. Для сборки коробки Иван использует шесть прямоугольных деревянных плиток. Каждая плита представляет собой одну из сторон коробки.



Петр подбирает плитки для Ивана. Петр недостаточно умен и поэтому часто допускает ошибки – он приносит Ивану такие плитки, из которых невозможно собрать коробку. Но Иван не доверяет Петру. Поэтому он всегда тратит массу времени на то, чтобы объяснить Петру то, где он допустил ошибку.

К счастью, Петр обожает все, что связано с компьютерами и верит в то, что компьютеры никогда не ошибаются. Иван решил, что можно использовать это в их работе. Иван попросил Вас написать программу, которая по заданным размерам шести плиток скажет: возможно ли построить из них коробку.

Входные данные

Входной файл INPUT.TXT содержит шесть строк, каждая из которых содержит два натуральных числа w и h ($1 \leq w, h \leq 10\,000$) – ширина и высота плитки в миллиметрах.

Выходные данные

В выходной файл OUTPUT.TXT выведите «POSSIBLE», если возможно собрать коробку из данных плит, и «IMPOSSIBLE» в противном случае.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	1345 2584 2584 683 2584 1345 683 1345 683 1345 2584 683	POSSIBLE
2	1234 4567 1234 4567 4567 4321 4322 4567 4321 1234 4321 1234	IMPOSSIBLE

Задача №401 Шары и коробки

(Время: 1 сек. Память: 16 Мб Сложность: 52%)

У вас имеется N выстроенных в ряд коробок, A красных и B синих шаров. Все красные шары (аналогично и синие) идентичны. Вы можете класть шары в коробки. Разрешается размещать в коробках шары как одного, так и двух видов одновременно. Так же разрешается оставлять некоторые из коробок пустыми. Не обязательно класть все шары в коробки.

Требуется написать программу, которая определяет количество различных способов, которыми возможно заполнить коробки шарами.

Входные данные

Входной файл INPUT.TXT содержит целые числа N, A, B . ($1 \leq N \leq 20, 0 \leq A, B \leq 20$)

Выходные данные

В выходной файл OUTPUT.TXT выведите ответ на задачу.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	1 1 1	4
2	2 1 1	9

Задача №402 Коттеджный поселок

(Время: 3 сек. Память: 16 Мб Сложность: 75%)

Поблизости от столицы Флатландии одна компания решила построить коттеджный поселок. Строительная компания, которая занимается возведением коттеджей, решила раскрасить некоторые коттеджи в розовый цвет, а остальные – в голубой. Но они не могут решить, какой коттедж раскрасить в какой цвет.

Директор компании утверждает, что раскраска симпатичная, если есть хотя бы один розовый коттедж, хотя бы один голубой коттедж, и можно провести такую прямую, что все розовые коттеджи окажутся с одной стороны от нее, а все голубые – с другой стороны (при этом на самой прямой коттеджей быть не должно). На это главный дизайнер возразил, что есть несколько способов сделать симпатичную раскраску.

Помогите им определить, сколько существует различных симпатичных раскрасок.

Входные данные

Первая строка входного файла INPUT.TXT содержит число N – количество коттеджей ($1 \leq N \leq 300$). Следующие N строк содержат координаты коттеджей, каждая строка содержит два целых числа x_i и y_i ($-10^4 \leq x_i, y_i \leq 10^4$).

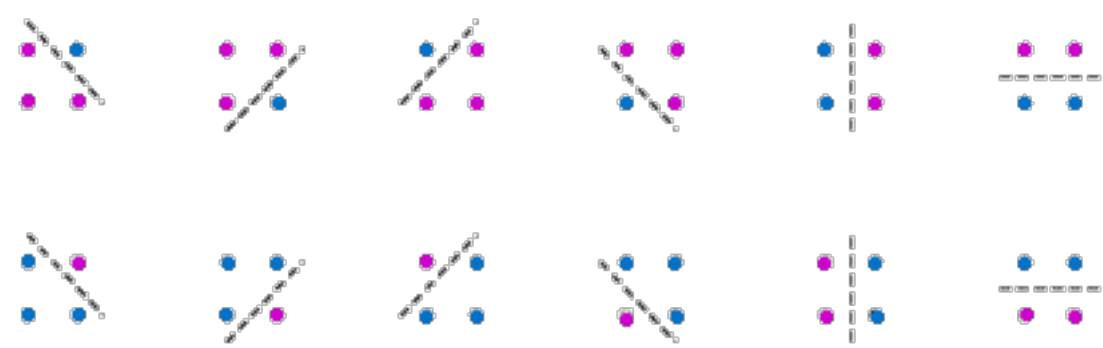
Выходные данные

В выходной файл OUTPUT.TXT выведите одно число – ответ на задачу.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	4 0 0 1 0 1 1 0 1	12

Пояснение к примеру



Задача №403 Обмен валюты

(Время: 1 сек. Память: 16 Мб Сложность: 70%)

Петя работает в обменном пункте во Флатландии. Недавно Петя получил от начальства набор цифр для отображения обменного курса. К сожалению, набор содержит всего по две копии каждой цифры. Теперь Петя хочет узнать, сколько различных обменных курсов он сможет отобразить.

Петя обменивает флатландские доллары на крайландские тугрики. Петя уверен, что курс обмена будет целым числом, которое находится в диапазоне от L до R, включительно.

Входные данные

Входной файл INPUT.TXT содержит два целых числа L и R ($1 \leq L \leq R \leq 10^{18}$).

Выходные данные

В выходной файл OUTPUT.TXT выведите одно целое число - количество обменных курсов, которые Петя может отобразить с использованием полученного набора.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	1 1000	990
2	1 100	100

Задача №404 Игра с камнями

(Время: 2 сек. Память: 16 Мб Сложность: 63%)

Правила игры: исходно на столе располагается кучка, содержащая N камней. Два игрока по очереди берут камни из кучки. Если перед ходом игрока кучка содержит K камней, игрок может взять из нее от 1 до $\lfloor \sqrt{K} \rfloor$ (целую часть от квадратного корня из K) камней, включительно. Например, если в кучке 10 камней, разрешается взять 1, 2 или 3 камня. Если в кучке не осталось камней, то игрок, который должен сделать ход, проигрывает.

Требуется определить: сможет ли первый игрок выиграть, если оба игрока будут придерживаться оптимальной стратегии.

Входные данные

Входной файл INPUT.TXT содержит натуральное число N – число камней в кучке ($N \leq 10^{12}$).

Выходные данные

В выходной файл OUTPUT.TXT выведите WIN, если первый игрок может выиграть вне зависимости от действий второго игрока, либо LOSE, если второй игрок может заставить первого игрока проиграть.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	3	WIN
2	5	LOSE

Задача №405 Туристическое агентство

(Время: 5 сек. Память: 32 Мб Сложность: 80%)

Антон работает в межгалактическом туристическом агентстве. Довольно часто ему приходится прокладывать путь с одной планеты на другую с использованием существующих рейсов космических кораблей. К сожалению, количество рейсов невелико, поэтому пассажирам часто приходится пересаживаться на промежуточных планетах.

Антон заметил, что некоторые планеты используются в качестве промежуточных чаще, чем другие. Он решил провести исследование – для каждой планеты А он хотел бы узнать, сколько существует пар различных планет (В,С), таких что любой путь с планеты В на планету С проходит через планету А.

Помогите Антону!

Входные данные

Первая строка входного файла INPUT.TXT содержит два целых числа: N и M – количество планет и количество рейсов космических кораблей, соответственно ($2 \leq N \leq 20\,000$, $1 \leq M \leq 200\,000$). Следующие M строк описывают рейсы космических кораблей. Каждый рейс связывает две планеты, и им можно воспользоваться в любом из двух направлений. С любой планеты можно добраться до любой другой.

Выходные данные

В выходной файл OUTPUT.TXT выведите N целых чисел – для каждой планеты А выведите количество пар различных планет, таких что любой путь с одной планеты на другую проходит через А.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	7 9	18
	1 2	6
	1 3	6
	1 4	6
	1 5	6
	1 6	6
	1 7	6
	2 3	

4	5
6	7

Задача №406
Криптография

(Время: 1 сек. Память: 16 Мб Сложность: 38%)

Подстановочным называется шифр, при использовании которого каждая буква сообщения заменяется какой-нибудь другой буквой, а все пробелы и знаки препинания удаляются. При этом для сохранения возможности однозначной расшифровки никакие две буквы не превращаются в одну.

Зная тот факт, что частоты появления в осмысленном тексте букв различаются, можно с хорошей точностью установить, каким образом были зашифрованы буквы. Среди всех возможных способов перестановки букв следует выбрать такой, для которого минимально значение величины

$$\sum_c |e_c - p_c|$$

Здесь через e_c и p_c обозначены частоты символа c , рассчитанные для среднего текста и для записки, получающейся из зашифрованного текста после применения данной перестановки.

Требуется по заданным частотам написать программу, расшифровывающую закодированное послание.

Входные данные

В первой строке входного файла INPUT.TXT записаны два целых числа – N и M ($1 \leq N \leq 26$, $1 \leq M \leq 10^5$). Следующие N строк задают буквы, которые, по сведениям Кристины, могли встречаться в исходном тексте. Соответствующая буква c_i – первый символ строки, через пробел от него записана частота этой буквы p_i , заданная с 4 знаками после десятичной точки. Все p_i неотрицательны, а их сумма равна 1. Последняя строка содержит m символов – текст, который необходимо расшифровать.

Выходные данные

В выходной файл OUTPUT.TXT выведите N символов по одному на строке, которые, если верить описанному методу, должны быть сопоставлены символом из входного файла. Если оптимальных решений несколько, выведите любое из них.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	3 5 t 0.1667 r 0.5000 q 0.3333 ecere	r e c

Задача №407
Сдача

(Время: 1 сек. Память: 16 Мб Сложность: 44%)

Когда Миша и Маша покупали подарок, возникла интересная ситуация. У них была в

распоряжении только одна большая купюра, а у продавца – некоторое количество мелочи. Дело происходило утром, поэтому продавцу нужно было экономить мелочь, и он хотел отдать сдачу минимальным количеством монет. Подумав некоторое время, они точно определили, с каким количеством монет продавцу придется расстаться.

А вы сможете решить такую задачу?

Входные данные

В первой строке входного файла INPUT.TXT записано число N ($1 \leq N \leq 10$) – количество различных номиналов монет, содержащихся в кассе. Можно считать, что количество монет каждого номинала достаточно. На следующей строке содержится N целых чисел a_i ($0 < a_i \leq 2000$) – номиналы монет. В третьей строке записано одно число K ($1 \leq K \leq 10^6$) – сумма, которую нужно набрать.

Выходные данные

В выходной файл OUTPUT.TXT выведите минимальное количество монет, которое придется отдать продавцу, или -1, если продавец вообще не сможет дать им сдачу.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	3 1 5 7 19	3
2	5 7 8 9 10 11 13	-1

Задача №408 Письмо

(Время: 1 сек. Память: 16 Мб Сложность: 31%)

Вася хочет отправить Пете письмо на листе шириной K . Он хочет распечатать на нем текст, состоящий из N строк. Необходимо изменить количество ведущих и концевых пробелов так, чтобы строки оказались посередине листа, и их длина стала равна K . Вася считает, что строка находится посередине листа, если количество ведущих пробелов не превосходит количества концевых пробелов и, если при сдвиге строки на один символ вправо (т.е. при добавлении одного пробела в начало строки), указанное свойство нарушается.

Входные данные

Первая строка входного файла INPUT.TXT содержит два целых числа K и N ($1 \leq K \leq 100$, $1 \leq N \leq 1000$). Следующие N строк содержат текст Васиного письма. Каждая строка письма содержит хотя бы один символ, отличный от пробела. Длина каждой строки во входном файле не превосходит 100.

Выходные данные

Если Вася сможет написать письмо, удовлетворяющее всем его требованиям, то в выходной файл OUTPUT.TXT выведите отформатированный текст письма, иначе, выведите фразу “Impossible.” (без кавычек).

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	20 3 ++Привет!++ +Напиши+мне.++ ++++Пока+=)+	++++++Привет!++++++ ++++Напиши+мне.++++ ++++++Пока+=)++++++
2	5 1 Привет.	Impossible.

В приведенных примерах для большей наглядности пробелы заменены на плюсы!

Задача №409 Железная дорога

(Время: 1 сек. Память: 16 Мб Сложность: 26%)

При строительстве новой железной дороги возникли проблемы. Дорога пролегает по холмистой местности, однако сами пути должны идти строго горизонтально. Поэтому руководство строительной компании приняло решение выровнять поверхность земли на этом участке. Главная проблема состоит в том, что привозить или вывозить землю на стройку стоит 10000\$ за кубический метр. Поскольку бюджет железной дороги невелик, этого нельзя себе позволить.

Поэтому главный инженер принял решение выровнять поверхность, используя только землю, из которой состоят холмы. Теперь самая сложная задача состоит в том, чтобы выяснить высоту над уровнем моря, на которой будет пролегать дорога. Это ответственное задание было поручено Вам.

Через каждый метр от начала участка была измерена высота над уровнем моря. Напишите программу, которая по данным измерений рассчитывает искомую высоту.

Входные данные

Первая строка входного файла INPUT.TXT содержит количество N ($1 < N \leq 30000$) точек, в которых была замерена высота. Вторая строка содержит результаты замеров – i -ое число строки содержит высоту над уровнем моря точки, находящейся на расстоянии $(i-1)$ метр от начала участка. Все высоты – целые неотрицательные числа, не превосходящие 10000. Считайте, что между соседними точками измерений земная поверхность строго прямолинейна.

Выходные данные

В выходной файл OUTPUT.TXT выведите ответ на задачу с точностью, не меньшей 10^{-5} .

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	4 0 1 1 0	0.6666666667
2	5 2 2 2 2 2	2.0000000000

Задача №410 Цифровое колдовство

(Время: 1 сек. Память: 16 Мб Сложность: 75%)

В современном мире все становится цифровым. Шаманы в одной сказочной северной стране тоже стараются не отставать от жизни. Последние изыскания показали, что в качестве материальных компонентов для колдовства с большой эффективностью можно использовать длинные веревки, сплетенные из моржовых усов. Каждая веревка кодирует магическое целое число следующим образом: сначала она определенным образом делится на части, соответствующие цифрам магического числа, в каждой из которых можно завязать от нуля до девяти узелков (глобализация вынуждает шаманов использовать обычную десятичную систему счисления). Конечно, при этом для задания очень больших чисел могут понадобиться длинные веревки. Можно надеяться, что вводная часть вам ясна. Шаман племени X наслал порчу на племя Y.

Шаману племени Y удалось выяснить, какое магическое число N было использовано при проведении страшного ритуала. Теперь ему нужно создать веревку с таким числом K, чтобы N + K делилось на фундаментальную магическую константу M. Необходимо сделать это как можно быстрее, поэтому он хочет выполнить задачу, завязав наименьшее число узелков, даже если для этого придется взять самую длинную веревку из потайного хранилища.

Поможете ли вы спасти племя Y (а в след за ним, может быть, и весь мир) от ужасного шамана племени X?

Входные данные

Во входном файле INPUT.TXT записаны два целых числа – N и M. Ограничения: 0 <= N <= 1000, 1 < M <= 1000.

Выходные данные

В выходной файл OUTPUT.TXT выведите целое неотрицательное число K – ответ на задачу.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	5 8	11
2	6 15	24

Задача №411
Квадратное уравнение

(Время: 1 сек. Память: 16 Мб Сложность: 28%)

Сложно найти человека, который любит решать однообразные задачки по математике. В последнее время школьникам стало легче, ведь с появлением компьютеров почти в каждой квартире стало существенно проще проверять себя.

Но программы, в которых решение уравнений является стандартной функцией, установлены не везде. Напишите программу, которая сможет решить уравнение

$$ax^2 + bx + c = 0$$

при заданных коэффициентах a, b и c.

Входные данные

Единственная строка входного файла INPUT.TXT содержит три целых числа a, b и c, каждое из которых не превосходит по модулю 30000. Числа разделяются пробелами.

Выходные данные

На первой строке выходного файла OUTPUT.TXT выведите число действительных корней заданного уравнения. Затем выведите сами корни по одному на строке с ошибкой, не превосходящей 10^{-4} . Если для заданных коэффициентов корней бесконечно много, на единственной строке выходного файла выведите -1.

Примеры

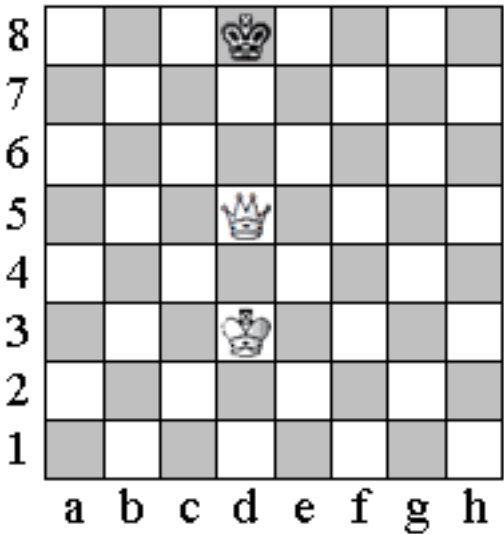
№	INPUT.TXT	OUTPUT.TXT
1	1 -3 2	2 1.000000 2.000000
2	0 -1 6	1 6.000000

Задача №412 Ферзь и король

(Время: 1 сек. Память: 16 Мб Сложность: 32%)

Вася продолжает заниматься шахматами, и теперь он изучает различные окончания. Оказалось, что компьютер очень удобно использовать для анализа позиций. Теперь Вася просит вас написать программу, которая сможет определить, что черный король находится под шахом.

Для начала Васе подойдет программа, которая анализирует игровую ситуацию с тем предположением, что на доске находятся три фигуры - белые король и ферзь и черный король. Черный король находится под шахом, если белый ферзь может за один ход попасть на занимаемую им клетку. Шахматный ферзь может перемещаться по вертикали, горизонтали или диагонали, но, в отличие от коня, не может “перепрыгивать” через другие фигуры.

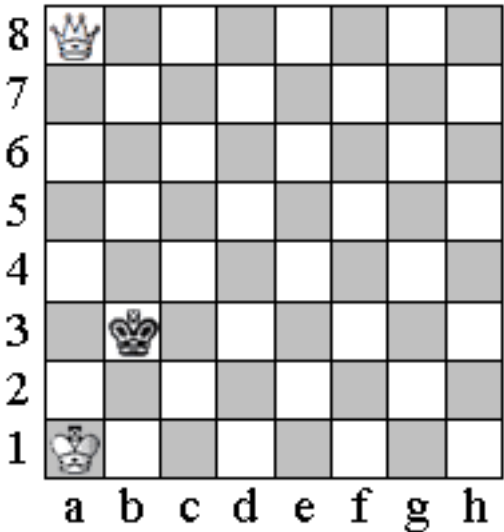


Входные данные

В единственной строке входного файла INPUT.TXT записаны обозначения трех клеток шахматной доски, разделенные пробелами: положения белого короля, белого ферзя и черного короля соответственно. При этом гарантируется, что черный и белый короли не находятся на соседних клетках.

Выходные данные

В выходной файл OUTPUT.TXT выведите слово YES, если черный король находится под шахом и NO, если шаха нет.



Примеры

№	INPUT.TXT	OUTPUT.TXT
1	d3 d5 d8	YES
2	a1 a8 b3	NO

Задача №413

Военная база

(Время: 1 сек. Память: 16 Мб Сложность: 42%)

Со спутника-шпиона получено изображение в некотором волновом диапазоне сверхсекретной военной базы предполагаемого противника. База расположена на Антарктиде, все постройки на ней высечены из кубов льда и имеют на фотографии квадратную форму и не имеют общих фрагментов стен ненулевой длины (по всей видимости, это сделано в целях маскировки от локаторов, работающих в инфракрасном спектре). Благодаря мастерству операторов оказалось, что стены разных построек параллельны границам фотографии.

Для того, чтобы составить сверхсрочный отчет для командования, необходимо узнать, сколько зданий находятся на базе. Напишите программу, которая это сделает.

Входные данные

В первой строке входного файла INPUT.TXT записаны числа N и M ($1 \leq M$, $N \leq 500$) – размеры фотографии в пикселях по вертикали и по горизонтали. Следующие N строк содержат по M символов каждая: символ '.' соответствует пустому месту, '#' – элементу постройки.

Выходные данные

В выходной файл OUTPUT.TXT выведите единственное число – количество построек на базе.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	8 6##. ...##.###.. .###.. .###..	2

Задача №414

Расследование

(Время: 1 сек. Память: 16 Мб Сложность: 40%)

В городском управлении милиции одного прибрежного города ведется расследование крупного дела, в котором могут быть замешаны сотрудники милиции. Было принято решение о тайной установке оборудования для просмотра информации, поступающей через Интернет. Под подозрение попадают два отдела, но добиться выделения денег на покупку двух комплектов оборудования не удалось. К счастью, внутренняя сеть управления имеет древовидную структуру, то есть каждый отдел имеет выход в Интернет через какой-либо другой отдел. Исключение составляет отдел по борьбе с компьютерными преступлениями, который имеет непосредственный доступ в Интернет по модемной линии.

Можно было бы установить оборудование для слежения прямо в этом отделе, но для предотвращения злоупотреблений лучше найти такое расположение, чтобы нарушалась

секретность как можно меньшего количества лишних отделов.

Как наиболее опытному в подобных вопросах сотруднику, решение этой задачи поручили вам. Подчиненные уже пронумеровали все отделы натуральными числами, начиная с 1, первый номер присвоен отделу по борьбе с компьютерными преступлениями.

Входные данные

Первая строка входного файла INPUT.TXT содержит натуральное число N ($N \leq 30000$) – количество отделов. Во второй строке записаны номера отделов, за которыми необходимо установить слежение. На третьей строке находятся n-1 натуральных чисел, i-е из них не больше i и задает номер отдела, к которому подсоединен отдел i + 1.

Выходные данные

В выходной файл OUTPUT.TXT выведите одно число – номер отдела, в котором следует установить следящее оборудование.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	4 3 4 1 1 3	3
2	8 3 6 1 1 2 4 5 1 1	1

Задача №415 Подпись

(Время: 1 сек. Память: 16 Мб Сложность: 34%)

Марсиане Миша и Маша решили вместе подобрать подарок на день рождения Кати. Когда они наконец нашли то, что хотели, и упаковали предмет в красивую коробку, надо было решить, как подписать подарок. Друзья подумали, что лучшим решением будет составить общую подпись так, чтобы в ней как подстроки содержались их имена.

Учтите, что на Марсе принято подписываться полными именами, а они у марсиан могут быть достаточно длинными.

Входные данные

Входной файл INPUT.TXT содержит две строки, в которых записаны полные имена друзей. Имена, как ни странно, состоят из букв латинского алфавита, из которых только первая – прописная. Длина имен не превосходит 1000.

Выходные данные

В выходной файл OUTPUT.TXT выведите кратчайшую строку, в которой встречаются имена Миши и Маши одновременно. Буквы, с которых имена начинаются в этой строке нужно сделать большими. Если существует несколько решений, выведите то, которое меньше в алфавитном порядке.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	Misha Masha	MashaMisha
2	Julya Lyalya	JuLyalya

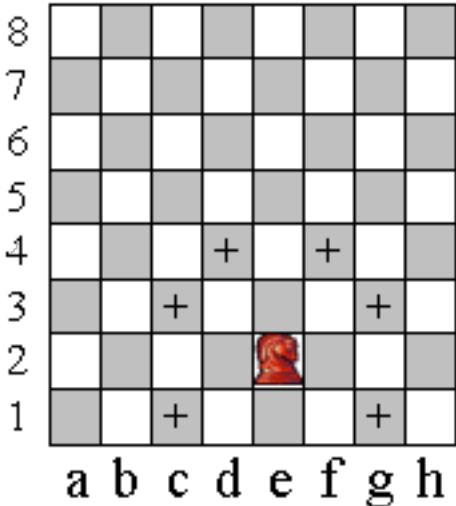
Задача №416

Шахматный конь

(Время: 1 сек. Память: 16 Мб Сложность: 25%)

Вася решил научиться играть в шахматы. Он нашел книгу с записями партий и внимательно их изучает. Может быть, когда-нибудь Вася станет великим шахматистом, но пока он еще учится в начальной школе, и ему нелегко дается шахматная нотация. Больше всего трудностей у Васи вызывают ходы шахматного коня. Он попросил вас написать программу, которая сможет сообщить Васе, на какие клетки можно пойти конем с заданной клетки.

Вы, наверное, тоже знаете, что конь в шахматах всегда перемещается либо на две клетки по горизонтали и на одну по вертикали, либо на одну по горизонтали и на две по вертикали. Вертикали обозначаются маленькими латинскими буквами от а до h, а горизонтали - цифрами от 1 до 8. Любая клетка на шахматной доске обозначается буквой соответствующей вертикали и цифрой соответствующей горизонтали, например, с6 или е2.



Входные данные

Во входном файле INPUT.TXT записано 2 символа – координаты клетки, где стоит конь.

Выходные данные

В выходной файл OUTPUT.TXT в произвольном порядке выведите все координаты клеток, на которые за один ход может попасть конь, находящийся на заданной клетке.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	e2	c1 c3 d4 f4 g1 g3
2	a1	b3 c2

Задача №417

Даты

(Время: 1 сек. Память: 16 Мб Сложность: 34%)

При обработке различной информации часто приходится оперировать с данными о датах. В этой задаче вам нужно составить программу, которая сможет вывести число, месяц и день недели,

которые наступят через K дней, начиная с первого января 2008 года. Напомним, что это был вторник.

Входные данные

Входной файл INPUT.TXT содержит одно целое число K ($0 \leq K \leq 1000$) - количество дней, после первого января, через которое наступит (или наступила) интересующая дата.

Выходные данные

В выходной файл OUTPUT.TXT ваша программа должна записать, какие день недели, число и месяц наступят по прошествии заданного времени. Результаты проверяются автоматически, поэтому вам следует придерживаться формата, показанного в примерах.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	0	Tuesday, 01.01
2	5	Sunday, 06.01

Примечание

По-английски дни недели называются так: Monday, Tuesday, Wednesday, Thursday, Friday, Saturday, Sunday.

Задача №418 Редактор

(Время: 1 сек. Память: 16 Мб Сложность: 39%)

Немногие знают, что первые версии текстового редактора из предыдущей задачи были написаны в России программистом Колей. Для этого он несколько месяцев почти не выходил из подвала, где стояли только диван и компьютер.

Вот одна из проблем, с которыми столкнулся Коля в те времена. Во время работы пользователь набирает какой-то текст, а так же может его редактировать. При этом, даже если итоговый результат полностью помещается на экран, в процессе работы отдельные строки могут иметь слишком большую длину. Мы не будем просить вас повторить Колин подвиг и заново написать редактор. Определите, какой максимальной длины строка получалась в течение набора текста, если вам известно, какие клавиши и в каком порядке нажимал пользователь.

Входные данные

Во входном файле INPUT.TXT записана строка из различных символов - последовательность кнопок на клавиатуре, которые нажимал пользователь. Переводы строк заменены на символ '\'. Первые версии редактора поддерживали три управляющие команды, которые закодированы следующим образом:

- '<' - удаление предыдущего символа (если курсор находится в начале строки, и эта строка не первая, то удаляется предшествующий перевод строки);
- '^' - перемещение в конец предыдущей строки (игнорируется, если курсор находится на первой строке);
- '|' - перемещение в конец следующей строки (игнорируется, если курсор находится на последней строке).

Все остальные символы, содержащиеся в файле, имеют коды от 32 (пробел) и выше и должны пониматься как есть. Число нажатий клавиш не превосходит 10⁵.

Выходные данные

В выходной файл OUTPUT.TXT выведите одно целое число - максимальную длину строки, которая была достигнута в течение работы редактора.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	Hello, World???<<<!	15
2	Hello, World?\This is^<!!! a sample.	17

Задача №419 Палиндром

(Время: 1 сек. Память: 16 Мб Сложность: 41%)

Одно из известных развлечений со словами - составление палиндромов. Палиндромом называется предложение, которое, после удаления из него всех пробелов и знаков препинания, читается одинаково справа налево и слева направо. Создатели одного известного текстового редактора пишут новую версию модуля для проверки орфографии. Они хотят реализовать возможность вывода подсказки для пользователя на тот случай, если он допустил опечатку при наборе какого-нибудь палиндрома. Конечно же, они решили обратиться именно к вам.

Более точно, по заданной строке нужно определить, может ли она быть результатом замены, удаления или добавления одного символа в некотором палиндроме. При этом строчные и прописные латинские буквы не различаются, а все остальные символы должны игнорироваться.

Входные данные

Во входном файле INPUT.TXT содержится заданная строка. Гарантируется, что она содержит хотя бы одну букву. Длина строки не превосходит 10⁵.

Выходные данные

В первой строке выходного файла OUTPUT.TXT выведите YES, если строка может быть получена каким-нибудь из описанных выше преобразований из некоторого палиндрома, и NO в противном случае. В случае положительного ответа во второй строке выведите какой-нибудь из палиндромов, в которых мог допустить опечатку пользователь.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	Never odd or even	YES NeVErODDoREVen
2	Eat it!	NO
3	Mums are not set as a test on Erasmus.	YES Sums are not set as a test on Erasmus.

Задача №420

Химическая формула

(Время: 1 сек. Память: 16 Мб Сложность: 36%)

Запись химической реакции всегда содержит описания нескольких веществ. В свою очередь, описание одного химического вещества - строка, в которой входящие в него атомы химических элементов перечисляются в определенном порядке. При этом последовательности из двух и более одинаковых атомов, идущих подряд, группируются: записывается сокращенное название химического элемента и количество одинаковых элементов подряд. Например, вместо НН пишут Н2. Обозначения химических элементов состоят из одной или двух латинских букв, из которых первая - прописная, а вторая - строчная. В этой задаче не будут рассматриваться более сложные правила. Например, не используются скобки. Вы должны проверить, что заданная последовательность символов подходит под данное выше описание формулы химического вещества. При этом не нужно рассматривать корректность заданной строки, исходя из каких-либо других соображений, даже если они продиктованы здравым смыслом.

Входные данные

В единственной строке входного файла INPUT.TXT записана последовательность символов, содержащая только цифры и строчные и прописные латинские буквы. Гарантируется, что в последовательности перед каждой строчной буквой идет прописная, а все однобуквенные и двухбуквенные подстроки, начинающиеся с прописной буквы - правильные обозначения химических элементов (поэтому здесь даже не приводится их список). Длина последовательности не превосходит 1000 символов.

Выходные данные

В выходной файл OUTPUT.TXT выведите одно слово YES, если данная строка подходит под упрощенное описание формулы химического вещества из условия и NO, если не подходит.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	ОНNaОНNa	YES
2	H2O	YES
3	НН	NO
4	CHC	YES

Задача №421

Треугольники - 2

(Время: 1 сек. Память: 16 Мб Сложность: 37%)

Когда Вова зашел в класс математики, на школьной доске было в беспорядке нарисовано некоторое количество треугольников. Вове очень понравилась идея разрисовывания доски треугольниками, и он принялся изучать ситуацию более внимательно. Вова подумал, что наиболее простой и быстрый способ ровно нарисовать такое количество треугольников - вырезать из какого-нибудь материала шаблон и обводить его на доске мелом. Кроме того Вова предположил, что тот, кто рисовал треугольники, должно быть, очень спешил, и потому перемещал шаблон, не отрывая никакую его часть от доски.

Помогите Вове проверить его гипотезу, если он готов сообщить вам тщательно измеренные координаты вершин треугольников.

Входные данные

В первой строке входного файла INPUT.TXT записано одно целое число - количество треугольников N ($2 \leq N \leq 100$). Последующие N строк описывают треугольники, строка i+1 содержит 3 пары положительных целых чисел, не превосходящих 10000 - координаты вершин треугольника номер i.

Выходные данные

В выходной файл OUTPUT.TXT выведите ответ на задачу: YES, если все треугольники можно было нарисовать, обводя один шаблон и не отрывая этот шаблон от доски, и NO в противном случае.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	3 1 1 1 2 2 1 2 1 2 2 1 2 1 1 2 1 2 2	YES
2	2 1 1 4 1 2 2 1 1 4 1 3 2	NO

Задача №422 Упорядоченные дроби

(Время: 1 сек. Память: 16 Мб Сложность: 38%)

Требуется вывести в порядке возрастания все несократимые дроби, заключённые между 0 и 1, знаменатели которых не превышают N.

Входные данные

Входной файл INPUT.TXT содержит натуральное число N. ($2 \leq N \leq 255$)

Выходные данные

В выходной файл OUTPUT.TXT выведите в каждой строке по дроби – элементу искомой последовательности в таком же формате, как в примерах.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	3	1 / 3 1 / 2 2 / 3
2	5	1 / 5 1 / 4 1 / 3 2 / 5 1 / 2 3 / 5

		2 / 3
		3 / 4
		4 / 5

Задача №423

Сообщение

(Время: 1 сек. Память: 16 Мб Сложность: 46%)

В сообщении, состоящем из заглавных русских букв и пробелов, каждую букву заменили её порядковым номером в русском алфавите (А - 1, Б - 2, ..., Я - 33), а пробел - нулем.

Требуется по заданной последовательности цифр найти количество исходных сообщений, из которых она могла получиться.

Входные данные

Входной файл INPUT.TXT содержит последовательность цифр, состоящую не более чем из 100 цифр.

Выходные данные

В выходной файл OUTPUT.TXT выведите ответ на задачу.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	1025	4
2	21705	3
3	33222	8

Задача №424

Игра умножения

(Время: 1 сек. Память: 16 Мб Сложность: 52%)

Слава и Оля играют в игру умножения: умножают целое число P на одно из чисел от 2 до 9. Слава всегда начинает с $P = 1$, делает умножение, затем число умножает Оля, затем Слава и т.д. Перед началом игры им задают случайное число N , и победителем считается тот, кто первым получит $P \geq N$.

Требуется определить, кто выиграет при заданном числе N , если известно, что Слава и Оля играют наилучшим образом.

Входные данные

Входной файл INPUT.TXT содержит целое N . ($2 \leq N \leq 4\,294\,967\,295$)

Выходные данные

В выходной файл OUTPUT.TXT выведите «Stan wins.», если победит Слава, или «Ollie wins.», если победит Оля.

Примеры

--	--	--

№	INPUT.TXT	OUTPUT.TXT
1	162	Stan wins.
2	17	Ollie wins.
3	34012226	Stan wins.

Задача №425
Прямая и квадраты

(Время: 1 сек. Память: 16 Мб Сложность: 49%)

В прямоугольной декартовой системе координат прямая задана двумя принадлежащими ей точками (0, W) и (100•N, E). Также заданы N² квадратов со сторонами, параллельными осям координат. Квадрат S_{i,j} имеет координаты углов (100•i, 100•j) и (100•i - 100, 100•j - 100), i, j = 1, 2, ..., N.

Требуется найти количество квадратов, имеющих общую точку с прямой.

Входные данные

Входной файл INPUT.TXT содержит в одной строке числа N, W и E, разделенные пробелами. (1 <= N <= 100, 0 <= W, E <= 100•N)

Выходные данные

В выходной файл OUTPUT.TXT выведите количество квадратов, имеющих общую точку с заданной прямой.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	3 150 50	4
2	2 110 120	2

Задача №426
Lines - 2

(Время: 1 сек. Память: 16 Мб Сложность: 50%)

В таблице из N строк и N столбцов некоторые клетки заняты шариками, другие свободны. Выбран шарик, который нужно переместить, и место, куда его нужно переместить. Выбранный шарик за один шаг перемещается в соседнюю по горизонтали или вертикали свободную клетку. Требуется выяснить, возможно ли переместить шарик из начальной клетки в заданную, и, если возможно, то найти путь из наименьшего количества шагов.

Входные данные

В первой строке входного файла INPUT.TXT находится число N, в следующих N строках - по N символов. Символом точки обозначена свободная клетка, латинской заглавной O - шарик, @ - исходное положение шарика, который должен двигаться, латинской заглавной X - конечное положение шарика. (2 <= N <= 250)

Выходные данные

В выходной файл OUTPUT.TXT выведите в первой строке «Y», если движение возможно, или «N», если нет. Если движение возможно, то далее следует вывести N строк по N символов - как и на вводе, но буква X, а также все точки по пути следует заменить плюсами. Если решений несколько, выведите любое.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	5X .0000 0000. @.....	Y +++++ +0000 +++++ 0000+ @++++
2	5 ..X.. 00000@..	N

Задача №427 Несоставляемое число

(Время: 1 сек. Память: 16 Мб Сложность: 48%)

Дано N натуральных чисел. Требуется найти минимальное натуральное число, не представимое суммой никаких из этих чисел, если в эту сумму каждое исходное число может входить не более одного раза.

Входные данные

Входной файл INPUT.TXT в первой строке содержит натуральное число N, не превосходящее 10^4 , далее следуют N строк, в каждой из которых записано по одному натуральному числу, каждое из которых не превосходит 10^9 .

Выходные данные

В выходной файл OUTPUT.TXT выведите ответ на задачу.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	4 1 1 1 5	4
2	4 1 2 4 8	16

Задача №428

SMS

(Время: 1 сек. Память: 16 Мб Сложность: 74%)

Сообщения SMS сотового телефона MOBILA составлены из прописных латинских букв. Если буква первая на кнопке, нужно нажать эту кнопку один раз, чтобы добавить букву в сообщение. Если буква вторая - нужно нажать кнопку дважды и т.д. Так, чтобы набрать слово "SMS", нужно нажать

(PQRS)(PQRS)(PQRS)(PQRS)(MNO)(PQRS)(PQRS)(PQRS)
(PQRS)

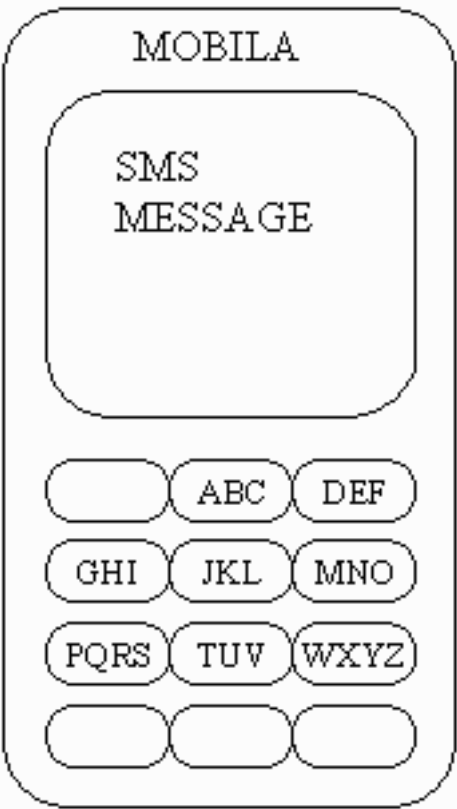
Чтобы ввести две буквы, находящиеся на одной кнопке, нужно между нажатиями клавиши сделать паузу. Например, чтобы ввести сообщение "AA", нужно нажать

(ABC)(пауза)(ABC)

Если на кнопке три буквы, то, как только такая кнопка нажата три раза, последняя буква добавляется в сообщение немедленно, а следующие нажатия той же кнопки относятся к следующей букве сообщения. Аналогично, если на кнопке четыре буквы, то после четырёх нажатий в сообщение будет добавлена последняя буква. То есть последовательность нажатий

(ABC)(ABC)(ABC)(ABC)(пауза)(ABC)

соответствует сообщению "CAA". К сожалению, сотовые телефоны этой модели давно не производятся, и остался только один такой телефон. Он может произвольно вставлять и игнорировать паузы во время ввода сообщения, что может привести к некоторым изменениям в сообщениях. Например, введя MOSCOWQUARTERFINAL, можно получить вместо этого OMSCMNWQTARTERPDEINAL. Вы получили SMS-сообщение и знаете, что оригинальное сообщение содержало N букв. Чтобы определить вероятность угадывания оригинального сообщения, найдите число возможных сообщений, которые могли превратиться в то, которое Вы получили.



Входные данные

Входной файл INPUT.TXT в первой строке содержит длину оригинального сообщения N (1 <= N <= 80). Вторая строка содержит полученное SMS-сообщение, состоящее только из прописных латинских букв длиной от 1 до 80.

Выходные данные

В выходной файл OUTPUT.TXT выведите число сообщений из N букв, которые будучи набранными на этом телефоне, могут превратиться в данное сообщение.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	4 МАМА	1
	2	2

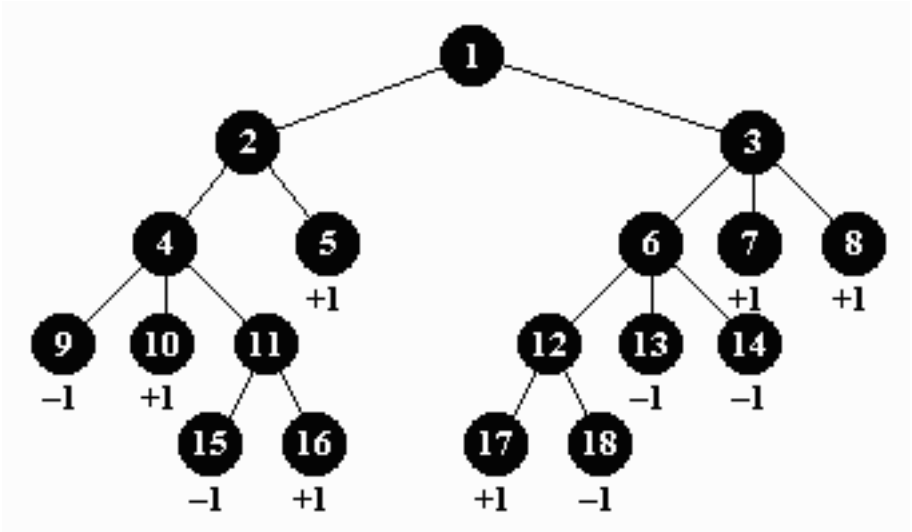
2	WWW	
3	80 QUARTERFINAL	0

Задача №429

Дерево игры

(Время: 1 сек. Память: 16 Мб Сложность: 56%)

Игра для двух игроков определяется её деревом. Соперники делают ходы по очереди. Первый игрок начинает игру. Игра кончается или вничью, или победой одного из игроков. Листья дерева этой игры могут иметь значения, равные одному из трёх чисел: +1 - победа первого игрока, -1 - победа второго игрока, 0 - ничья. Ваша задача - определить, кто выиграет, если оба противника следуют правильной стратегии.



Входные данные

Узлы дерева пронумерованы последовательными целыми числами. Корень дерева всегда имеет номер 1. Первая строка входного файла INPUT.TXT содержит натуральное N - число узлов в дереве игры (N <= 1000). Следующая N - 1 строка описывает узлы - одна строка для каждого узла (за исключением первого). Вторая строка содержит описание второго узла дерева, третья - третьего узла и т.д. Если узел является листом, первый символ строки - L, затем идёт пробел, затем номер родительского узла, ещё пробел и результат игры (+1 - победа первого игрока, -1 - победа второго, 0 - ничья). Если узел внутренний, то строка содержит N - первый символ, затем пробел и номер родительского узла.

Выходные данные

В выходной файл OUTPUT.TXT выведите +1, если выигрывает первый игрок, -1, если второй, и 0 - в случае ничейного исхода.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	7 N 1 N 1 L 2 -1 L 2 +1 L 3 +1 L 3 +1	+1
2	7 N 1 N 1 L 2 -1 L 2 +1 L 3 +1 L 3 0	0

3	18			+1
	N 1			
	N 1			
	N 2			
	L 2 +1			
	N 3			
	L 3 +1			
	L 3 +1			
	L 4 -1			
	L 4 +1			
	N 4			
	N 6			
	L 6 -1			
	L 6 -1			
	L 11 -1			
	L 11 +1			
	L 12 +1			
	L 12 -1			

Задача №430

Дуга на сфере

(Время: 1 сек. Память: 16 Мб Сложность: 51%)

На поверхности планеты, являющейся шаром радиусом R, заданы две точки своими широтой и долготой. Требуется найти минимальную длину пути по поверхности этой планеты из одной точки в другую.

Входные данные

В первой строке входного файла INPUT.TXT находится число R, во второй строке заданы широта и долгота первой точки, в третьей строке - широта и долгота второй точки. Широта измеряется в градусах от -90 до 90, долгота – в градусах от -180 до 180. Радиус R меняется в пределах от 100 до 10000. Все числа вещественные.

Выходные данные

В выходной файл OUTPUT.TXT выведите длину пути с двумя знаками после запятой.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	4000 45 120 0 120	3141.59
2	3437.5 41.766666 -50.233333 41.236388 -51.15	52.04

Задача №431

Путь коня

(Время: 1 сек. Память: 16 Мб Сложность: 46%)

Дана шахматная доска, состоящая из NхN клеток, несколько из них вырезано. Провести ходом коня через невырезанные клетки путь минимальной длины из одной заданной клетки в другую.

Входные данные

В первой строке входного файла INPUT.TXT задано число N (2 <= N <= 50). В следующих N строках содержится по N символов. Символом # обозначена вырезанная клетка, точкой - невырезанная клетка, @ - заданные клетки (таких символов два), соответствующие началу и концу пути коня.

Выходные данные

Если путь построить невозможно, в выходной файл OUTPUT.TXT следует вывести "Impossible", в противном случае вывести такую же карту, как и на входе, но пометить все промежуточные положения коня символом @.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	5@@..@. .@@..@
2	5 @...@. ..##.	@...@. ..##. .@...@ ...@.. @.....
3	5 @..... ..#.. .#...@	Impossible

Задача №432 Грядки

(Время: 1 сек. Память: 16 Мб Сложность: 47%)

Прямоугольный садовый участок шириной N и длиной M метров разбит на квадраты со стороной 1 метр. На этом участке вскопаны грядки. Грядкой называется совокупность квадратов, удовлетворяющая таким условиям:

- из любого квадрата этой грядки можно попасть в любой другой квадрат этой же грядки, последовательно переходя по грядке из квадрата в квадрат через их общую сторону;
- никакие две грядки не пересекаются и не касаются друг друга ни по вертикальной, ни по горизонтальной сторонам квадратов (касание грядок углами квадратов допускается).

Подсчитайте количество грядок на садовом участке.

Входные данные

В первой строке входного файла INPUT.TXT находятся числа N и M через пробел, далее идут N строк по M символов. Символ # обозначает территорию грядки, точка соответствует незанятой территории. Других символов в исходном файле нет. (1 <= N, M <= 200)

Выходные данные

В выходной файл OUTPUT.TXT выведите количество грядок на садовом участке.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	5 10 ##.....#. .#..#...#. .###.....#. ..##.....#.#.	3
2	5 10 ##..#####. .#.#.#..... ###..##.##. ..##.....# .####.#####	5

Задача №433

Школа танцев

(Время: 1 сек. Память: 64 Мб Сложность: 62%)

В школу бальных танцев профессора Падеграса записались n учеников — мальчиков и девочек. Профессор построил их в один ряд, и хочет отобрать из них для первого занятия группу стоящих подряд учеников, в которой количество мальчиков и девочек одинаково. Сколько вариантов выбора есть у профессора?

Входные данные

В первой строке входного файла INPUT.TXT задано число n (1 ≤ N ≤ 10⁶). Во второй строке задается описание построенного ряда из мальчиков и девочек — строка из n символов a и b (символ a соответствует девочке, a символ b — мальчику).

Выходные данные

В единственной строке выходного файла OUTPUT.TXT должно содержаться единственное число — количество вариантов выбора требуемой группы.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	3 bab	2
2	8	13

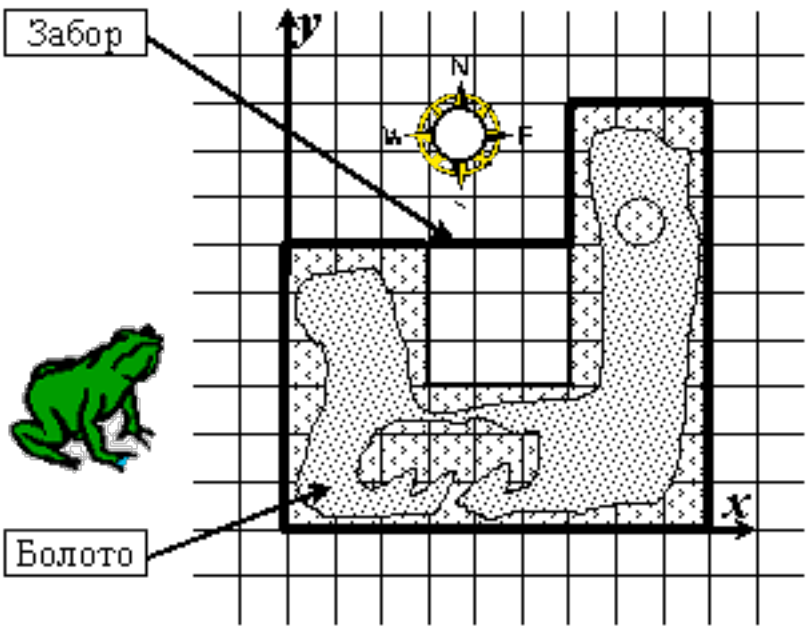
Задача №434

Стеклянный забор

(Время: 1 сек. Память: 16 Мб Сложность: 72%)

В известном городе Санкт-Тверь решили построить новый микрорайон, представляющий в плане прямоугольную область. Границы микрорайона и его улицы по проекту ориентированы строго по сторонам света, причем улицы разбивают микрорайон на кварталы размером 1 км × 1 км.

Во время привязки исходного проекта к местности выяснилось, что некоторые кварталы по проекту микрорайона оказываются полностью или частично расположенными на топком болоте. Область, занимаемая болотом, связна и со всех сторон окружена подлежащими застройке кварталами микрорайона (область связна, если из любой ее точки можно добраться в любую другую, не выходя за пределы области).



Для сохранения экологии местности и обеспечения безопасности жителей занятую болотом область решили оградить стеклянным забором. Забор должен проходить только по границам кварталов проектируемого микрорайона, отделяя болото, и, возможно, некоторые кварталы проекта, не занятые болотом, от остальной части микрорайона.

Для экономии строительных материалов забор должен иметь минимальную длину. Среди всех заборов минимальной длины нужно выбрать тот, для которого площадь части микрорайона, попадающей внутрь забора, минимальна.

Требуется написать программу, которая спроектирует забор с заданными выше свойствами.

Входные данные

Входной файл INPUT.TXT содержит описание многоугольника — границы области, состоящей только из кварталов с заболоченными участками. Стороны многоугольника параллельны осям координат.

В первой строке задано целое число N — количество вершин в многоугольнике ($4 \leq N \leq 10^5$, N четное). В каждой из следующих N строк заданы два целых числа — координаты очередной вершины при обходе этого многоугольника против часовой стрелки. Все числа не превосходят 10^9 по абсолютной величине. Никакие три последовательные вершины границы не лежат на одной прямой. Граница многоугольника не содержит самопересечений и самокасаний.

Выходные данные

Выходной файл OUTPUT.TXT должен содержать описание многоугольника, определяющего искомый забор. Формат описания многоугольника тот же, что и для входных данных. Никакие три последовательные вершины этого многоугольника не должны лежать на одной прямой.

Пример

№	INPUT.TXT	OUTPUT.TXT

1	8	6
	0 0	0 0
	9 0	9 0
	9 9	9 9
	6 9	6 9
	6 3	6 6
	3 3	0 6
	3 6	
	0 6	

Задача №435

Несчастливые номера

(Время: 1 сек. Память: 16 Мб Сложность: 88%)

Обычно автобусный билет с номером, состоящим из 6 цифр, считается счастливым, если сумма первых трех цифр его номера была равна сумме трех последних. Школьник Вася очень любил получать счастливые билеты, однако это случалось не так часто. Поэтому для себя он изменил определение счастливого билета.

Счастливым он считал тот номер, сумма некоторых цифр которого равнялась сумме оставшихся цифр. В его представлении билет с номером 561743 счастливый, так как 5+1+4+3=6+7.

Вася вырос, но по привычке в номерах различных документов пытается найти признаки счастливого номера 😊. Для этого он расширил свое определение счастливого номера на N значные номера лицевых счетов и других документов, состоящих из цифр от 0 до K. Номер документа он называет счастливым, если сумма некоторых цифр этого номера равняется сумме оставшихся. Остальные номера для него несчастливые. К сожалению, несмотря на расширенное понимание “счастья”, несчастливых номеров остается еще много...

Вам предлагается определить количество несчастливых N-значных номеров, которые можно составить, используя цифры от 0 до K. В номерах допускается любое количество ведущих нулей.

Входные данные

Входной файл INPUT.TXT содержит описание вида номеров в виде двух чисел N и K, разделенных пробелом. (1 <= N <= 100, 1 <= K <= 9, N*K <= 300)

Выходные данные

В выходной файл OUTPUT.TXT выведите количество несчастливых номеров для заданных N и K.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	1 7	7
2	4 3	164
3	11 9	50184219171

Задача №436

Счастливые цифры

(Время: 1 сек. Память: 16 Мб Сложность: 70%)

Школьнику Васе нравятся числа, которые заканчиваются счастливыми для него цифрами К. Поэтому каждый раз, когда он видит какое-нибудь натуральное число N, он сразу пытается подобрать такое D ($D \geq 2$), что число N в системе счисления с основанием D заканчивается как можно большим количеством цифр К.

Требуется написать программу, которая по заданным числам N и К найдет такое D, чтобы число N в системе счисления с основанием D заканчивалось как можно большим количеством цифр К.

Входные данные

Входной файл INPUT.TXT содержит два целых десятичных числа N и К ($1 \leq N \leq 10^{11}$; $0 \leq K \leq 9$).

Выходные данные

В выходной файл OUTPUT.TXT выведите два числа: D – искомое основание системы счисления и L – количество цифр К, которым заканчивается запись числа N в этой системе счисления. Если искомым D несколько, выведите любое из них, не превосходящее 10^{12} (такое всегда существует).

Примеры

№	INPUT.TXT	OUTPUT.TXT	Комментарий
1	49 1	3 2	$49_{10} = 1211_3$
2	7 5	3 0	Ни в одной системе счисления 7 не заканчивается на цифру 5

Задача №437 Тапкодер

(Время: 3 сек. Память: 16 Мб Сложность: 78%)

Ассоциация Тапкодер организует Всемирное парное соревнование сильнейших программистов. К участию в соревновании допущены первые 2^K зарегистрировавшихся участников, которым присвоены номера от 1 до 2^K .

Соревнование будет проходить по олимпийской системе. В первом туре первый участник встречается со вторым, третий с четвертым и так далее. В каждой паре победителем становится участник, первым решивший предложенную задачу, при этом ничьих не бывает. Все победители очередного тура и только они являются участниками следующего тура. В каждом туре пары составляются из участников в порядке возрастания присвоенных им номеров. Соревнование продолжается до тех пор, пока не останется один победитель.

Организаторам стало известно, что некоторые пары участников заранее договорились о результате встречи между собой, если такая встреча состоится. Для всех остальных встреч, кроме п договорных, возможен любой исход.

Некоторые M участников соревнования представили свои резюме в ассоциацию Тапкодер с целью поступления на работу. Организаторов интересует, до какого тура может дойти каждый из претендентов при наиболее благоприятном для него стечении обстоятельств. При этом для каждого участника в отдельности считается, что все недоговорные встречи, в том числе те, в которых он не участвует, закончатся так, как ему выгодно, а все состоявшиеся договорные встречи закончатся в соответствии с имеющимися договоренностями.

Требуется написать программу, которая для каждого из претендентов определяет максимальный номер тура, в котором он может участвовать.

Входные данные

В первой строке входного файла INPUT.TXT заданы три целых числа K ($1 \leq K \leq 60$), N ($0 \leq N \leq 100\,000$) и M ($1 \leq M \leq 100\,000$). В следующих N строках описаны N пар участников, которые договорились между собой о том, что первый из двух участников пары выиграет встречу, если она состоится. Гарантируется, что каждая пара участников присутствует во входных данных не более одного раза, при этом, если задана пара $X\ Y$, то пары $Y\ X$ быть не может, кроме того, $X \neq Y$. В последней строке файла перечислены номера участников, желающих работать в Тапкодере, в порядке возрастания их номеров. Все номера претендентов на работу различны.

Выходные данные

Выходной файл OUTPUT.TXT должен содержать M целых чисел — максимальные номера туров, до которых могут дойти соответствующие претенденты на работу. Туры нумеруются от 1 до K .

Примеры

№	INPUT.TXT	OUTPUT.TXT	Комментарий
1	2 0 3 1 3 4	2 2 2	У каждого из участников есть возможность выйти в финал, так как договорных матчей нет
2	3 1 1 3 1 1	3	Если четвертый участник выиграет у третьего, то договорная встреча первого и третьего не состоится, что благоприятно для первого
3	3 3 4 1 2 1 3 4 1 1 2 3 4	3 1 2 3	Первому участнику благоприятно во втором туре играть с третьим, а не с четвертым, в свою очередь, четвертый может выиграть у третьего и также выйти в финал

Задача №438 Сочи-2014

(Время: 1 сек. Память: 16 Мб Сложность: 90%)

К предстоящей олимпиаде в Сочи требуется возвести N олимпийских объектов. Процесс строительства каждого объекта определяется освоением выделяемых на него денежных средств.

В строительстве объектов готовы участвовать K фирм. Фирмы имеют разные строительные мощности, выраженные в количестве денежных средств, которые фирма может осваивать в единицу времени.

В каждый момент времени фирма может осуществлять работы только на одном объекте. В строительстве одного объекта не могут одновременно участвовать несколько фирм. В любой момент времени любой объект может быть передан для продолжения строительства любой фирме.

Администрация строительства олимпийских объектов заинтересована в скорейшем освоении денежных средств, поэтому хочет составить такой график работ, при следовании которому строительство будет завершено в кратчайшие сроки. В графике будет указано время, в течение которого тот или иной объект будет строиться какой-то фирмой.

Напишите программу, результаты работы которой позволят администрации построить

требуемый график.

Входные данные

Первая строка входного файла INPUT.TXT содержит целое число N — количество объектов ($1 \leq N \leq 50$). Во второй строке содержатся разделенные пробелами целочисленные значения $S_1, S_2, S_3, \dots, S_N$ объемов денежных средств, выделяемых для строительства каждого из объектов. Числа S_i выражены в тысячах рублей, положительные и не превышают 1000.

В третьей строке находится целое число K — количество строительных фирм ($1 \leq K \leq 50$). Четвертая строка содержит разделенные пробелами целочисленные значения мощностей каждой из фирм $V_1, V_2, V_3, \dots, V_K$ в тыс. руб / час. Числа V_j положительные и не превышают 1000.

Выходные данные

Первая строка выходного файла OUTPUT.TXT содержит действительное число T — время в часах окончания всех работ, считая с начала строительства, выведенное не менее чем с тремя точными знаками после запятой. Далее в каждой строке выходного файла содержатся разделенные пробелами три числа: t, i, j , где действительное число t — время от начала строительства в часах, в которое j -я фирма приступает к строительным работам на i -м объекте.

Значения времен необходимо выводить с максимально возможной точностью. Строки должны быть отсортированы по неубыванию t .

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	2 24 20 2 3 2	8.800 0 1 1 0 2 2 6.4000000 1 2 6.4000000 2 1
2	3 100 100 100 4 5 5 10 10	12.00000 0 1 3 0 2 4 0 3 1 4 2 2 4 3 4 8 1 1 8 3 4 8 2 3

Задача №439 Закон Амдала

(Время: 1 сек. Память: 16 Мб Сложность: 23%)

Параллельное программирование изучает методы построения программ, которые будут выполняться на нескольких процессорах. В результате решения одной из первых задач этого раздела информатики появился закон Амдала.

Задача Амдала формулировалась так. Имеется n процессоров и p процентов вычислений не могут выполняться параллельно. Во сколько раз быстрее можно выполнить вычисления по сравнению с одним процессором?

Например, если $n = 10$, $p = 50$, а на одном процессоре все вычисления выполняются за время t . Тогда первая половина вычислений (50%) будет выполнена за время $t/(2 \cdot 10)$, а вторая - за время $t/2$. Общее время вычислений в этом случае составит $t/2 + t/20 = 11 \cdot t/20$, а ускорение по сравнению с одним процессором составит 20/11 раз.

Если же $n = 10$, $p = 25$, и на одном процессоре все вычисления выполняются за время t . Тогда 75% вычислений будут выполнены за время $3 \cdot t/(4 \cdot 10)$, а оставшиеся 25% - за время $t/4$. Общее время вычислений в этом случае составит $t/4 + 3 \cdot t/40 = 13 \cdot t/40$, а ускорение по сравнению с одним процессором составит 40/13 раз.

Даны числа n и p . Напишите программу, решающую задачу Амдала.

Входные данные

Входной файл INPUT.TXT содержит 2 целых числа n и p . ($1 \leq n \leq 1000$, $0 \leq p \leq 100$).

Выходные данные

В выходной файл OUTPUT.TXT выведите ответ на задачу с точностью, не меньшей 10^{-6} .

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	10 50	1.818181818
2	10 25	3.076923077
3	1000 100	1.000000000

Задача №440 Биатлон

(Время: 1 сек. Память: 16 Мб Сложность: 24%)

На Зимних Олимпийских Играх традиционно проводятся соревнования по биатлону. Как известно, этот вид спорта содержит лыжные гонки и стрельбу по мишеням из винтовки. На каждом огневом рубеже расположены 5 мишеней. Каждая из них имеет форму круга радиусом 10 см, а расстояния между центрами соседних мишеней одинаковы и равны 25 см. Центры мишеней при этом расположены на одной горизонтали.

Введем прямоугольную систему координат так, что начало координат расположено в центре самой левой мишени, ось Ox направлена вправо, а ось Oy - вверх. Таким образом, центры мишеней имеют координаты $(0, 0)$, $(25, 0)$, $(50, 0)$, $(75, 0)$ и $(100, 0)$.

Для информационного обеспечения проведения соревнований было решено разработать систему подсчета количества пораженных мишеней. Эта система по точкам, в которые попали пули после выстрелов спортсмена, должна определять количество пораженных мишеней. Мишень считается пораженной, если в нее попала хотя бы одна пуля (при этом, разумеется, если в мишень попали две или больше пуль, то попадание считается только один раз).

На спринтерской гонке на каждом огневом рубеже у спортсмена есть 5 пуль. Вам даны координаты точек, в которые попали пули после выстрелов спортсмена. Определите количество пораженных мишеней.

Входные данные

Входной файл INPUT.TXT содержит ровно пять строк: i -ая из них содержит два целых числа x_i

и y_i - координаты точки, в которую попала пуля после i -ого выстрела спортсмена. Все числа во входном файле не превосходят 1000 по модулю.

Выходные данные

В выходной файл OUTPUT.TXT выведите единственное число – число пораженных мишеней.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	0 0 25 0 50 0 75 0 100 0	5
2	0 0 0 0 0 0 75 0 100 0	3

Задача №441 Карточки - 2

(Время: 1 сек. Память: 32 Мб Сложность: 73%)

Вася и Петя играют в следующую игру. Вася кладет на стол два ряда карточек. Первый ряд состоит из n карточек, на каждой из которых написано некоторое число a_i . Второй ряд состоит из n карточек, на каждой из которых написано некоторое число b_i .

Пете требуется переупорядочить карточки первого ряда так, чтобы на столе получилось два одинаковых ряда карточек. За одну секунду Петя может поменять местами i -ую и $(i+1)$ -ую $(1 \leq i \leq N-1)$ карточки первого ряда.

Помогите Пете переупорядочить карточки, затратив на это минимальное время.

Входные данные

Первая строка входного файла INPUT.TXT содержит целое число N $(1 \leq N \leq 10^5)$. Вторая строка содержит N целых чисел a_i . Третья строка содержит N целых чисел b_i . Все числа по абсолютной величине не превосходят 10^6 .

Выходные данные

В выходной файл OUTPUT.TXT выведите количество секунд, за которые Петя сможет переупорядочить карточки требуемым образом. Если переупорядочить карточки требуемым образом невозможно, выведите единственное число -1.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	5 3 2 3 4 5	7

	5	4	3	2	3	
2	8					-1
	1	2	1	2	1	2
	2	1	2	1	2	2

Задача №442

Шрифты

(Время: 1 сек. Память: 16 Мб Сложность: 45%)

Третий за неделю чек по заправке картриджа для принтера в бухгалтерии приняли без особого удовольствия. Судя по всему, надо серьезно подойти к вопросу экономного использования тонера - на учет должна быть поставлена каждая буква. Вам поручена реализация утилиты, которая будет обрабатывать HTML-документы старых версий и подсчитывать в них количество символов каждого размера. HTML-документ состоит из обычного текста и тегов - управляющих элементов, заключенных в угловые скобки. Для управления размером шрифта используется тег font с атрибутом size: . Если x - натуральное число, то размер шрифта устанавливается равным x. Кроме того, x может иметь вид +у или -у, где у - целое положительное число. В этом случае размер шрифта соответственно увеличивается или уменьшается на у. Действие тега заканчивается с появлением соответствующего закрывающего тега . Все остальные теги вы не должны обрабатывать. Можете считать, что теги не содержат лишних пробельных символов. По умолчанию размер шрифта равен 10, и изменения не будут выводить его из интервала от 1 до 50.

Входные данные

Входной файл INPUT.TXT содержит описание HTML-документа, по длине не превосходящее 5000 символов.

Выходные данные

В выходной файл OUTPUT.TXT выведите для каждого встречающегося в документе размера шрифта на отдельной строке через пробел его размер и количество соответствующих ему непробельных (с ASCII кодами не равными 9, 10, 13 и 32) символов в порядке возрастания размера.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	<pre> <HTML> <BODY> Fonts: normal tiny bigger <p> as before very big smaller again ... </BODY> </HTML> </pre>	<pre> 2 4 7 12 10 20 11 6 15 10 </pre>

Задача №443

Хеш-функция

(Время: 2 сек. Память: 16 Мб Сложность: 37%)

В задачах поиска часто используются так называемые хеш-функции. Одним из важнейших классов хеш-функций являются так называемые полиномиальные хеш-функции.

Пусть дана строка $S = s_1s_2 \dots s_l$, состоящая из цифр от 0 до 9. Тогда значение полиномиальной хеш-функции $p(S, x, m)$ вычисляется следующим образом:

$$p(S, x, m) = (\sum_{i=1}^l s_i x^{i-1}) \bmod m$$

($a \bmod b$ обозначает остаток от деления числа a на число b). Например, пусть $S = 0123$, тогда $p(S, 2, 5) = (0 \cdot 1 + 1 \cdot 2 + 2 \cdot 4 + 3 \cdot 8) \bmod 5 = 4$.

Вам даны множество из n строк $(S^{(1)}, S^{(2)}, \dots, S^{(n)})$, каждая из которых состоит только из цифр от 0 до 9, и числа m и x . Необходимо найти количество таких пар (i, j) , где $1 \leq i, j \leq n, i < j$, что $p(S^{(i)}, x, m) = p(S^{(j)}, x, m)$.

Входные данные

Первая строка входного файла INPUT.TXT содержит три целых числа: n ($1 \leq n \leq 30000$), m ($1 \leq m \leq 2000$), x ($1 \leq x \leq 100$). Далее идут n строк, каждая из которых содержит по одной строке из данного множества: 2-ая строка входного файла содержит $S^{(1)}$, 3-я - $S^{(2)}$, ..., $(i + 1)$ -ая - $S^{(i)}$, ..., $(n + 1)$ -ая - $S^{(n)}$. Длины $S^{(i)}$ не превосходят 100, $S^{(i)}$ не пусты и состоят только из цифр от 0 до 9.

Выходные данные

В выходной файл OUTPUT.TXT выведите ответ на задачу.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	8 3 8 1234 239 366 261 32890 43823490 382390 3043840	11
2	5 10 100 1 2 3 4 5	0

(Время: 1 сек. Память: 16 Мб Сложность: 30%)

В наше время создатели офисных приложений стараются сделать все для удобства пользователя. Поэтому даже такая мелочь, как представление на экране списков чисел - например, для вывода номеров страниц, - должна быть тщательно проработана.

Вы должны реализовать функцию, которая по заданному набору целых чисел будет формировать строку, являющуюся его самым коротким текстовым представлением. Текстовое представление - строка, состоящая из разделенных запятыми чисел и диапазонов чисел вида $< a, \dots, b >$, которые используются для записи набора всех чисел от a до b . При этом все числа, входящие в строку, должны быть упорядочены по возрастанию в том порядке, в котором они встречаются в строке.

Входные данные

Входной файл INPUT.TXT содержит целое число N ($1 \leq N \leq 1000$) - размер набора. Вторая строка содержит N задающих набор целых чисел, по абсолютной величине не превосходящих 10000, разделенные пробелами. Одно число может встречаться в этом описании несколько раз.

Выходные данные

В выходной файл OUTPUT.TXT выведите одно из кратчайших текстовых представлений заданного набора чисел. Следите за правильной расстановкой пробелов. Выходной файл в примере содержит ровно четыре пробела.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	7 1 3 5 -1 3 4 6	-1, 1, 3, ..., 6

Задача №445 Покупки

(Время: 1 сек. Память: 16 Мб Сложность: 56%)

Во многих фирмах, занимающихся торговлей, существует должность менеджера по закупкам. Как известно, они занимаются тем, что по торговому плану, представляющему собой список наименований товаров, для каждого из которых указано необходимое количество, закупает указанные в нем товары на оптовых базах. Торговый план при этом составляется руководством компании. Иногда у менеджеров по закупкам возникает желание принести выгоду не только своей фирме, но и себе.

Только что, как раз после подписания очередного торгового плана на заказ n наименований товаров, открылась новая оптовая база. Как это часто бывает сразу после открытия, ее цены на многие товары ниже заложенных в план. Наверное, этим можно воспользоваться.

На закупку товаров были выделены деньги из расчета того, что все товары будут закупаться на старой оптовой базе. Менеджер хочет, воспользовавшись возможностью покупать товары на новой базе, потратить как можно меньше денег на закупку требуемого количества товаров (не потраченные деньги он, конечно, сможет забрать себе).

Чтобы не вызывать сильных подозрений, производить на новой базе закупки, суммарная запланированная стоимость которых была больше, чем d денежных единиц, не следует. Осталось только рассчитать, какие товары и в каком количестве следует закупать на новой базе, чтобы осталось как можно больше не потраченных денег.

Входные данные

Первая строка содержит четыре числа: n ($1 \leq n \leq 1000$), d , а так же k_1 и k_2 ($1 \leq k_1, k_2 \leq 1000$) - количества наименований товаров, имеющих на открытых ранее и новой базе соответственно. После этого идут n строк, каждая из которых содержит название товара в плане и его количество (положительное вещественное число). За ними следуют два блока из k_1 и k_2 строк соответственно, отделенные от предыдущего и разделенные между собой переводом строки - наименования товаров на базах и цены за единицу товара соответственно. Все цены являются положительными числами, даже на новой базе.

Названия товаров состоят из не более, чем 100 латинских букв и символов подчеркивания, при этом регистр букв не учитывается. Вещественные числа заданы не более чем с двумя знаками после десятичной точки и по величине не превосходят 10^6 . Гарантируется, что все товары из плана можно купить на старой базе.

Выходные данные

В выходной файл OUTPUT.TXT выведите n вещественных чисел, по одному на строке, задающих количество соответствующего товара, закупаемого на новой базе. На i -ой строке выведите количество товара, идущего i -ым в плане. Ошибки менее 0.01 будут игнорироваться.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	4 11.00 5 4	2.0000
	Salt 3	0.5000
	suGar 0.5	0.0000
	Powdered_Milk 7	0.0000
	fLoUR 8	
	flOuR 1	
	CINNAmOn 8	
	saLt 0.5	
	SuGAR 20	
	poWdErED_MILk 3	
	FLOur 2	
	SALT 0.4	
	liQUID_HydRoGen 10000	
	sUGAR 10	

Задача №446 Табло

(Время: 1 сек. Память: 16 Мб Сложность: 20%)

На хоккейном стадионе в одном большом городе расположено большое прямоугольное табло. Оно имеет n строк и m столбцов (то есть состоит из $n \times m$ ячеек). Во время хоккейного матча это табло служит для отображения счета и времени, прошедшего с начала тайма, а в перерывах на нем показывают различную рекламу.

В связи с этим возникла задача проверки возможности показа на этом табло определенной рекламной заставки. Заставка также, как и табло, имеет размер n строк на m столбцов. Каждая из ячеек заставки окрашена в один из четырех цветов - трех основных: красный - R, зеленый - G,

синий - В и черный - .(точка).

Каждая из ячеек табло характеризуется своими цветопередаточными возможностями. Любая из ячеек табло может отображать черный цвет - это соответствует тому, что на нее вообще не подается напряжение. Также каждая из ячеек может отображать некоторое подмножество множества основных цветов. В этой задаче эти подмножества будут кодироваться следующим образом:

- 0 - ячейка может отображать только черный цвет;
- 1 - ячейка может отображать только черный и синий цвета;
- 2 - ячейка может отображать только черный и зеленый цвета;
- 3 - ячейка может отображать только черный, зеленый и синий цвета;
- 4 - ячейка может отображать только черный и красный цвета;
- 5 - ячейка может отображать только черный, красный и синий цвета;
- 6 - ячейка может отображать только черный, красный и зеленый цвета;
- 7 - ячейка может отображать только черный, красный, зеленый и синий цвета.

Напишите программу, которая по описанию табло и заставки определяет: возможно ли на табло отобразить эту заставку.

Входные данные

Первая строка входного файла INPUT.TXT содержит целые числа n и m ($1 \leq n, m \leq 100$). Далее идут n строк по m символов каждая - описание заставки. Каждый из символов описания заставки принадлежит множеству {R, G, B, .} . Их значения описаны выше.

После этого идет описание табло. Оно содержит n строк по m чисел, разделенных пробелами. Значения чисел описаны выше.

Выходные данные

В выходной файл OUTPUT.TXT выведите YES, если на табло возможно отобразить заставку и NO - в противном случае.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	3 3 .GB R.B RG. 0 1 2 3 4 5 6 7 0	NO
2	2 3 RGB .G. 7 7 7 7 7 7	YES

Требуется найти последнюю ненулевую цифру числа $N! = 1*2*3*\dots*N$.

Входные данные

Входной файл INPUT.TXT содержит единственное натуральное число N ($N \leq 9999$).

Выходные данные

В выходной файл OUTPUT.TXT выведите ответ на задачу.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	1	1
2	5	2

Задача №448 Простые гири

(Время: 1 сек. Память: 16 Мб Сложность: 54%)

Имеются гири с массами: 1 г, 2 г, ..., N г. Требуется написать программу, распределяющую эти гири на максимально возможное количество пар так, чтобы суммарный вес гирь в каждой паре выражался простым числом.

Входные данные

Входной файл INPUT.TXT содержит единственное натуральное число N , не превосходящее 500 000.

Выходные данные

В выходной файл OUTPUT.TXT выведите список найденных пар. Каждая пара выводится в одной строке через пробел.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	7	1 6 7 4 5 2

Задача №449 Плавающие числа

(Время: 1 сек. Память: 16 Мб Сложность: 41%)

Дано N целых чисел. Каждое из них можно один раз изменить не более чем на целую величину L как в сторону увеличения, так и в сторону уменьшения или оставить без изменения. Если после такой операции некоторые из чисел оказываются равными, то они засчитываются за одно. С данными числами произвели указанную операцию таким образом, что осталось минимально возможное количество чисел. Требуется написать программу для определения этого количества.

Входные данные

Входной файл INPUT.TXT содержит в первой строке натуральные числа L и N ($N \leq 100$, $L \leq 3200$), во второй строке N чисел (в диапазоне от -32000 до 32000), записанных через пробел.

Выходные данные

В выходной файл OUTPUT.TXT выведите единственное число – ответ на задачу.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	10 3 11 21 27	1
2	5 3 6 10 27	2

Задача №450 Бутылки - 2

(Время: 1 сек. Память: 16 Мб Сложность: 51%)

В цех вторичной переработки поступают бутылки N видов: A, B, C, ... (первые N заглавных букв латинского алфавита). Бутылки поступают на переработку партиями из N контейнеров, причем в каждом контейнере могут находиться бутылки различных видов. Перед вторичной переработкой бутылок специальные рабочие сортируют их по видам таким образом, чтобы после сортировки в каждом из поступивших контейнеров остались бутылки только одного вида. В каждом из контейнеров может помещаться неограниченное количество бутылок.

Требуется написать программу, которая определяет минимальное количество перемещений бутылок, обеспечивающих их сортировку по видам, причем за каждое перемещение можно переместить только одну бутылку из одного контейнера в другой.

Входные данные

Входной файл INPUT.TXT состоит из N+1 строк. В первой строке записано число N ($1 \leq N \leq 8$). Во второй строке располагаются разделенные пробелами N целых числа, соответствующие количеству бутылок вида A, B, C, ... в первом контейнере. В последующих строках содержится аналогичная информация для второго, третьего, ..., N-го контейнеров соответственно. Известно, что количество бутылок в каждом из контейнеров не превосходит 32767.

Выходные данные

Выходной файл OUTPUT.TXT должен состоять из двух строк. В первой располагается последовательность из символов A, B, C, ..., которая определяет какого вида бутылки находятся после сортировки в 1-м, 2-м, ..., N-м контейнерах. Во второй строке располагается число, определяющее искомое количество перемещений бутылок. Если возможно несколько вариантов ответа, то необходимо выдать любой из них.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	4 1 2 3 4 5 6 7 8 9 10 11 12	ABCD 102

Задача №451

Арифметическое выражение

(Время: 1 сек. Память: 16 Мб Сложность: 56%)

Требуется вычислить значение арифметического выражения, в записи которого могут использоваться вещественные числа, круглые скобки, операции «+», «-», «*» и «/», а так же функции cos(x) и sin(x). Вычисление следует проводить согласно синтаксису языка Pascal.

Входные данные

Входной файл INPUT.TXT содержит единственную строку, в которой находится запись арифметического выражения. Длина строки не превышает 200 символов. В строке могут содержаться любые символы. Гарантируется, что при корректном вычислении выражения не будет возникать переполнения 6-байтного вещественного типа и деления на ноль.

Выходные данные

В выходной файл OUTPUT.TXT выведите значение заданного арифметического выражения с точностью, не меньше 10^{-3} . В том случае, когда в выражении присутствуют синтаксические ошибки, следует вывести «Error».

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	2+3	5
2	7*8 - 120/6 + 4	40
3	10/(6-2)+8	10.5
4	sin(3.14/2)+cos(2-6/3)	2
5	three plus five	Error

Задача №452

Система счисления Фибоначчи

(Время: 1 сек. Память: 16 Мб Сложность: 51%)

Числа Фибоначчи U_1, U_2, \dots определяются начальными значениями и соотношением:

$$U_1=1; U_2=2; U_n=U_{n-1}+U_{n-2}.$$

Рассмотрим систему счисления с двумя цифрами 0 и 1, в которой, в отличие от двоичной системы, весами являются не степени двойки 1, 2, 4, 8, 16, ..., а числа Фибоначчи 1, 2, 3, 5, 8, 13, В этой системе счисления каждое положительное целое число единственным способом представляется в виде строки из нулей и единиц, которая начинается с 1 и в которой нет двух единиц, стоящих рядом.

Требуется написать программу, которая по двум заданным строкам, представляющим числа A и B в системе счисления Фибоначчи, находила строку, представляющую число A+B также в этой системе счисления.

Например, исходные строки 10101 и 100 представляют числа $1*8+0*5+1*3+0*2+1*1=8+3+1=12$ и $1*3+0*2+0*1=3$. Ответом является строка 100010, представляющая число $1*13+0*8+0*5+0*3+1*2+0*1=13+2=15=12+3$.

Входные данные

Входной файл INPUT.TXT в первой строке содержит число А и В во второй. Длина записи чисел А, В и их суммы А+В в системе счисления Фибоначчи не превышает 255 знаков.

Выходные данные

В выходной файл OUTPUT.TXT выведите значение суммы А+В. В случае неоднозначности достаточно вывести любое возможное решение.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	10101 100	100010

Задача №453 Раз-два, раз-два

(Время: 1 сек. Память: 16 Мб Сложность: 49%)

Для заданного натурального числа N требуется найти число, состоящее только из цифр 1 и 2, которое делилось бы на 2^N .

Входные данные

Входной файл INPUT.TXT содержит натуральное число N, не превосходящее 300.

Выходные данные

В выходной файл OUTPUT.TXT выведите искомое число, состоящее не более чем из 10 000 цифр.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	1	2
2	2	12

Задача №454 Оставшееся число

(Время: 1 сек. Память: 16 Мб Сложность: 45%)

Задан ряд последовательных натуральных чисел от М до N ($M < N$), из которого удаляют сначала все числа, стоящие на нечетных местах. Затем, из оставшегося ряда удаляют все числа, стоящие в нем на четных местах. Эти действия повторяют до тех пор, пока не останется одно число.

Требуется написать программу, которая находит оставшееся число.

Входные данные

Входной файл INPUT.TXT содержит число М в первой строке и число N во второй ($M < N <$

10⁹).

Выходные данные

В выходной файл OUTPUT.TXT выведите оставшееся число.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	1 4	2
2	2 9	7

Задача №455 Умножение дроби

(Время: 1 сек. Память: 16 Мб Сложность: 64%)

Задана некоторая правильная периодическая дробь Q и натуральное число N .

Требуется написать программу, которая определяет результат умножения Q на N , то есть непериодическую часть и минимальный период числа $Q \times N$. В случае получения результата умножения в виде конечной дроби скобки опускаются.

Входные данные

Входной файл INPUT.TXT в первой строке содержит число Q , во второй строке записано число N . В записи числа Q используется не более 100 цифр. При изображении дроби Q периодическая часть заключается в круглые скобки. N – натуральное число, не превосходящее 1000.

Выходные данные

В выходной файл OUTPUT.TXT выведите результат умножения Q на N .

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	0.1(6) 3	0.5
2	12.(3) 2	24.(6)

Задача №456 Сундук Билли Бонса

(Время: 1 сек. Память: 16 Мб Сложность: 30%)

Билли Бонс положил в сундук некоторое количество золотых монет. На второй год он вынул из сундука сколько-то монет. Начиная с третьего года, он добавлял столько монет, сколько было в сундуке два года назад.

Требуется написать программу, которая определит, сколько монет было в сундуке в первый и во второй года, если в X -м году там оказалось ровно Y монет.

Входные данные

Входной файл INPUT.TXT содержит натуральные числа X и Y ($3 \leq X \leq 20$, $1 \leq Y \leq 32767$).

Выходные данные

В выходной файл OUTPUT.TXT выведите через пробел количество монет в первый и второй года. Гарантируется, что решение задачи всегда существует.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	6 25	5 2

Пояснение к примеру

Если в первый год положить 5 монет, а во второй год вынуть 3 монеты, то начиная с первого года в сундуке будет 5, 2, 7, 9, 16, 25, ... монет.

Задача №457 Постоянная Капрекара

(Время: 1 сек. Память: 16 Мб Сложность: 22%)

Возьмем четырехзначное число, в котором не все цифры одинаковы, например 6264. Расположим цифры сначала в порядке убывания - 6642; затем, переставив их в обратном порядке, получим 2466. Вычтем последнее число из 6642. На следующем шаге с полученной разностью проделаем тоже самое. Через несколько таких действий получится число, переходящее само в себя и называемое постоянной Капрекара.

Требуется написать программу, которая находит эту постоянную и количество шагов для ее получения из заданного четырехзначного числа.

Входные данные

Входной файл INPUT.TXT содержит одну строку, в которой записано четырехзначное число.

Выходные данные

В выходной файл OUTPUT.TXT записываются: в первой строке постоянная Капрекара, во второй – количество шагов для ее получения.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	1234	6174 3

Задача №458 Шифровка - 2

(Время: 1 сек. Память: 16 Мб Сложность: 35%)

Для кодирования сообщения используют следующие действия: сообщение записывают, опуская

пробелы, в прямоугольник заданной высоты по столбцам, а затем прочитывают строки в заданном порядке.

1 П Р И А
2 Р А Р Н
3 О М О И
4 Г М В Е

а затем, если выбрать порядок строк 3, 1, 2, 4, получают закодированное сообщение ОМОИПРИАРАРНГМВЕ.

Требуется написать программу, которая по заданным высоте прямоугольника и порядке прочтения строк при кодировке декодирует заданное сообщение.

Входные данные

Входной файл INPUT.TXT содержит: в первой строке высоту прямоугольника (не больше 10), во второй – порядок прочтения строк (числа записаны через пробел), в третьей – закодированное сообщение, длина которого не превышает 200.

Выходные данные

В выходной файл OUTPUT.TXT записывается декодированное сообщение.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	4 3 1 2 4 ОМОИПРИАРАРНГМВЕ	ПРОГРАММИРОВАНИЕ
2	2 2 1 ииафзк	физика

Задача №459

Нить Ариадны

(Время: 1 сек. Память: 16 Мб Сложность: 40%)

Тезею из лабиринта Минотавра помог выйти клубок ниток. Вы можете вместо клубка использовать персональный компьютер.

Требуется написать программу, которая вводит маршрут Тезея в лабиринте и находит кратчайший обратный путь, по которому Тезей сможет выйти из лабиринта, не заходя в тупики и не делая петель.

Входные данные

Входной файл INPUT.TXT содержит маршрут Тезея, который представлен строкой, состоящей из букв: N, S, W, E и длиной не более 200.

Буквы означают:

- N - один "шаг" на север,
- S - один "шаг" на юг,
- W - один "шаг" на запад,

Е - один "шаг" на восток.

Выходные данные

В выходной файл OUTPUT.TXT записывается аналогично входному файлу найденный обратный путь. Если маршрут неоднозначен, то следует выбирать согласно следующему приоритету: N, E, S, W.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	EENNESWSSWE	NWW

Задача №460 Пятерки

(Время: 1 сек. Память: 16 Мб Сложность: 57%)

Задано натуральное N. Требуется написать программу, которая определит количество цифр 5 в записи всех натуральных чисел от 1 до N включительно.

Входные данные

Во входном файле INPUT.TXT в единственной строке задается число $N \leq 10^{18}$.

Выходные данные

В выходной файл OUTPUT.TXT записывается найденное число – количество пятерок.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	27	3
2	120	22

Задача №461 Выборы

(Время: 1 сек. Память: 16 Мб Сложность: 26%)

В одном из государств все решения традиционно принимались простым большинством голосов на общем собрании граждан, которых, к счастью, было не очень много. Одна из местных партий, стремясь прийти к власти как можно более законным путем, смогла добиться некоторой реформы избирательной системы. Главным аргументом было то, что население острова в последнее время значительно возросло, и проведение общих собраний перестало быть легкой задачей.

Суть реформы состояла в следующем: с момента введения ее в действие все избиратели острова делились на K групп (необязательно равных по численности). Голосование по любому вопросу теперь следовало проводить отдельно в каждой группе, причем считалось, что группа высказывается "за", если "за" голосует более половины людей в этой группе, а в противном случае считалось, что группа высказывается "против". После проведения голосования в группах подсчитывалось количество групп, высказавшихся "за" и "против", и вопрос решался положительно в том и только том случае, когда групп, высказавшихся "за", оказывалось более половины общего количества групп.

Эта система вначале была с радостью принята жителями острова. Когда первые восторги рассеялись, очевидны стали, однако, некоторые недостатки новой системы. Оказалось, что сторонники партии, предложившей систему, смогли оказать некоторое влияние на формирование групп избирателей. Благодаря этому, они получили возможность проводить некоторые решения, не обладая при этом реальным большинством голосов.

Пусть, например, на острове были сформированы три группы избирателей, численностью в 5, 5 и 7 человек соответственно. Тогда партии достаточно иметь по три сторонника в каждой из первых двух групп, и она сможет провести решение всего 6-ю голосами "за", вместо 9-и, необходимых при общем голосовании.

Требуется написать программу, которая по заданному разбиению избирателей на группы определит минимальное количество сторонников партии, достаточное для принятия любого решения.

Входные данные

Входной файл INPUT.TXT состоит из двух строк. В первой строке записано натуральное число $K < 1001$ - количество групп избирателей. Во второй строке через пробел записаны K натуральных чисел, которые задают количество избирателей в группах. Население острова не превосходит 30000 человек.

Выходные данные

В выходной файл OUTPUT.TXT выведите ответ на задачу.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	3 5 5 7	6
2	5 4 2 1 3 7	5

Задача №462 Числа

(Время: 1 сек. Память: 16 Мб Сложность: 36%)

Задано натуральное число N . Требуется написать программу, которая находит количество натуральных чисел, не превышающих N и не делящихся ни на одно из чисел 2, 3, 5.

Входные данные

Входной файл INPUT.TXT содержит число N ($1 \leq N \leq 10^9$).

Выходные данные

В выходной файл OUTPUT.TXT выведите ответ на задачу.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	10	2

2	20	6
---	----	---

Задача №463
Остаток

(Время: 1 сек. Память: 16 Мб Сложность: 44%)

Дано натуральное число n в k -ичной системе счисления.
Требуется написать программу, которая находит остаток от деления этого числа на заданное натуральное число m .

Входные данные

Входной файл INPUT.TXT содержит в первой строке натуральные числа k и m ($2 \leq k \leq 36, m \leq 32767$), записанные через пробел. Во второй строке задается натуральное число n в k -ой системе счисления, состоящее не более чем из 10000 цифр. Числа k и m записываются в десятичной системе счисления. Для записи цифр числа в системе счисления с основанием, большим 10, используются латинские заглавные буквы (10 – A, 11 – B, ..., 35 – Z).

Выходные данные

В выходной файл OUTPUT.TXT выведите искомое число, записанное в десятичной системе счисления.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	10 2 17	1
2	16 35 7DF99	23

Задача №464
Число в последовательности

(Время: 1 сек. Память: 16 Мб Сложность: 35%)

Последовательность 011212201220200112... строится следующим образом: сначала пишется 0, затем повторяется следующее действие: уже написанную часть приписывают справа с заменой 0 на 1, 1 на 2, 2 на 0, и т.д.
Требуется написать программу, которая по заданному натуральному числу N определяет, какое число стоит на N -ом месте.

Входные данные

Входной файл INPUT.TXT содержит число N ($1 \leq N \leq 2147483647$).

Выходные данные

Выходной файл OUTPUT.TXT должен содержать одно искомое число.

Примеры

--	--	--

№	INPUT.TXT	OUTPUT.TXT
1	1	0
2	10	2

Задача №465

Последовательности из 0 и 1

(Время: 1 сек. Память: 16 Мб Сложность: 47%)

Рассмотрим последовательности длины N, состоящие из 0 и 1. Требуется написать программу, которая по заданному натуральному числу N определяет количество тех из них, в которых никакие две единицы не стоят рядом.

Входные данные

Входной файл INPUT.TXT содержит число N ($1 \leq N \leq 1000$).

Выходные данные

В выходной файл OUTPUT.TXT выведите ответ на задачу.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	2	3
2	3	5

Задача №466

Функция

(Время: 1 сек. Память: 16 Мб Сложность: 42%)

Функция $f(n)$ определена следующим образом:

$f(0)=0, f(1)=1, f(2n)=f(n), f(2n+1)=f(n)+f(n+1).$

Требуется написать программу, которая по заданному натуральному числу N определяет значение функции $f(N)$.

Входные данные

Входной файл INPUT.TXT содержит число N ($1 \leq N \leq 2147483647$).

Выходные данные

В выходной файл OUTPUT.TXT выведите значение $f(N)$.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	2	1
2	5	3

Задача №467
Закраска прямой - 2

(Время: 1 сек. Память: 16 Мб Сложность: 61%)

Определение. Интервал прямой с целочисленными координатами [a, b) содержит левую границу – точку a и не содержит правую границу – точку b.

Интервал от 0 до 10^9 выкрасили в белый цвет. Затем было выполнено N операций перекрашивания. При каждой операции цвета в интервале, границы которого задаются, меняются на противоположный (белый на черный, черный на белый).

Требуется написать программу, которая найдет самый длинный интервал белого цвета после заданной последовательности операций перекрашивания.

Входные данные

Входной файл INPUT.TXT содержит в первой строке число N ($1 \leq N \leq 500$) и затем N строк с границами интервалов (числа в диапазоне от 0 до 10^9).

Выходные данные

В выходной файл OUTPUT.TXT выведите одно число – длину самого большого белого интервала.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	4 20 50 10 35 40 90 100 1000000000	15

Задача №468
Числовая последовательность - 2

(Время: 1 сек. Память: 16 Мб Сложность: 66%)

Дана последовательность натуральных чисел 7, 11, 13, 14, 19, 21, 22, 25,

Требуется написать программу, которая по заданному N находит N-ый член этой последовательности.

Входные данные

Входной файл INPUT.TXT содержит число N ($1 \leq N \leq 2147483647$).

Выходные данные

В выходной файл OUTPUT.TXT выведите N-ый член последовательности.

Примеры

№	INPUT.TXT	OUTPUT.TXT

1	1	7
2	6	21

Задача №469

Химическая тревога

(Время: 1 сек. Память: 16 Мб Сложность: 50%)

Произошло радиоактивное заражение местности. Составлена карта зараженности. Она представляет собой прямоугольную таблицу N*M, в клетках которой записана зараженность соответствующего участка.

Требуется написать программу, которая найдет путь из левой верхней клетки таблицы в правую нижнюю клетку с минимальной суммарной дозой радиации.

Входные данные

Входной файл INPUT.TXT содержит в первой строке числа N и M, а в следующих N строках – по M чисел – карта зараженности местности. Числа в строках разделяются одним пробелом. 1<=N<=30, 1<=M<=30, зараженность участка – целое число от 0 до 100.

Выходные данные

Выходной файл OUTPUT.TXT должен содержать одно число – суммарную долю радиации.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	3 5 2 100 0 100 100 1 100 0 0 0 1 0 3 100 2	9

Задача №470

Земельный комитет

(Время: 2 сек. Память: 16 Мб Сложность: 41%)

Земельный комитет города принял решение о сдаче в аренду части муниципальной территории, имеющей форму прямоугольника размером N на W километров. Стоимость аренды каждого квадратного участка 1 x 1 км была определена с учётом локальных условий, и занесена в таблицу.

С целью организации открытого тендера на аренду, земельный комитет решил выставить на своём веб-сайте карту территории, и предоставить посетителям возможность узнавать суммарную стоимость аренды для произвольной прямоугольной группы соседних участков.

Данное предложение вызвало большой интерес у населения и предпринимателей, и нагрузка на сервер очень высока.

Требуется написать программу, позволяющую как можно более эффективно рассчитывать стоимость аренды для N запросов. В каждом запросе требуется определить общую стоимость участков внутри прямоугольной группы с противоположными углами, расположенными в элементах таблицы (ai, bi) и (ci, di).

Входные данные

В первой строке входного файла INPUT.TXT находятся числа H, W, N ($1 \leq H, W \leq 100, 1 \leq N \leq 1000000$). В следующих H строках содержится по W чисел (стоимости участков находятся в диапазоне от 0 до 10000). Далее идут N строк с числами $a_i b_i c_i d_i$ ($1 \leq a_i \leq c_i \leq H, 1 \leq b_i \leq d_i \leq W$).

Выходные данные

В выходной файл OUTPUT.TXT должен содержать N чисел, по одному числу в строке.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	2 3 1 5 1 2 6 7 3 2 1 2 3	16

Задача №471 Ход конем

(Время: 1 сек. Память: 16 Мб Сложность: 53%)

Шахматная ассоциация решила оснастить всех своих сотрудников такими телефонными номерами, которые бы набирались на кнопочном телефоне ходом коня. Например, ходом коня набирается телефон 340-49-27. При этом телефонный номер не может начинаться ни с цифры 0, ни с цифры 8.

Требуется написать программу, определяющую количество телефонных номеров длины N, набираемых ходом коня.

1	2	3
4	5	6
7	8	9
	0	

Входные данные

Входной файл INPUT.TXT содержит натуральное число N ($N \leq 100$).

Выходные данные

В выходной файл OUTPUT.TXT выведите искомое количество телефонных номеров.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	1	8
2	2	16

Задача №472 Подарки

(Время: 1 сек. Память: 16 Мб Сложность: 39%)

Приближался Новый год и отец купил своим детям по подарку. Оказалось, что в них разное количество конфет. Тогда отец купил еще конфет и стал их раскладывать по подаркам следующим образом: брал один из подарков с наименьшим количеством конфет и добавлял в него одну

конфету.

Требуется написать программу, которая найдет наименьшее количество конфет, оказавшихся в одном из подарков после завершения раскладывания всех конфет.

Входные данные

Входной текстовый файл INPUT.TXT содержит в первой строке N – количество детей и M – количество купленных конфет. Числа записаны через пробел, $1 \leq N \leq 10\,000$, $1 \leq M \leq 1\,000\,000$. Далее в N строках записаны числа в диапазоне от 1 до 30000 – количество конфет в подарках.

Выходные данные

Выходной файл OUTPUT.TXT должен содержать одно найденное число.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	2 4 1 2	3

Задача №473 Автомобильные пробки

(Время: 1 сек. Память: 16 Мб Сложность: 48%)

Автомобильные пробки случаются везде, даже в нашем небольшом городе. Дороги у нас имеют по две полосы в одном направлении, а автомобили только двух видов: легковые (в пробке занимают квадратное место 1x1 от ширины одной полосы) и грузовые (занимают прямоугольное место 1x2). Автомобилисты очень дисциплинированы: не становятся поперек полосы, не занимают чужую площадь, но и не оставляют свободных мест.

Требуется написать программу, которая определит количество различных автомобильных пробок длины N.

Входные данные

Входной файл INPUT.TXT содержит одно натуральное число N ($N \leq 1000$).

Выходные данные

Выходной файл OUTPUT.TXT должен содержать найденное количество автомобильных пробок.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	2	4
2	3	9

Задача №474 Последовательность Кеане

(Время: 1 сек. Память: 16 Мб Сложность: 57%)

Бесконечная последовательность битов, предложенная Кеане, равна 001001110001001110110110001... и формируется следующим алгоритмом: вначале записывается 0, потом 001, далее 001001110, то есть, для получения следующего члена, предыдущий записывается дважды, а справа приписывается его отрицание. Элементы этого ряда являются начальными подпоследовательностями Кеане.

Требуется написать программу, которая по заданному n определит N-й бит этой последовательности.

Входные данные

Входной файл INPUT.TXT содержит число N ($N \leq 10^{200}$).

Выходные данные

В выходной файл OUTPUT.TXT должен содержать найденный бит.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	3	1
2	18	0

Задача №475
Арифметическая прогрессия - 2

(Время: 1 сек. Память: 16 Мб Сложность: 38%)

Задана последовательность натуральных чисел из диапазона [1, 2147483647]. Количество чисел в этой последовательности не превышает 100 000. Необходимо определить: можно ли выстроить эти числа в отрезок арифметической прогрессии. При необходимости порядок чисел в последовательности можно изменять.

Требуется написать программу для решения этой задачи.

Входные данные

Входной файл INPUT.TXT содержит заданную последовательность натуральных чисел. Числа в файле разделены пробелами или символами перехода на новую строку.

Выходные данные

Выходной файл OUTPUT.TXT должен содержать либо «Yes» в случае положительного ответа, либо «No» в противном случае.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	80 50 10 30 70 40 20 60 90	Yes
2	1 2 3 5	No

Задача №476

Ферзя в угол!

(Время: 1 сек. Память: 16 Мб Сложность: 43%)

Рассмотрим бесконечную вправо и вверх шахматную доску, на которой стоит ферзь. Двое по очереди двигают этого ферзя. Разрешается двигать ферзя только вниз, влево или по диагонали вниз влево. Цель игры – задвинуть ферзя в угол, то есть клетку с координатами (1, 1). На рисунке показаны разрешенные движения ферзя.

Требуется написать программу, которая найдет номер игрока, который выиграет при правильной игре.

Входные данные

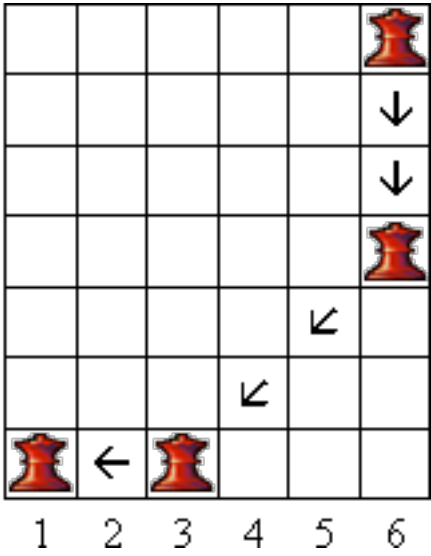
Входной файл INPUT.TXT содержит координаты ферзя перед первым ходом - два числа М и N, записанные через пробел (1 ≤ М, N ≤ 250).

Выходные данные

Выходной файл OUTPUT.TXT должен содержать найденный номер победителя.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	3 2	2
2	6 7	1



Задача №477 Простая игра

(Время: 1 сек. Память: 16 Мб Сложность: 50%)

Дед Мазай и заяц играют в очень простую игру. Перед ними – огромная куча из N одинаковых морковок. Каждый из них во время своего хода может взять из этой кучи любое количество морковок, равное неотрицательной степени числа 2, т.е. 1, 2, 4, 8,... . Начинает игру либо дед Мазай, либо заяц. Затем игроки ходят по очереди. Тот, кто возьмет последнюю морковку, тот и выигрывает.

Требуется написать программу, которая при заданных исходных данных определяет победителя в этой игре. При этом следует учитывать, что игроки играют оптимально.

Входные данные

Входной файл INPUT.TXT содержит единственное целое положительное число N ($N \leq 10^{250}$), задающее число морковок в начале игры.

Выходные данные

Выходной файл OUTPUT.TXT должен содержать в первой строке цифру «1», если выиграет тот, кто ходит первым, или цифру «2» – в противном случае. Если игру выиграл тот, кто ходил первым, то во второй строке этого файла должно содержаться минимальное число морковок, которое должен взять игрок, выполнявший ход первым, чтобы гарантировать свою победу.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	3	2
2	8	1 2

Задача №478

Коррозия металла

(Время: 1 сек. Память: 16 Мб Сложность: 55%)

Для хранения двух агрессивных жидкостей А и В используется емкость с многослойной перегородкой, которая изготавливается из имеющихся N листов. Для каждого листа i (i = 1, ..., N) известно время его растворения жидкостью А — a_i и жидкостью В — 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.

Входные данные

В первой строке входного файла INPUT.TXT записано число N (1 <= N <= 256). В каждой из последующих N строк содержатся два положительных вещественных числа a_i и b_i, разделенные пробелом.

Выходные данные

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 А к жидкости В, разделяя числа пробелами.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	4	6.000
	1 2	4 2 1 3
	1 2	
	0.5 1.5	
	7 3.5	

Задача №479

Муравей и дерево

(Время: 1 сек. Память: 16 Мб Сложность: 63%)

Муравей находится в лесу с плоской поверхностью почвы в точке с координатами (x1, y1) и направляется в точку (x2, y2). В лесу растет дерево, основание ствола которого имеет форму круга с центром в точке (x, y) и радиусом r. Дерево, возможно, мешает муравью дойти до цели по прямой. В таком случае ему придется обойти дерево вокруг ствола.

Требуется написать программу, которая определит длину кратчайшего пути муравья.

Входные данные

Входной файл INPUT.TXT содержит вещественные числа $x_1, y_1, x_2, y_2, x, y, r$. Числа записаны через пробел и находятся в диапазоне от 0 до 1000, $r > 0$. Начальная и конечная точки пути муравья не могут находиться внутри круга.

Выходные данные

Выходной файл OUTPUT.TXT должен содержать единственное вещественное число – длину кратчайшего пути. Абсолютная ошибка результата не должна превосходить 0.01 (т.е. следует выводить число с точностью не менее 3 знаков после запятой).

Пример

№	INPUT.TXT	OUTPUT.TXT
1	0 0 4 4 2 2 1	6.014

Задача №480

Игра в монеты

(Время: 1 сек. Память: 16 Мб Сложность: 47%)

Гриша и Дима играют в следующую игру: они разложили однокопеечные монетки в стопки (в разных стопках может быть различное количество монет), а стопки расположили на столе перед собой в ряд слева направо. Затем Гриша и Дима по очереди делают ходы. На каждом ходе один из игроков берет слева несколько стопок, не меньше одной, но и не больше, чем перед этим взял его соперник. Первый игрок своим первым ходом берет не более K стопок. Игра заканчивается, когда стопок не остается.

Требуется написать программу, позволяющую вычислить, какое максимальное число монет может получить первый участник после окончания игры, если второй – тоже старается ходить так, чтобы получить как можно больше монет.

Входные данные

Входной файл INPUT.TXT состоит из одной строки, в которой записаны: число стопок N ($1 \leq N \leq 180$), за ним идут N чисел, задающих количество монет в стопках слева направо (количество монет в стопке — не менее 1 и не более 20000), а затем число K , ограничивающее количество стопок, которые первый игрок может взять на первом ходе ($1 \leq K \leq 80$). Все числа в строке разделены пробелом.

Выходные данные

В выходной файл OUTPUT.TXT необходимо вывести одно число — максимальное количество монет, которое заведомо может получить первый игрок, как бы ни играл второй.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	3 4 9 1 3	14
2	4 2 2 1 7 3	5
3	5 3 4 8 1 7 2	18

Задача №481

Количество палиндромов

(Время: 1 сек. Память: 16 Мб Сложность: 51%)

Непустая строка, содержащая некоторое слово, называется палиндромом, если это слово одинаково читается как слева направо, так и справа налево. Пусть задана строка, в которой записано слово S, состоящее из N прописных букв латинского алфавита. Путем вычеркивания из этого слова некоторого набора символов, можно получить строку, которая будет палиндромом.

Требуется написать программу, с помощью которой можно определить, сколько существует способов вычеркивания из заданного слова некоторого (возможно пустого) набора символов, чтобы образованная таким образом строка была палиндромом. Способы, отличающиеся порядком вычеркивания символов, считаются одинаковыми.

Входные данные

В первой и единственной строке входного файла INPUT.TXT записано слово S, состоящее из N символов (1 <= N <= 30).

Выходные данные

В выходной файл OUTPUT.TXT выведите найденное число способов.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	AAA	7
2	BAOBAV	22

Задача №482
Короткая последовательность

(Время: 1 сек. Память: 16 Мб Сложность: 28%)

Дано целое число N. Рассмотрим последовательность S₁S₂S₃...S_k..., где каждая группа цифр S_k состоит из записанных одно за другим чисел от 1 до k. Например, первые 75 цифр последовательности выглядят так:

112123123412345123456123456712345678123456789123456789101234567891011123456.

Требуется написать программу, которая определит: какая цифра находится на N-ой позиции в построенной последовательности.

Входные данные

Входной файл INPUT.TXT содержит одно число N (0 < N < 32768).

Выходные данные

В выходной файл OUTPUT.TXT выведите цифру, которая стоит на N-ой позиции в последовательности.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	3	2
2	20	5

Задача №483

Двоичная машина

(Время: 1 сек. Память: 16 Мб Сложность: 63%)

На вход некоторой двоичной машине подается n-разрядное двоичное число. Машина подвергает поданное число следующим преобразованиям:

- 1. Записывает исходное число в регистры А и В размером N двоичных разрядов (бит).
- 2. N-1 раз выполняет следующие действия: циклически сдвигает регистр В на один бит влево (при этом самый старший бит переходит в самый младший); прибавляет к значению регистра А значение регистра В по модулю 2^N .
- 3. К регистру А применяется побитовая операция логического отрицания
- 4. К регистру А прибавляется 1 по модулю 2^N .
- 5. Значение регистра А выдается в качестве результата с отбрасыванием ведущих (незначащих) нулей.

Например, если на вход машине подать число 1001, то будут выполнены следующие преобразования:

№ шага	A	B	Result
1.	1001	1001	
2. (повтор. 1)	1100	0011	
2. (повтор. 2)	0010	0110	
2. (повтор. 3)	1110	1100	
3.	0001	1100	
4.	0010	1100	
5.	0010	1100	10

Напишите программу, которая по заданному числу возвращает результат преобразования согласно описанному алгоритму работы двоичной машины.

Входные данные

Первая строка входного файла INPUT.TXT содержит число N – количество двоичных разрядов ($N \leq 100\,000$). Вторая строка содержит ровно N двоичных разрядов (нулей или единиц) – входное число.

Выходные данные

В выходной файл OUTPUT.TXT выведите ответ – результат преобразования в двоичном виде без ведущих нулей.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	4 1001	10
2	7 1101101	101

Задача №484

Трубопровод

(Время: 1 сек. Память: 16 Мб Сложность: 60%)

В некоторой корпорации при производстве различных химических веществ используется сложная система трубопроводов для слива отходов. При монтаже трубопровода использовались стандартные элементы, состоящие из двух задвижек, двух входов и одного выхода (рис. 1).

Такие элементы объединяются вместе, как показано на рисунке 2, где изображен трубопровод до 4 уровня. Здесь обозначение Layer – указывает на слой (уровень) задвижек.

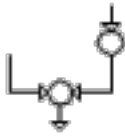


Рис. 1

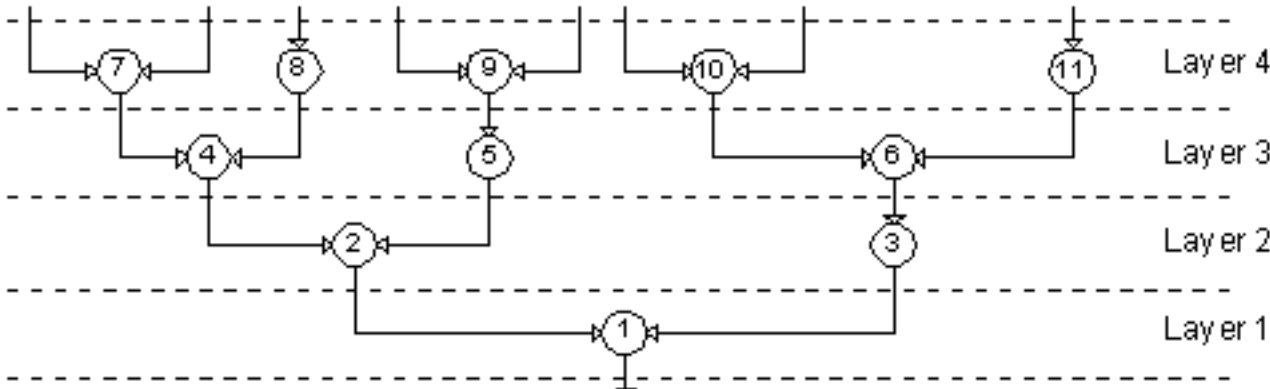


Рис. 2

В последующем ограничимся рассмотрением трубопровода до сорокового слоя включительно. Все задвижки в трубопроводе пронумерованы от первого до сорокового слоя. Внутри каждого слоя задвижки пронумерованы слева направо (см. рис. 2).

Напишите программу, которая по заданному порядковому номеру S начальной задвижки определяет и выводит в выходной файл номера всех тех задвижек, которые следует открыть, чтобы осуществить сброс отходов.

Входные данные

Входной файл INPUT.TXT содержит одно целое число S – номер начальной задвижки ($1 \leq S \leq 433494435$).

Выходные данные

Выходной файл OUTPUT.TXT должен содержать последовательность номеров задвижек. Последовательность должна начинаться с числа S и заканчиваться числом 1. Элементы последовательности идут в порядке убывания, разделяются пробелами и (или) символами конца строк.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	8	8 4 2 1
2	5	5 2 1
3	6	6 3 1
4	10	10 6 3 1

Задача №485

Рыбаки

(Время: 1 сек. Память: 16 Мб Сложность: 29%)

Говорят, что однажды Дираку предложили решить следующую задачу.

Трое рыбаков отправились на рыбалку. Наловив рыбы, они устроились на ночлег. Проснувшись, один из рыбаков решил взять свою долю (третью часть) и отправиться домой. Пересчитав рыбу, он заметил, что на троих выловленная рыба не делится – остается одна лишняя. Недолго думая, первый рыбак выкинул лишнюю рыбу в море, взял свою долю и отправился домой.

Затем проснулся второй рыбак. Не заметив, что один из его товарищей уже уехал, он тоже стал делить оставшуюся рыбу на троих, и у него тоже получилась одна лишняя рыба. Поступив с лишней рыбой, также как и первый рыбак, он забрал долю, которую считал своей и поехал домой.

Третий рыбак также не обнаружил исчезновения своих товарищей и тоже стал делить рыбу на троих. И у него получилась одна лишняя рыба, которую он выкинул, после чего забрал третью часть и отбыл домой.

Вопрос, заданный Дираку, звучал так: сколько рыбы поймали рыбаки?

Обобщим данную задачу следующим образом.

Однажды N рыбаков отправились на рыбалку, где поймали X рыб. После этого рыбаки легли спать. Утром, просыпаясь друг за другом, каждый из них делил выловленную рыбу на N частей. Каждый раз в остатке оставалось ровно K рыб ($0 < K < N$). Эти K рыб выбрасывались обратно в море.

Ваша задача – определите при заданных N и K минимально возможное целое положительное значение X – число рыб, удовлетворяющее условию задачи.

Входные данные

Входной файл INPUT.TXT содержит два целых числа N и K , разделенные пробелом ($2 \leq N \leq 8$, $0 < K < N$).

Выходные данные

Выходной файл OUTPUT.TXT должен содержать одно целое положительное число X – наименьшее возможное количество выловленной рыбаками рыбы.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	3 1	25
2	4 3	247

Задача №486

Рыбаки - 2

(Время: 1 сек. Память: 16 Мб Сложность: 53%)

Однажды N рыбаков отправились на рыбалку, где поймали X рыб. После этого рыбаки легли спать. Утром, просыпаясь друг за другом, каждый из них делил выловленную рыбу на N частей. Каждый раз в остатке оставалось ровно K рыб ($0 < K < N$). Эти K рыб выбрасывались обратно в море.

Ваша задача – определите при заданных N и K минимально возможное целое положительное

значение X – число рыб, удовлетворяющее условию задачи.

Входные данные

Входной файл INPUT.TXT содержит два целых числа N и K , разделенные пробелом ($2 \leq N \leq 15$, $0 < K < N$).

Выходные данные

Выходной файл OUTPUT.TXT должен содержать одно целое положительное число X – наименьшее возможное количество выловленной рыбаками рыбы.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	3 1	25
2	4 3	247

Задача №487 Игра "Баше"

(Время: 1 сек. Память: 16 Мб Сложность: 49%)

Студент и профессор играют в игру «Баше» по следующим правилам.

На столе лежат N экзаменационных билетов. Игроки делают ходы поочередно, и в свой ход каждый из игроков может взять от 1 до K билетов. Выигрывает тот игрок, который возьмет билет, оставшийся последним. Т.е. если его возьмет студент, то он получит «зачет», иначе он получит «незачет».

Будем называть сделанный ход ошибочным, если в этой ситуации можно было сходить иначе, гарантируя себе в дальнейшем выигрыш независимо от игры соперника. Будем называть ход правильным (или допустимым), если он не является ошибочным.

Ваша задача – проанализировать уже сыгранную партию и указать для каждого хода, был ли он правильным или ошибочным.

Входные данные

В первой строке входного файла INPUT.TXT записаны три целых числа: N , K , P ($2 \leq N \leq 10000$, $2 \leq K \leq 100$, $2 \leq P$). Здесь P – количество ходов, которые сделали студент и профессор. В последующих P строках записаны числа (по одному числу на строке) в диапазоне от 1 до K .

Выходные данные

Выходной файл OUTPUT.TXT должен содержать P строк по одному символу на строке: «Т» (правильный или допустимый ход – от слова True) или «F» (неверный ход – от слова False).

Пример

№	INPUT.TXT	OUTPUT.TXT
1	10 5 3 3 3 4	F F T

Задача №488
Праздничный торт

(Время: 1 сек. Память: 16 Мб Сложность: 58%)

На дне рождения Пети на стол подали круглый торт, украшенный разноцветными розочками. Причем, на торте было ровно по две небольших розочки каждого цвета, а различных цветов было ровно K . Тем самым, общее число розочек было равно $2 \cdot K$.

Одноклассник Пети Вася, славящийся своими математическими способностями, разрезал торт прямолинейными разрезами так, что каждый разрез прошел через две розочки одинакового цвета. (Понятно, что таких разрезов оказалось ровно K .)

При этом оказалось, что:

- никакие три разреза не прошли через одну точку,
- никакие два разреза не совпали и не оказались параллельными.

Напишите программу, которая по заданному радиусу торта, и координатам розочек определяет, на сколько частей был разделен торт.

Входные данные

В первой строке входного файла INPUT.TXT содержатся два числа: действительное число R (радиус торта) и целое число K – количество различных цветов ($1 \leq K \leq 100$). В каждой из последующих K строк записано по четыре числа, разделенных пробелами: X_1, Y_1, X_2, Y_2 – координаты двух розочек каждого цвета. Центр торта находится в начале координат. Радиус торта и координаты розочек – действительные числа, не превышающие по абсолютной величине 100.

Выходные данные

В выходной файл OUTPUT.TXT выведите целое число – количество частей, на которые был разрезан торт.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	3 1 -1 -1 1 1	2
2	5 3 1 4 1 -3 -3.5 0 4 1.5 -3 3 0 -4	6

Задача №489
Монеты

(Время: 1 сек. Память: 16 Мб Сложность: 40%)

На столе лежат произвольным образом N сказочных монет: M_1, M_2, \dots, M_n . Ученик Паша проводит следующий эксперимент. Он K раз выполняет следующие действия: берет некоторую монету, переворачивает ее, кладет ее в произвольное свободное место на столе и записывает номинал этой монеты в протокол.

После этого Паша закрывает одну монету (любую) и предлагает Маше угадать номинал и

верхнюю сторону закрытой монеты.

Маше доступны следующие исходные данные:

- начальный набор монет,
- набор монет на столе после эксперимента,
- протокол переворачиваний.

Пусть Паша положил на стол монеты (N=8): 10, −5, 1, 50, 1, 100, −1, 9. Здесь абсолютная величина числа соответствует номиналу монеты, а знаком «−» помечены монеты, положенные орлом вверх.

Пусть в протоколе записано 7 ходов (K=7): 10, 10, 5, 10, 50, 100, 1 (здесь номиналы указываются без знака).

После того, как Паша закрыл одну монету, на столе остались монеты (перечисляются в произвольном порядке): −100, −10, −50, 1, 5, 1, 9.

Подумав некоторое время, Маша сказала, что закрыта монета номиналом 1 решкой вверх.

Напишите программу, которая по известным N и K определяет закрытую монету и ее верхнюю сторону. Если вам потребуется, можно дополнительно использовать протокол переворачиваний, который также будет задан во входном файле.

Входные данные

В первой строке входного файла INPUT.TXT записаны два целых числа: N и K – количество монет на столе и количество переворачиваний ($1 \leq N \leq 10000$, $1 \leq K \leq 100000$). Во второй строке записаны N целых чисел (монеты в начале эксперимента). В третьей строке записано N-1 целое число (монеты в конце эксперимента). Последующие одна или несколько строк содержат K целых положительных чисел (номиналы тех монет, которые переворачиваются), разделенных пробелами или признаком конца строки. Условные номиналы монет заданы целыми числами в диапазоне от 1 до 1000.

Выходные данные

В выходном файле OUTPUT.TXT должно быть записано одно целое число – номинал закрытой монеты либо без знака (если монета лежит решкой вверх), либо со знаком «−», если монета лежит орлом вверх.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	8 7 10 −5 1 50 1 100 −1 9 −100 −10 −50 1 5 1 9 10 10 5 10 50 100 1	1

Задача №490 Дни рождения

(Время: 1 сек. Память: 16 Мб Сложность: 27%)

Два одноклассника Петя и Вася родились не ранее 1993 и не позднее 1994 года, причем, Петя старше Васи.

Напишите программу, которая по заданным дням рождения определяет: на сколько дней Петя

старше Васи.

Заметим, что 1993 и 1994 года не являются високосными, т.е. в феврале в них ровно 28 дней.

Входные данные

Входной файл INPUT.TXT содержит дату рождения Пети в первой строке и дату рождения Васи во второй. Даты заданы в формате «ДД.ММ.ГГ», например, строка 06.02.93 означает дату рождения 6 февраля 1993 года.

Выходные данные

В выходной файл OUTPUT.TXT выведите единственное число – искомое количество дней.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	01.01.93 02.01.93	1
2	05.02.94 05.03.94	28

Задача №491 Антипалиндром

(Время: 1 сек. Память: 16 Мб Сложность: 28%)

Палиндромом называют строку, читающуюся одинаково с обеих сторон. Задана строка s. Найдите ее наибольшую по длине подстроку, не являющуюся палиндромом.

Входные данные

Входной файл INPUT.TXT содержит строку s. Она состоит только из строчных букв латинского алфавита, не пуста, а ее длина не превышает 100 000 символов.

Выходные данные

В выходной файл OUTPUT.TXT выведите ответ на задачу. Если все подстроки s являются палиндромами, выведите в выходной файл NO SOLUTION.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	abba	abb
2	abc	abc
3	aaaaa	NO SOLUTION

Задача №492 Сближение с целью

(Время: 1 сек. Память: 16 Мб Сложность: 37%)

Вы являетесь одним из разработчиков программного обеспечения боевой информационной системы для ракетного крейсера нового поколения РК-2000. Один из компонентов этой системы

отвечает за решение задач тактического маневрирования. В настоящее время вы занимаетесь решением задачи о сближении с целью.

Заданы координаты x_0 и y_0 цели в начальный момент времени, а также вектор $(V_x; V_y)$ ее скорости. Считается, что цель движется равномерно и прямолинейно. В начальный момент времени РК-2000 находится в начале координат. Его максимальная скорость равна V .

Необходимо выяснить, может ли РК-2000 через заданное время t оказаться ровно на заданном расстоянии d от цели. Для простоты считайте, что РК-2000 может мгновенно изменять свою скорость.

Входные данные

Первая строка входного файла INPUT.TXT содержит два числа x_0 и y_0 ($|x_0| \leq 10^9, |y_0| \leq 10^9$). Вторая строка входного файла содержит два числа V_x и V_y ($|V_x| \leq 10^6, |V_y| \leq 10^6$). Третья строка входного файла содержит три целых числа: V, t, d ($1 \leq V \leq 10^6, 1 \leq t \leq 10^3, 1 \leq d \leq 10^9$).

Выходные данные

В выходной файл OUTPUT.TXT выведите YES, если РК-2000 может через заданное время t оказаться ровно на заданном расстоянии d от цели, и NO – в противном случае.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	1 1 1 1 1 1 1	NO
2	1 1 1 1 1 1 2	YES

Задача №493 Морской бой - 2

(Время: 1 сек. Память: 16 Мб Сложность: 21%)

«Морской бой» - игра для двух участников, в которой игроки по очереди называют координаты на неизвестной им карте соперника. Если у соперника по этим координатам имеется корабль, то корабль или его часть «топится», а попавший получает право сделать еще один ход. Цель игрока - первым поразить все корабли противника.

«Морской бой» очень популярен среди учеников одной физико-математической школы. Ребята очень любят в него играть на переменах. Вот и сейчас ученики Иннокентий и Емельян начали новую партию.

Правила, по которым ребята расставляют корабли перед началом партии, несколько отличаются от классических. Во-первых, игра происходит на поле размером $N \times M$, а не 10×10 . Во-вторых, число кораблей, их размер и форма выбираются ребятами перед партией - так играть намного интереснее.

Емельян уже расставил все свои корабли, кроме одного однопалубного. Такой корабль занимает ровно одну клетку.

Задана расстановка кораблей Емельяна. Найдите число способов поставить оставшийся

однопалубный корабль. При этом учитывайте, что по правилам его можно ставить только в ту клетку, все соседние с которой не заняты. В этой задаче соседними считаются клетки, имеющие общую сторону.

Входные данные

Первая строка входного файла INPUT.TXT содержит два числа: N и M ($1 \leq N, M \leq 100$). Последующие N строк описывают игровое поле - каждая из них содержит M символов. Символом «.» (точка) обозначена свободная клетка, символом «*» (звездочка) - занятая кораблем.

Выходные данные

В выходной файл OUTPUT.TXT выведите ответ на задачу.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	4 4 * * * * * * . . * . . . * . . .	4
2	4 3 * * * * * *	0

Задача №494 Отрезок и окружности

(Время: 1 сек. Память: 16 Мб Сложность: 58%)

На плоскости задана система концентрических окружностей, центры которых находятся в начале координат, а радиусы равны 1, 2, 3, Также на плоскости задан отрезок, концы которого находятся в точках (x1, y1) и (x2, y2).

Необходимо найти число общих точек этого отрезка и указанной системы окружностей.

Входные данные

Входной файл INPUT.TXT содержит четыре целых числа: x1, y1, x2 и y2. Эти числа не превосходят 10^3 по абсолютной величине. Заданный отрезок имеет ненулевую длину.

Выходные данные

В выходной файл OUTPUT.TXT выведите одно целое число – количество общих точек.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	1 1 2 1	1
2	1 2 2 1	0

Задача №495

Двойственная ломаная

(Время: 1 сек. Память: 16 Мб Сложность: 35%)

Пусть задана некоторая замкнутая ломаная P . Двойственной для P ломаной называется ломаная, вершины которой находятся в серединах звеньев P .

Определим понятие двойственной для P ломаной k -ого порядка $\text{dual}(P, K)$ следующим образом:

- двойственной ломаной нулевого порядка является сама ломаная P ($\text{dual}(P, 0) = P$);
- если $K > 0$, то двойственной ломаной K -ого порядка является ломаная, двойственная для $\text{dual}(P, K - 1)$.

Задана замкнутая ломаная P и число K . Найдите длину двойственной для P ломаной k -ого порядка $\text{dual}(P, K)$.

Входные данные

Входной файл INPUT.TXT содержит число звеньев исходной ломаной N ($3 \leq N \leq 100$). Каждая из последующих N строк содержит по два целых числа x_i и y_i - координаты i -ой вершины ломаной. Все x_i и y_i не превосходят 10^5 по абсолютной величине. Последняя строка входного файла содержит целое число K ($0 \leq K \leq 10$).

Выходные данные

В выходной файл OUTPUT.TXT выведите искомую длину с точностью не хуже, чем 10^{-4} .

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	4 0 0 1 0 1 1 0 1 0	4
2	4 0 0 1 0 1 1 0 1 1	2.8284271247

Задача №496

Сбор черники

(Время: 1 сек. Память: 16 Мб Сложность: 17%)

В фермерском хозяйстве в Карелии выращивают чернику. Она растет на круглой грядке, причем кусты высажены только по окружности. Таким образом, у каждого куста есть ровно два соседних. Всего на грядке растет N кустов.

Эти кусты обладают разной урожайностью, поэтому ко времени сбора на них выросло

различное число ягод – на i-ом кусте выросло a_i ягод.

В этом фермерском хозяйстве внедрена система автоматического сбора черники. Эта система состоит из управляющего модуля и нескольких собирающих модулей. Собирающий модуль за один заход, находясь непосредственно перед некоторым кустом, собирает ягоды с этого куста и с двух соседних с ним.

Напишите программу для нахождения максимального числа ягод, которое может собрать за один заход собирающий модуль, находясь перед некоторым кустом заданной во входном файле грядки.

Входные данные

Первая строка входного файла INPUT.TXT содержит целое число N ($3 \leq N \leq 1000$) – количество кустов черники. Вторая строка содержит N целых положительных чисел a_1, a_2, \dots, a_N – число ягод черники, растущее на соответствующем кусте. Все a_i не превосходят 1000.

Выходные данные

В выходной файл OUTPUT.TXT выведите ответ на задачу.

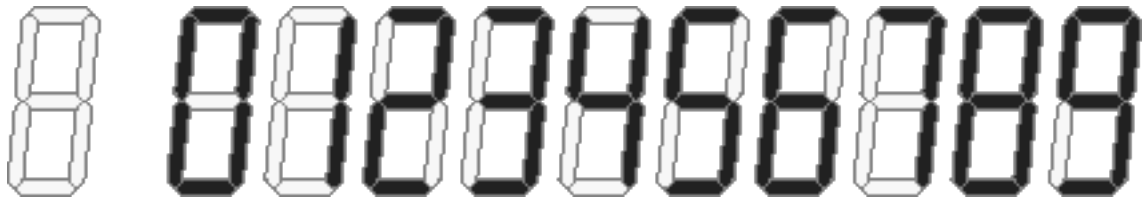
Примеры

№	INPUT.TXT	OUTPUT.TXT
1	4 1 2 3 4	9
2	3 1 2 3	6

Задача №497 Индикатор

(Время: 1 сек. Память: 16 Мб Сложность: 52%)

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.

Входные данные

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

Выходные данные

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
1	5 15	10117 97111
2	10 1	NO SOLUTION

Задача №498 К-перестановки

(Время: 1 сек. Память: 16 Мб Сложность: 44%)

Назовем перестановку из N чисел $1 \dots N$ *К-перестановкой*, если любые два соседних в ней элемента отличаются не более, чем на K .

Требуется найти число K -перестановок из N чисел от 1 до N .

Входные данные

Входной файл INPUT.TXT содержит натуральные числа N и K ($K \leq N \leq 9$).

Выходные данные

В выходной файл OUTPUT.TXT выведите целое число - ответ на задачу.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	3 1	2
2	4 2	12

Задача №499 Турист

(Время: 1 сек. Память: 16 Мб Сложность: 22%)

Гена собирается на туристический слет учеников своей школы. В своем классе он был назначен ответственным за палатки. У себя дома он нашел 3 палатки: первая из них весит a_1 килограмм и вмещает b_1 человек, вторая весит a_2 килограмм и вмещает b_2 человек, третья весит a_3 килограмм и вмещает b_3 человек.

В классе Гены K человек. Выясните, может ли он выбрать палатки так, чтобы в них все могли поместиться. При этом учитывайте, что выбранные палатки должны суммарно весить не более W килограмм.

Входные данные

Первая строка входного файла INPUT.TXT содержит два целых числа: K и W ($1 \leq K \leq 15$, 1

$\leq W \leq 30$). Вторая строка содержит шесть целых чисел: $a_1, b_1, a_2, b_2, a_3, b_3$ ($1 \leq a_1, a_2, a_3 \leq 10, 1 \leq b_1, b_2, b_3 \leq 15$).

Выходные данные

В выходной файл OUTPUT.TXT выведите YES, если палатки указанным образом выбрать можно, и NO в противном случае.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	10 10 5 5 6 6 4 5	YES
2	10 10 5 5 6 6 7 7	NO

Задача №500 Агент

(Время: 1 сек. Память: 16 Мб Сложность: 34%)

Агент Джеймс Бонд пошел на пенсию, но неугомонный характер требовал новых впечатлений. Поэтому Джеймс Бонд с удовольствием согласился провести мастер-класс в некоторых группах школы «Молодого агента». Тема одного из занятий – работа агента с напарником. В таком опасном деле, как разведка, важно иметь очень надёжного напарника, поэтому напарниками могут стать только агенты, которые максимально близки по возрасту (т.е. два агента не могут стать напарниками, если в группе существует третий агент, который старше одного и младше другого).

Задание Бонда состоит в том, чтобы агенты нашли друг другу напарников таким образом, чтобы у каждого агента был хотя бы один напарник (всего у агента может быть 2 напарника – один младше, и один старше него, но эти двое не считаются напарниками между собой). Очевидно, что группа из 4 и более агентов может поделиться несколькими способами.

После нескольких занятий Бонд узнал способности групп, обучающихся в школе «Молодого агента», и оценил риск раскрытия каждого агента в отдельности. Но специфика работы с напарником такова, что в паре риску подвергается только старший из двух агентов, поэтому группу надо распределить так, чтобы суммарный риск был минимален.

Входные данные

В первой строке входного файла INPUT.TXT находится одно целое число N – количество агентов в группе ($2 \leq N \leq 10000$). Во второй строке находятся N пар целых положительных чисел, разделенных пробелом. Первое число в паре – это возраст агента (в днях) из диапазона $[5000, 16000]$, второе – риск раскрытия агента, число в диапазоне $[1, 1000]$. Известно, что в любой группе все агенты разного возраста.

Выходные данные

В выходной файл OUTPUT.TXT выведите единственное число – минимальное значение суммарного риска раскрытия группы.

Примеры

№	INPUT.TXT	OUTPUT.TXT

1	3 6000 2 5500 3 5000 4	5
2	5 5005 1 5004 2 5003 3 5002 4 5001 5	7

Задача №501

Строение

(Время: 1 сек. Память: 16 Мб Сложность: 25%)

Администрация города подбирает площадку для строительства новых спортивных сооружений. На рассмотрении несколько проектов, каждый проект требует выделения некоторого прямоугольного участка земли. Некоторые участки оказались частично или полностью в пользовании частными лицами, а в случае утверждения проекта администрация будет вынуждена выкупить этот участок, поэтому для определения стоимости очень важно знать площадь пересечения участков. Участки частников также прямоугольной формы (рис. 1) и стороны всех участков параллельны координатным осям. Для каждого проекта был построен план, включающий подобранный участок и его окружение. В приведенном примере показано пересечение участков частников (тонкая линия) с участком, подобранным для строительства (толстая линия). Помогите определить суммарную площадь пересечения участков частников с участком, выбранным для строительства. При этом следует учесть, что выкупаемая земля может принадлежать сразу нескольким участникам и в этом случае необходимо выкупать землю многократно (одна и та же площадь земли может быть посчитана несколько раз).

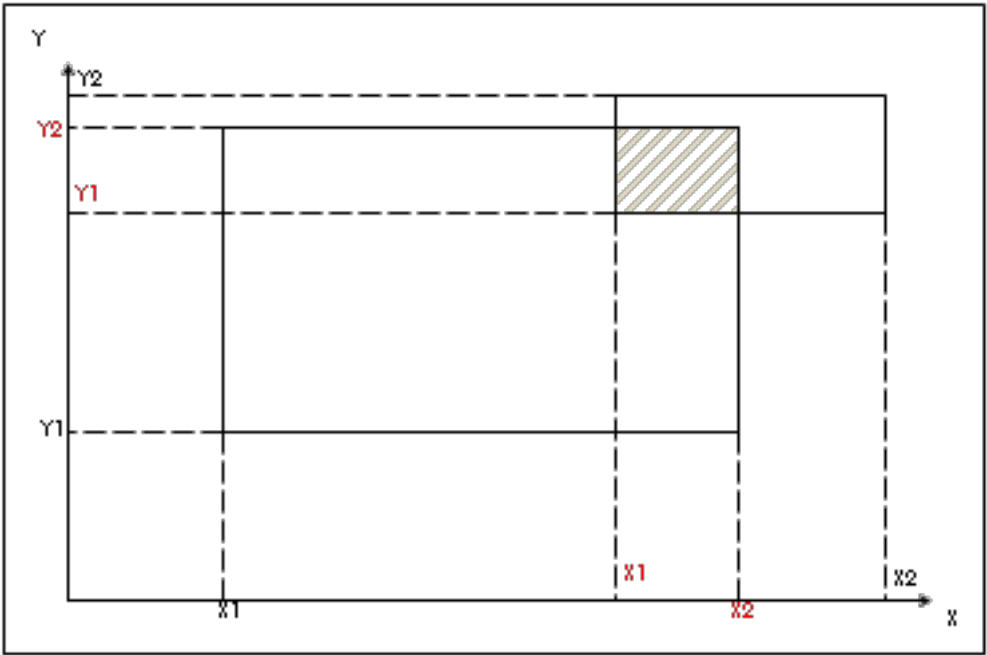


Рис. 1.

Входные данные

В первой строке входного файла INPUT.TXT содержится число N - количество участков частников, отображенных на плане. Затем следуют N строк с координатами двух вершин этих прямоугольных участков. В последней строке плана координаты участка, выбранного для строительства. Координаты противоположных вершин одного прямоугольника описываются в формате $X_1\ Y_1\ X_2\ Y_2$. Координатами вершин являются целые, неотрицательные числа, не больше 100. Количество исходных прямоугольников не больше 20.

Выходные данные

В единственную строку выходного файла OUTPUT.TXT следует вывести одно целое число – искомую суммарную площадь пересечения прямоугольников.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	1 15 15 25 25 10 10 20 20	25
2	2 15 15 25 25 5 5 12 12 10 10 20 20	29

Задача №502 Лягушонок

(Время: 1 сек. Память: 16 Мб Сложность: 39%)

Многие, вероятно, слышали песни про приключения лягушонка Crazy Frog. На этот раз неугомонное милое создание решило подкрепиться, но даже такое простое действие решило выполнить в виде игры. Итак, в каждой клетке квадратного игрового поля, разбитого на $N \times N$ ($N \leq 50$) клеток, находится один комар весом a_{ij} (вес комара – натуральное число ≤ 50), i – номер строки, j – номер столбца. Лягушонок, прыгая с клетки на клетку, ест комаров. Правила игры таковы – в каждом столбце можно съесть не более одного комара. Всякий раз при съедании комара запоминаем номер строки, откуда съеден комар, и сумма номеров строк, в которых были съедены комары, в конце игры должна быть в точности равна N . Учтите, если из какой-то строки съедено несколько комаров, то номер данной строки участвует в суммировании более одного раза.

Определите максимальный вес комаров, который можно съесть при следовании приведённым правилам.

Входные данные

Первая строка входного файла INPUT.TXT содержит число N . Следующие N строк содержат по N чисел a_{ij} , разделенных пробелами.

Выходные данные

В выходной файл OUTPUT.TXT выведите целое число – вес съеденных комаров.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	3 8 2 1 1 2 6 2 7 2	14
2	5 8 2 1 2 3 1 2 6 2 4 2 7 2 3 4 1 3 2 4 4 1 3 4 3 1	19

Задача №503

Бюро путешествий

(Время: 3 сек. Память: 32 Мб Сложность: 80%)

Далеко не все в Тентуре имеют право носить малиновые штаны, и конечно, не все владеют пепелцем с гравицапой, поэтому для большинства жителей проблема перемещения между планетами была неразрешимой. Но с некоторых пор один предприимчивый чатланин с планеты Плюк вышел на рынок пассажирских перевозок, и за немного чатлов, готов перевозить желающих с планеты на планету. Рейс начинается с планеты Плюк, включает нескольких других планет, и завершается там же, на планете Плюк. Однако при подготовке рейса возникли неожиданные проблемы. Например, если чатланин с планеты Плюк хочет попасть на планету, которая является в рейсе предпоследней – ему невольно придётся посетить все планеты, которые находятся в рейсе между планетой Плюк и его точкой назначения. Очевидно, что часть планет в этом списке могут оказаться пацакскими. Но каждый чатланин обязан носить цак на пацакской планете и, наоборот, каждый пацак должен носить цак на чатланской планете. (Цак — колокольчик для носа, знак отличия для относительно низшей касты на данной планете). А процедура ношения цака унизительна во всех смыслах этого слова...

Поскольку данное бюро путешествий пока не имеет представительств на других планетах, перевозка осуществляется только с планеты Плюк на какую-либо другую, либо с другой планеты на Плюк. Задача планирования рейса упрощается – можно посещать планеты в произвольном порядке (но нельзя посещать одну и ту же планету дважды – в пути может закончиться луц). Необходимо вычислить такой порядок посещения планет, при котором надевать цак на промежуточных планетах придётся минимальное количество раз.

Входные данные

Первая строка входного файла INPUT.TXT содержит натуральное число $N \leq 22$ – количество планет, обслуживаемых данным рейсом (не считая планеты Плюк). N следующих строк содержат информацию о планетах, в следующем виде: тип планеты (латинская заглавная буква С – чатланская, Р – пацакская), количество чатлан, следующих до этой планеты с Плюка, количество пацаков, следующих до этой планеты с Плюка, количество чатлан, с данной планеты на Плюк, количество пацаков, с данной планеты на Плюк. Всего пассажиров ≤ 1000 .

Выходные данные

В выходной файл OUTPUT.TXT выводится, сколько раз придётся надевать цак при оптимальном маршруте, затем порядок посещения планет через пробел. Планеты, перечисленные во входном файле, нумеруются начиная с единицы, планета Плюк имеет номер ноль и всегда указывается в последовательности дважды – в начале и в конце последовательности. Если существуют несколько оптимальных маршрутов – то следует выбрать тот, где планета с меньшим номером посещается раньше.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	2 С 1 4 5 2 Р 2 5 1 4	5 0 2 1 0
2	4 С 3 0 0 0 С 3 0 0 1 С 3 0 0 1 С 3 0 0 1	3 0 1 2 3 4 0

Пояснение

В первом тесте возможны два варианта маршрута: 0 1 2 0, и 0 2 1 0. В первом варианте при посадке на чатланской планете 1 пятерым пацакам, которые следуют транзитом к своей планете 2, придётся надеть цак, а при следующей посадке на пацакской планете 2, пятерым чатланинам, которые сели на планете 1 и следуют до Плюка, также придётся надеть цак. Итого в первом варианте маршрута цак надевается 10 раз. В варианте 2 цак на промежуточных посадках надевается 4 и 1 раз соответственно, всего 5 раз. Второй вариант предпочтительнее.

Во втором тесте все планеты чатланские, поэтому надевать цак придётся только пацакам. С Плюка пацаки не отправляются, но прибывают с трёх разных планет по одному. Первой посещается единственная планета, где пацак не заходит в пепелац, затем следуют три планеты с одинаковым набором пассажиров – на втором шаге из трёх оставшихся равноценных планет выбирается планета номер 2 как имеющая наименьший номер, затем 3 и оставшаяся 4. Пассажир с последней планеты не имеет промежуточных посадок, с предпоследней совершает одну посадку и надевает цак 1 раз, и оставшийся пассажир надевает цак на двух промежуточных остановках, итого цак надевается 3 раза.

Задача №504 Цветочки

(Время: 1 сек. Память: 16 Мб Сложность: 12%)

В рождественский вечер на окошке стояло три цветочка, слева на право: герань, крокус и фиалка. Каждое утро Маша вытирала окошко и меняла местами стоящий справа цветок с центральным цветком. А Таня каждый вечер поливала цветочки и меняла местами левый и центральный цветок. Требуется определить порядок цветов ночью по прошествии К дней.

Входные данные

Во входном файле INPUT.TXT содержится натуральное число К – число дней ($K \leq 1000$).

Выходные данные

В выходной файл OUTPUT.TXT требуется вывести три латинских буквы: «G», «C» и «V» (заглавные буквы без пробелов), описывающие порядок цветов на окошке по истечении К дней (слева направо). Обозначения: G – герань, C – крокус, V – фиалка.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	1	VGC
2	5	CVG

Задача №505 Забор

(Время: 1 сек. Память: 16 Мб Сложность: 46%)

Для парка культуры и отдыха было решено изготовить забор. Чтобы забор не портил своим невзрачным видом облик города, архитекторы решили сделать забор фигурным, и разработали несколько различных шаблонов для изготовления секций забора. Шаблон представляет собой многоугольник, где три стороны всегда одинаковы, а четвертая сторона (верхняя часть забора)

представляет ломаную линию. Проекции вершин этой ломаной линии на основание забора следуют равномерно, и таким образом шаблон описывается как последовательность высот точек. Высота точек указывается в целых миллиметрах и варьируется от 0 до 2047.

Количество точек в шаблоне равняется L . Архитекторы разработали M шаблонов, и завод изготовил по шаблонам N секций. Пример секции с $L = 7$ изображен на рисунке.

При транспортировке секций забора до парка произошла авария, и готовые секции забора рассыпались и перемешались. При разборе завалов были выполнены измерения секций, и теперь, зная высоты, необходимо определить, к какому шаблону принадлежит каждая секция забора. Кроме того, при падении секции забора могли быть повреждены (например, могли обломаться зубья забора), поэтому необходимо также посчитать случаи, когда извлечённая из завала секция не совпадает ни с одним шаблоном.

Впрочем, дело несколько упрощается – внутренняя (обращённая в парк) сторона секции отличается по цвету от внешней стороны, поэтому можно сказать с уверенностью, что измерения высот секции проделаны в том же порядке, что и в шаблоне.

Входные данные

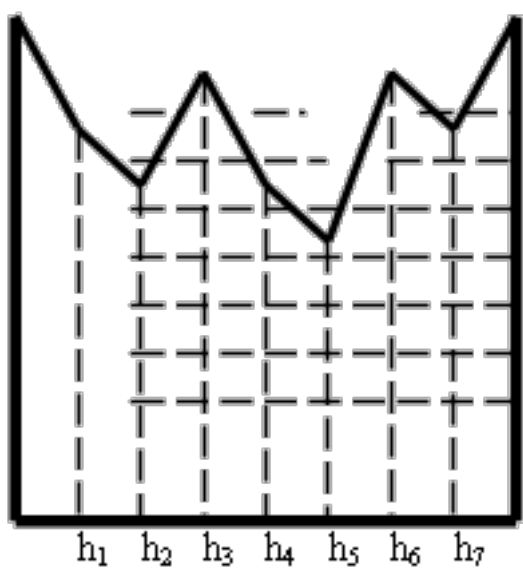
Входной файла INPUT.TXT содержит целые значения L – число точек в шаблоне и секции, M – число различных шаблонов, N – число найденных секций ($1 \leq L, M \leq 1000, N \leq 1000$). Далее M строчек содержат информацию о шаблоне: номер шаблона (натуральное число) и L точек шаблона. Далее N строчек содержат информацию о секциях забора, в каждой строке содержится L точек одной секции забора.

Выходные данные

В выходной файл OUTPUT.TXT следует вывести N строк, каждая строка содержит номер шаблона, с которым совпала секция, либо, если подходящего шаблона не найдено, то выводится символ дефиса '-'. В последней строке теста выводится слово "OK=" (без кавычек) и число секций, сопоставленных с шаблонами, через пробел символы "BAD=" и число испорченных секций.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	4 2 4 1 250 123 0 66 2 22 31 120 100 250 123 0 66 22 31 120 100 25 31 120 100 250 123 0 66	1 2 - 1 OK=3 BAD=1
2	3 3 4 1 55 11 12 3 33 1 2 2 14 15 2 55 11 12 14 15 2 33 1 2 14 15 2	1 2 3 2 OK=4 BAD=0



Задача №506

Игра - 3

(Время: 1 сек. Память: 16 Мб Сложность: 55%)

Петя и Вася увлеклись азартной игрой на деньги. Игра состоит из множества раундов, в каждом из которых бросается монетка и если выпадает «Решка», то побеждает Петя и получает очко, в противном случае очко присуждается Васе. Набранные очки в разных раундах суммируются и тот, кто первым наберет N очков, выигрывает.

Вдруг, посреди игры прозвенел звонок, и ребятам пришлось закончить игру. Помогите Пете и Васе справедливо поделить поставленные на кон монеты между собой согласно наиболее вероятному исходу игры.

Входные данные

Входной файл INPUT.TXT содержит в одной строке 4 целых числа N , $K1$, $K2$ и S . Все числа разделены пробелами. N – количество очков, которые необходимо набрать для победы. $K1$ и $K2$ – текущие очки Пети ($K1$) и Васи ($K2$). S – число монет, которые на кону и которые необходимо поделить между игроками.

Ограничения: $1 \leq N \leq 50, 0 \leq K1, K2 < N, 1 < S < 10^{100}$.

Выходные данные

В выходной файл OUTPUT.TXT через пробел выведите два числа: количество монет, которые должны получить Петя и Вася. Не забудьте, что сумма этих чисел в точности должна быть равна S .

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	3 1 2 16	4 12
2	4 1 1 2	1 1
3	5 1 4 32	2 30

Задача №507

Адронный коллайдер

(Время: 1 сек. Память: 16 Мб Сложность: 52%)

Внутри адронного коллайдера образовалось N разновидностей новых частиц в количестве A_1, \dots, A_N единиц каждая. Большая часть новых частиц, однако, успевает прореагировать между собой раньше, чем эти частицы сможет уловить детектор, поэтому физикам очень важно предсказать конечный итог реакции между частицами.

Для простоты будем считать, что в реакции участвуют 2 частицы, с одним из следующих результатов, в зависимости от типа:

- Первая частица уничтожает вторую
- Обе частицы взаимно уничтожаются
- Частицы отскакивают друг от друга без какого-либо вреда

Необходимо определить все возможные исходы эксперимента.

Входные данные

В первой строке входного файла INPUT.TXT задано количество видов частиц N ($1 \leq N \leq 4$). В следующей строке записаны N чисел, определяющие начальное количество частиц каждого типа A_i ($1 \leq A_i \leq 2$). Следующие N строк формируют матрицу $A[N][N]$. Ненулевое значение ячейки A_{ij} указывает, что частица типа i при столкновении уничтожает частицу типа j .

Выходные данные

В первой строке выходного файла OUTPUT.TXT следует вывести количество возможных исходов. В каждой из последующих N строк должно содержаться описание исхода эксперимента в формате исходного файла.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	3 1 1 2 0 0 1 1 0 0 1 1 1	3 0 1 0 0 0 1 1 0 0
2	1 2 0	1 2

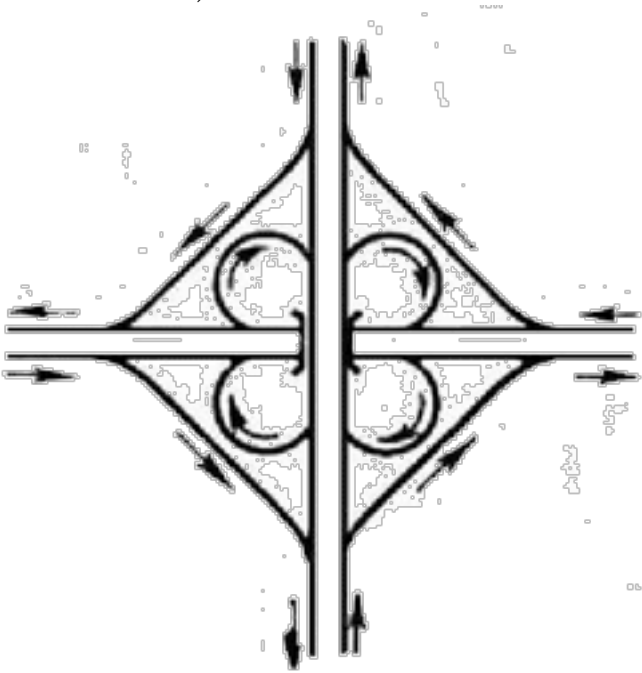
Задача №508 Клеверный лист

(Время: 1 сек. Память: 16 Мб Сложность: 36%)

В одном крупном городе строится объездная автодорога. Она не является кольцевой, то есть имеет западное и восточное окончание. Наиболее интересными инженерными объектами на дороге, являются, конечно же, мосты. Очевидно, что железобетонный мост слишком дорогой для участка с одним автомобилем в день, а деревянный мост не способен обслуживать шоссе с десятком машин в минуту. Поэтому для оправданного проектирования мостов необходимо знать, сколько автомобилей проезжает через мост.

На дороге будут использоваться преимущественно развязки "клеверный лист", схема такой развязки изображена на рисунке. Согласно этой схеме, автомобиль, поворачивающий направо - не едет через мост, автомобиль, поворачивающий налево (по этой развязке он поворачивает на 270 градусов направо и пересекает свой путь на другом уровне) - всегда едет через мост, и автомобиль, проезжающий прямо - может ехать, а может не ехать через мост, тут всё зависит от направления движения.

Исследован поток автомобилей, которые едут по дорогам, пересекающим объездную дорогу. Необходимо посчитать поток автомобилей, которые поедут через мосты объездной дороги. Следует помнить, что для некоторых перекрёстков объездная дорога проходит по земле, а мост содержит поперечная дорога - считать такие мосты не требуется. Хоть дорога двусторонняя, один тест исследует только одно направление объездной дороги (например, с запада на восток).



Входные данные

Первая строка входного файла INPUT.TXT содержит число $N \leq 100$ – количество перекрёстков, перекрёстки перечислены с запада на восток. Следующие N строк содержат информацию о типе пересечения, затем о потоке поперечной дороги. Тип пересечения - латинская буква L, означает что объездная дорога проходит по земле, а поперечная проходит через мост, буква В означает что поперечная дорога проходит по земле, а объездная через мост. Далее указаны 2 числа, показывающее количество автомобилей, покинувших объездную дорогу, первое число - с поворотом налево, второе - с поворотом направо. Далее указаны 2 числа, показывающее количество автомобилей, выехавших на объездную дорогу, первое число - с поворотом налево, второе - с поворотом направо. Количество машин по любой из поперечных дорог не превышает 10^9 .

Гарантируется корректность входных данных (изначально на трассе машин нет, покинуло трассу столько же машин, сколько и заехало на трассу, точка перекрёсток схода находится позднее перекрёстка захода).

Выходные данные

В выходной файл OUTPUT.TXT выводится одна строка, содержащая N чисел - потоки машин на мост объездной дороги. Если объездная дорога проходит по земле, а мост - над ней, то выводится -1.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	2 В 0 0 2 3 В 4 1 0 0	2 4
2	5 В 0 0 9 9 В 0 0 0 0 L 3 3 0 0 В 0 0 0 0 В 8 4 0 0	9 18 -1 12 8

Задача №509 Игра Jammed

(Время: 1 сек. Память: 16 Мб Сложность: 48%)

Всем известна игра «Пятнашки», где надо выстроить изначально неупорядоченную последовательность чисел, перемещая фишки с нанесёнными числами от 1 до 15 в квадрате 4×4. На основе данной игры была разработана другая – поле в ней лишь 4×2 клетки, на поле 7 фишек, но на фишках изображены буквы латинского алфавита и арабские цифры (на каждой фишке – один символ, но на разных фишках могут быть одинаковые символы). Цель игры прежняя – упорядочить в соответствии с образцом стартовую расстановку фишек за минимальное количество ходов.

Свободная клетка обозначается специальным символом «#» и используется для перемещения фишек по полю. Перемещать фишки на свободную клетку разрешается из соседних клеток, имеющих общую грань со свободной. Например, на рисунке более правый символ «0» можно переместить вниз на свободную клетку, тогда «0» будет в нижней клетке, а пустой станет верхняя клетка, либо в свободную клетку переместить букву «С» или цифру «2».

М	0	0	А
8	С	#	2

Входные данные

Входной файл INPUT.TXT содержит четыре строки: две первые строки содержат стартовую комбинацию символов, следующие две - образец. Каждая строка содержит 4 символа (латинский алфавит и арабские цифры), пустая клетка обозначается символом «#» (решетка).

Выходные данные

В выходной файл OUTPUT.TXT выведите минимальное количество перемещений, необходимых для получения искомой комбинации. Если нужную комбинацию получить нельзя, выведите число -1.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	АСМ8 002# АСМ# 2008	17
2	rogp mar# prog ram#	26

Задача №510
Шоколадка

(Время: 1 сек. Память: 16 Мб Сложность: 40%)

Саша, не сделал домашнюю работу, зато купил шоколадку. И, по глупости, начал распечатывать ее прямо на уроке... Шелест золотинки услышала учительница. Она хотела вызвать в школу родителей, но Саша уговорил ее не вызывать их, а дать дополнительное задание.

Учительница внимательно посмотрела на шоколадку (она была размером 3x4 плиток), разделила на кусочки по две плитки и угостила всех, кто сделал домашнюю работу. А Сашу попросила написать программу, которая определяет, сколько существует способов деления шоколадки размером 3xN плиток на кусочки по две плитки.

Для выполнения задания Саше нужна помощь.

Входные данные

Входной файл INPUT.TXT содержит натуральное число N – размер плитки, (N < 33).

Выходные данные

В выходной файл OUTPUT.TXT выведите одно целое число – количество возможных способов.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	2	3
2	4	11

Задача №511 Очередь

(Время: 1 сек. Память: 16 Мб Сложность: 20%)

Студент Василий живет в общежитии. Отделение банка, в котором он производит оплату за проживание, имеет всего две кассы, поэтому почти всегда длинная очередь к ним. Первая касса открывается в 8.00, вторая – в 8.05. Последний клиент будет принят в 20.00. Очередь единая, и очередной клиент обслуживается, как только освобождается одна из касс. На обслуживание одного клиента уходит ровно 10 минут. Василий приходит ровно в 8.00 и видит, сколько человек стоит перед ним. Требуется определить, сколько времени ему придется простоять в очереди, и вообще обслужат ли его сегодня.

Входные данные

Входной файл INPUT.TXT содержит единственное натурально число K – номер Василия в очереди ($K < 250$).

Выходные данные

В выходной файл OUTPUT.TXT выводится строка «NO», если Василий сегодня заплатить уже не успеет, и время его ожидания (в формате «X Y», где X – количество целых часов, которые простоит в очереди Василий, и Y – количество минут), если все же успеет заплатить.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	1	0 0
2	20	1 35
3	235	NO

Задача №512 Выпускной

(Время: 1 сек. Память: 16 Мб Сложность: 75%)

Вот и еще один учебный год подошел к концу, и старшеклассники спешно начали готовиться к экзаменам. Как всегда после всех экзаменов должен был состояться выпускной бал, и в школе уже всем ходом шла подготовка к нему. Одной, но очень обязательной частью этого мероприятия должен был стать вальс, кружиться в котором должно по задумке ровно K пар. Ответственным за танец был назначен школьный учитель трудов Александр Степанович. По его первому указанию по всей школе были развешаны объявления о наборе мальчиков и девочек, которые хотят принять участие в танце. По окончанию набора всех желающих, в списке у «Степаныча» было записано N мальчиков и M девочек. Теперь нужно было сформировать из всего этого ровно K пар, но оказалось, что сделать это не так, то просто. Оказывается, что не каждый мальчик захочет

танцевать с любой девочкой и не каждая девочка согласится вальсировать с любым мальчиком. Перед нашим героем встала нелегкая задача и, немного поразмыслив, он решил думать масштабно – а конкретно, сколько же всего существует вариантов сформировать из N мальчиков и M девочек ровно K пар с учетом их предпочтений? Так как в рубанках и молотках «Степаныч» понимает больше чем в математике, с просьбой помочь ему он обратился к вам.

Входные данные

В первой строке входного файла INPUT.TXT содержится 3 целых числа: N – количество мальчиков, M – количество девочек и K – количество пар ($1 \leq N, M, K \leq 10$). Далее идет описание пожеланий мальчиков и девочек – N строк длины M. Если символ, стоящий в i-ой строке на j-ом месте равен «Y» – это значит что i-й мальчик и j-я девочка могут стоять в паре и «N» если не могут.

Выходные данные

В выходной файл OUTPUT.TXT выведите целое число - количество различных способов формирования из N мальчиков и M девочек ровно K пар с учетом их пожеланий.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	3 4 3 Y Y Y Y Y Y Y Y Y Y Y Y	24
2	1 5 1 Y Y N N Y	3

Задача №513 Салаты

(Время: 1 сек. Память: 16 Мб Сложность: 26%)

Как-то раз, придя домой со школы, Света обнаружила записку от мамы, в которой она просила сделать салат. Света знала, что салат – это смесь двух или более ингредиентов, поэтому ей не составило труда выполнить мамину просьбу.

Но Света хочет стать математиком, поэтому, для тренировки, решила посчитать, сколько различных салатов она сможет сделать из имеющихся продуктов (майонез, огурцы, помидоры). После небольших расчетов она получила ответ: 4.

Зная, что вы любите интересные задачки, и хотите стать программистами, Света попросила вас написать программу, которая определяет количество различных салатов для произвольного числа ингредиентов.

Входные данные

Входной файл INPUT.TXT содержит натуральное число N – количество имеющихся ингредиентов ($N < 32$).

Выходные данные

В выходной файл OUTPUT.TXT выведите количество различных салатов.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	3	4
2	4	11

Задача №514

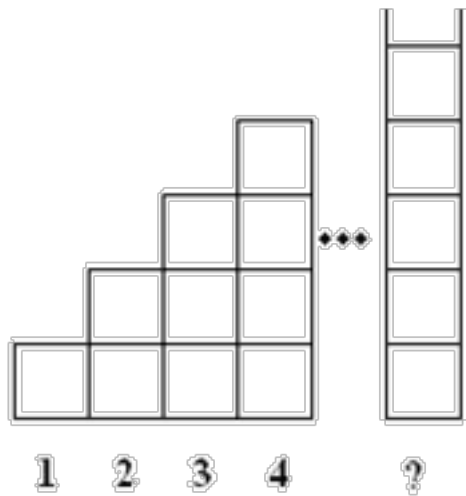
Лестница

(Время: 1 сек. Память: 16 Мб Сложность: 21%)

Мальчик Петя строит из кубиков лестницу. Лестница представляет собой несколько строящихся рядом башенок из кубиков, каждая из которых ровно на один кубик выше предыдущей. Требуется по имеющемуся у мальчика Пети числу кубиков определить, какой в кубиках будет высота последней ступеньки.

Входные данные

Входной файл INPUT.TXT содержит число K – количество кубиков у мальчика Пети ($1 \leq K \leq 10^6$).



Выходные данные

В выходной файл OUTPUT.TXT выведите количество кубиков в последней ступеньке у максимально высокой лестницы, которую можно построить из K кубиков.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	1	1
2	4	2
3	6	3

Задача №515

Ловушки

(Время: 1 сек. Память: 16 Мб Сложность: 24%)

Далеко не все в Тентуре имеют право носить малиновые штаны, и конечно, не все владеют пепелцем с гравицапой, зато один предприимчивый чатланин поставил несколько ловушек, захватывающих отваливающиеся детали с пролетающих мимо инопланетных кораблей. Каждое утро он их обходит, в надежде, что рано или поздно из этих запчастей ему удастся собрать какое-нибудь транспортное средство. Путь свой чатланин планирует по карте, на которой аккуратно отмечены ловушки, отсчитывая начало координат от собственного дома. Начинается обход от дома, и в нем же и заканчивается.

Требуется определить, какое расстояние проходит чатланин за день.

Входные данные

В первой строке входного файла INPUT.TXT находится натуральное число N – количество ловушек, установленных чатланином ($N \leq 10$), далее следуют N строк формата «X Y», описывающих координаты ловушек в порядке их обхода ($-10 \leq X, Y \leq 10$; X, Y – целые числа).

Выходные данные

В выходной файл OUTPUT.TXT выведите общее расстояние, которое пройдет чатланин, обойдя все ловушки в приведенном порядке и вернувшись в начало координат от последней ловушки, с точностью до трех знаков после запятой.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	1 1 0	2.000
2	2 1 0 1 1	3.414
3	3 1 0 1 1 0 1	4.000

Задача №516 2-простое число

(Время: 1 сек. Память: 16 Мб Сложность: 28%)

Число называется 2-простым, если являются простыми числа, составленные из цифр этого числа в возрастающем и убывающем порядках.

Требуется написать программу, которая по заданному числу определит его 2-простоту.

Входные данные

Входной файл INPUT.TXT содержит натуральное число N ($10 < N < 10^9$).

Выходные данные

В выходной файл OUTPUT.TXT выведите сообщение «Yes», если число N является 2-простым и «No» - иначе.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	13	Yes
2	23	No

Задача №517 Боулинг

(Время: 1 сек. Память: 16 Мб Сложность: 29%)

Цель при игре в боулинг – сбить шаром максимальное количество кеглей. Партия в этой игре состоит из 10 туров. Задача игрока – сбить все 10 кеглей в каждом туре. Для этого игрок может совершить 2 броска шара, за исключением:

- если 10 кеглей сбиты первым броском, то второй бросок не совершается;
 - если 10 кеглей сбиты первым броском в десятом туре, то игроку предоставляются два призовых броска, а если двумя бросками – один.
- Количество очков в каждом туре равно количеству сбитых кеглей, кроме двух бросков, называемых «Strike» и «Spare».
- Strike: игрок сбивает 10 кеглей первым броском, очки в этом туре начисляются из расчета – 10 + сумма очков за два последующих броска.
- Spare: игрок сбивает 10 кеглей двумя бросками, очки в этом туре начисляются из расчета – 10 + сумма очков за один последующий бросок.
- Результат партии складывается из результатов всех 10 туров.
- Требуется написать программу, которая определит количество набранных игроком очков.

Входные данные

Входной файл INPUT.TXT содержит в первой строке одно натуральное число, определяющее количество совершенных бросков. Вторая строка содержит натуральные числа (разделенные пробелом), обозначающие количество сбитых кеглей за каждый совершенный бросок.

Выходные данные

Выходной файл OUTPUT.TXT должен содержать одно целое число – количество набранных игроком очков.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	12 10 10 10 10 10 10 10 10 10 10 10 10 10	300
2	20 0	0
3	15 10 10 10 8 2 10 3 4 8 2 4 5 10 4 5	173

Задача №518

Количество путей в лабиринте

(Время: 1 сек. Память: 16 Мб Сложность: 38%)

Карта лабиринта представляет собой квадратное поле размером NхN. Некоторые квадраты этого поля запрещены для прохождения. Шаг в лабиринте – перемещение из одной разрешенной клетки к другой разрешенной клетке, смежной с первой по стороне. Путь – это некоторая последовательность таких шагов. При этом каждую клетку, включая начальную и конечную, можно посещать несколько раз.

Требуется написать программу, которая подсчитает количество различных путей из клетки (1, 1) в клетку (N, N) ровно за K шагов (то есть оказаться в клетке (N, N) после K-го шага).

Входные данные

Входной файл INPUT.TXT содержит в первой строке числа N и K, разделенные пробелом (1 < N <= 15, 0 < K <= 30). Следующие N строк, по N символов в каждой, содержат карту лабиринта,

начиная с клетки (1, 1). Символ «0» означает не запрещенную для прохождения клетку, а символ «1» - запрещенную. Начальная и конечная клетки всегда разрешены для прохождения.

Выходные данные

Выходной файл OUTPUT.TXT должен содержать количество возможных различных путей длины K. Во всех тестах это значение не будет превышать 2147483647.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	3 6 0 0 0 1 0 1 1 0 0	5
2	2 8 0 1 1 0	0

Задача №519

Наименьшее и наибольшее числа из тех же цифр

(Время: 1 сек. Память: 16 Мб Сложность: 35%)

Требуется написать программу, которая найдет наименьшее и наибольшее числа, состоящие из тех же цифр, что и заданное натуральное число N.

Входные данные

Входной файл INPUT.TXT содержит натуральное число N ($N \leq 2 \cdot 10^9$).

Выходные данные

Выходной файл OUTPUT.TXT должен содержать в одной строке наименьшее, а через пробел – наибольшее числа.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	7051	1057 7510
2	851	158 851

Задача №520

Оптовая покупка

(Время: 1 сек. Память: 16 Мб Сложность: 27%)

Пара носков стоит 10 руб. 50 коп., связка (12 пар) стоит 102 руб. 50 коп., а коробка (12 связок) стоит 1140 руб.

Требуется написать программу, которая по числу пар носков, которые хочет купить покупатель, вычисляет количества коробок, связок и пар носков, которые ему следует купить с наибольшей выгодой.

Входные данные

Входной файл INPUT.TXT содержит натуральное число N ($N \leq 10^9$) – число пар носков, которые желает купить покупатель.

Выходные данные

Выходной файл OUTPUT.TXT должен содержать три числа (первое – количество коробок, второе – связок, третье – пар носков), разделенные пробелами.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	11	0 1 0
2	500	3 5 8

Задача №521 Пасьянс старухи Шапокляк

(Время: 1 сек. Память: 16 Мб Сложность: 25%)

На столе лежат колоды игровых карт. В самой тоненькой колоде – p карт, во второй – $p+1$, в третьей – $p+2$, ..., в последней – k карт. Старуха Шапокляк раскладывает пасьянс. Беря в руки любую из колод, она, если число карт в ней четное, на место возвращает колоду, наполовину уменьшив число карт в ней (лишние убирает в ящик), а если количество карт в колоде нечетное, то утраивает их количество и добавляет еще одну карту, а уже тогда кладет колоду на стол (карт у нее в ящике для этой операции достаточно). Если в какой-то колоде остается две карты, она больше ее не трогает. Пасьянс сходится, если во всех колодах остается по две карты.

Требуется написать программу, которая определит сходится ли пасьянс, и если сходится – сколько раз должна старуха Шапокляк брать со стола карты.

Входные данные

Входной файл INPUT.TXT содержит 2 числа, записанные через пробел ($2 < p < k < 1000$).

Выходные данные

Выходной файл OUTPUT.TXT должен содержать 0, если пасьянс не сходится, и, если сходится, количество «ходов» старухи Шапокляк.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	2 3	6
2	5 8	28

Задача №522 Похожие массивы

(Время: 1 сек. Память: 16 Мб Сложность: 30%)

Два массива называются похожими, если совпадают множества чисел, встречающихся в этих массивах.

Требуется написать программу, которая определит: похожи ли два заданных массива.

Входные данные

Входной файл INPUT.TXT содержит в первой строке два числа M и N - длины массивов ($1 \leq M, N \leq 16000$). Во второй строке записаны M чисел – элементы первого массива. В третьей строке записаны N чисел – элементы второго массива. Числа в строках разделены пробелами, элементы массивов - целые числа, не превышающие 32000 по абсолютной величине.

Выходные данные

Выходной файл OUTPUT.TXT должен содержать 1, если массивы похожи и 0 иначе.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	4 3 1 2 3 2 1 2 3	1
2	2 3 1 2 2 3 1	0

Задача №523 Роман в томах

(Время: 1 сек. Память: 16 Мб Сложность: 41%)

В романе N глав. В i -той главе a_i страниц. Требуется издать роман в K томах так, чтобы объем самого «толстого» тома был минимален. В каждом томе главы располагаются по порядку своих номеров.

Требуется написать программу, которая найдет количество страниц в самом «толстом» томе.

Входные данные

Входной текстовый файл INPUT.TXT содержит в первой строке число N ($1 \leq N \leq 100$). Во второй строке через пробел записаны N чисел – количество страниц в каждой главе. Количество страниц в романе не превышает 32767. В третьей строке записано число K ($1 \leq K \leq N$).

Выходные данные

Выходной файл OUTPUT.TXT должен содержать количество страниц в самом «толстом» томе.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	3 1 2 1 2	3
2	4 1 2 1 1 3	2

Задача №524

Слон

(Время: 1 сек. Память: 16 Мб Сложность: 34%)

На шахматной доске 8*8 клеток стоит слон (фигура, которая ходит по диагонали).

Требуется написать программу, которая определит: сможет ли слон дойти до заданной клетки (x, y). Если сможет, то указать за какое наименьшее количество ходов. Если количество ходов больше одного, то указать через какие промежуточные клетки он должен пройти. Если таких маршрутов несколько, то указать любой из них.

Входные данные

Входной файл INPUT.TXT содержит четыре числа m, n, x, y. (m, n) – координаты клетки, на которой находится слон, (x, y) – координаты клетки, на которую надо попасть. Числа m, n, x, y задаются в диапазоне от 1 до 8 и записываются через пробел.

Выходные данные

Выходной файл OUTPUT.TXT должен содержать в первой строке k – минимальное количество ходов, а далее в k-1 строках по 2 числа через пробел – координаты посещенных клеток. Если слон не может попасть на заданную клетку, то вывести 0.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	1 1 3 1	2 2 2
2	1 1 3 3	1
3	1 1 4 1	0

Задача №525

Сумма степеней двойки

(Время: 1 сек. Память: 16 Мб Сложность: 43%)

Любое натуральное число можно представить в виде суммы натуральных слагаемых, каждое из которых является степенью числа 2. Суммы, различающиеся лишь порядком слагаемых, считаются одинаковыми. Например, для числа 7 таких представлений 6 (4+2+1, 4+1+1+1, 2+2+2+1, 2+2+1+1+1, 2+1+1+1+1+1, 1+1+1+1+1+1+1).

Требуется написать программу, которая найдет количество способов такого представления заданного числа N.

Входные данные

Входной файл INPUT.TXT содержит число N ($1 \leq N \leq 1000$).

Выходные данные

В выходной файл OUTPUT.TXT выведите одно число – найденное количество способов представления числа N.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	4	4
2	7	6

Задача №526 Уравнение по основанию

(Время: 1 сек. Память: 16 Мб Сложность: 26%)

Запись A_X обозначает, что A есть запись числа в системе счисления по основанию X . Если $X > 10$, то для записи числа используются кроме цифр от 0 до 9 заглавные латинские буквы от A до Z . При этом условии X не может быть больше 36.

Требуется написать программу, которая по заданным A и B найдет решение уравнения $A_X = B$, либо сообщит об отсутствии у него решений.

Входные данные

Входной файл INPUT.TXT содержит в первой строке число A , во второй число B ($1 \leq B \leq 10^7$).

Выходные данные

В выходной файл OUTPUT.TXT выведите либо наименьшее основание системы счисления, удовлетворяющее уравнению, либо 0, если уравнение не имеет решений.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	A1 161	16
2	201 26	0

Задача №527 Алгоритм Евклида

(Время: 1 сек. Память: 16 Мб Сложность: 40%)

Дима недавно начал изучать информатику. Одним из первых алгоритмов, который он изучил, был алгоритм Евклида для нахождения наибольшего общего делителя (НОД) двух чисел. Напомним, что наибольшим общим делителем двух чисел a и b называется наибольшее натуральное число x , такое, что и число a , и число b делится на него без остатка.

Алгоритм Евклида заключается в следующем:

1. Пусть a, b – числа, НОД которых надо найти.
2. Если $b = 0$, то число a – искомый НОД.
3. Если $b > a$, то необходимо поменять местами числа a и b .
4. Присвоить числу a значение $a - b$.
5. Вернуться к шагу 2.

Дима достаточно быстро освоил алгоритм Евклида и вычислил с его помощью много наибольших общих делителей. Поняв, что надо дальше совершенствоваться, ему пришла идея решить новую задачу. Пусть заданы числа a, b, c и d . Требуется узнать, наступит ли в процессе реализации алгоритма Евклида для заданной пары чисел (a, b) такой момент, когда число a будет равно c , а число b будет равно d .

Требуется написать программу, которая решает эту задачу.

Входные данные

Первая строка входного файла INPUT.TXT содержит количество наборов входных данных k ($1 \leq k \leq 100$). Далее идут описания этих наборов. Каждое описание состоит из двух строк. Первая из них содержит два целых числа: a, b ($1 \leq a, b \leq 10^{18}$). Вторая строка – два целых числа: c, d ($1 \leq c, d \leq 10^{18}$).

Выходные данные

Для каждого набора входных данных выведите в отдельной строке выходного файла OUTPUT.TXT слово «YES», если в процессе применения алгоритма Евклида к паре чисел (a, b) в какой-то момент получается пара (c, d) , или слово «NO» – в противном случае.

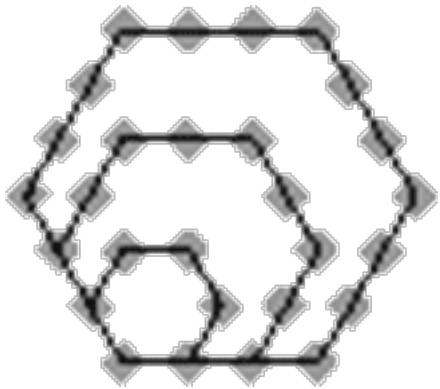
Пример

№	INPUT.TXT	OUTPUT.TXT
1	2 20 10 10 10 10 7 2 4	YES NO

Задача №528 Замок

(Время: 1 сек. Память: 16 Мб Сложность: 29%)

Замок состоит из K уровней. Каждый уровень - это правильный N -угольник, угол которого совпадает с углом предыдущего. На сторонах первого уровня находится по две комнаты (в углах), на сторонах каждого следующего - на одну больше. Сколько в замке комнат?



Входные данные

В первой строке входного файла INPUT.TXT указаны два целых числа N и K ($3 \leq N \leq 10^6$; $1 \leq K \leq 10^6$).

Выходные данные

В выходной файл OUTPUT.TXT выведите целое число - количество комнат в замке.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	6 3	28

Задача №529

Длина отрезка

(Время: 1 сек. Память: 16 Мб Сложность: 12%)

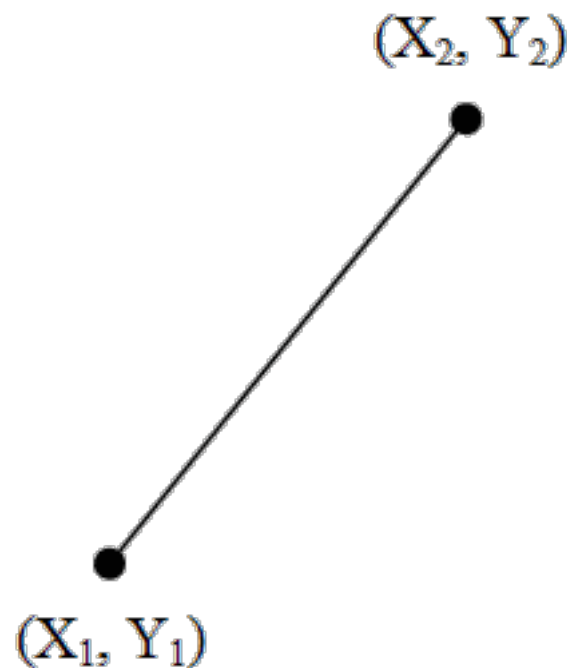
Отрезок задан координатами своих концевых точек. Требуется вычислить длину этого отрезка.

Входные данные

Входной файл INPUT.TXT содержит координаты концов отрезка в формате $X_1 Y_1 X_2 Y_2$. Все координаты – целые числа, не превышающие 1000 по абсолютной величине.

Выходные данные

В выходной файл OUTPUT.TXT выведите длину отрезка с точностью 10^{-5} .



Пример

№	INPUT.TXT	OUTPUT.TXT
1	3 4 8 4	5

Задача №530

Черно-белая графика

(Время: 1 сек. Память: 16 Мб Сложность: 26%)

Одна из базовых задач компьютерной графики – обработка черно-белых изображений. Изображения можно представить в виде прямоугольников шириной w и высотой h , разбитых на $w \times h$ единичных квадратов, каждый из которых имеет либо белый, либо черный цвет. Такие единичные квадраты называются пикселями. В памяти компьютера сами изображения хранятся в виде прямоугольных таблиц, содержащих нули и единицы.

Во многих областях очень часто возникает задача комбинации изображений. Одним из простейших методов комбинации, который используется при работе с черно-белыми изображениями, является попиксельное применение некоторой логической операции. Это означает, что значение пиксела результата получается применением этой логической операции к соответствующим пикселям аргументов. Логическая операция от двух аргументов обычно задается таблицей истинности, которая содержит значения операции для всех возможных комбинаций аргументов.

Например, для операции «исключающее ИЛИ» эта таблица выглядит так:

Первый аргумент	Второй аргумент	Результат
0	0	0
0	1	1
1	0	1
1	1	0

Требуется написать программу, которая вычислит результат попиксельного применения

заданной логической операции к двум черно-белым изображениям одинакового размера.

Входные данные

Первая строка входного файла INPUT.TXT содержит два целых числа w и h ($1 \leq w, h \leq 100$). Последующие h строк описывают первое изображение и каждая из этих строк содержит w символов, каждый из которых равен нулю или единице. Далее следует описание второго изображения в аналогичном формате. Последняя строка входного файла содержит описание логической операции в виде четырех чисел, каждое из которых – ноль или единица. Первое из них есть результат применения логической операции в случае, если оба аргумента – нули, второе – результат в случае, если первый аргумент – ноль, второй – единица, третье – результат в случае, если первый аргумент – единица, второй – ноль, а четвертый – в случае, если оба аргумента – единицы.

Выходные данные

В выходной файл OUTPUT.TXT необходимо вывести результат применения заданной логической операции к изображениям в том же формате, в котором изображения заданы во входном файле.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	5 3	11110
	01000	11100
	11110	11110
	01000	
	10110	
	00010	
	10110	
	0110	

Задача №531 Газон

(Время: 1 сек. Память: 16 Мб Сложность: 36%)

Фермер Иван с юности следит за своим газоном. Газон можно считать плоскостью, на которой в каждой точке с целыми координатами растет один пучок травы.

В одно из воскресений Иван воспользовался газонокосилкой и постриг некоторый прямоугольный участок газона. Стороны этого участка параллельны осям координат, а две противоположные вершины расположены в точках (x_1, y_1) и (x_2, y_2) . Следует отметить, что пучки травы, находящиеся на границе этого прямоугольника, также были пострижены.

Довольный результатом Иван купил и установил на газоне дождевальную установку. Она была размещена в точке с координатами (x_3, y_3) и имела радиус действия струи r . Таким образом, установка начала поливать все пучки, расстояние от которых до точки (x_3, y_3) не превышало r .

Все было хорошо, но Ивана заинтересовал следующий вопрос: сколько пучков травы оказалось и пострижено, и полито в это воскресенье?

Требуется написать программу, которая позволит дать ответ на вопрос Ивана.

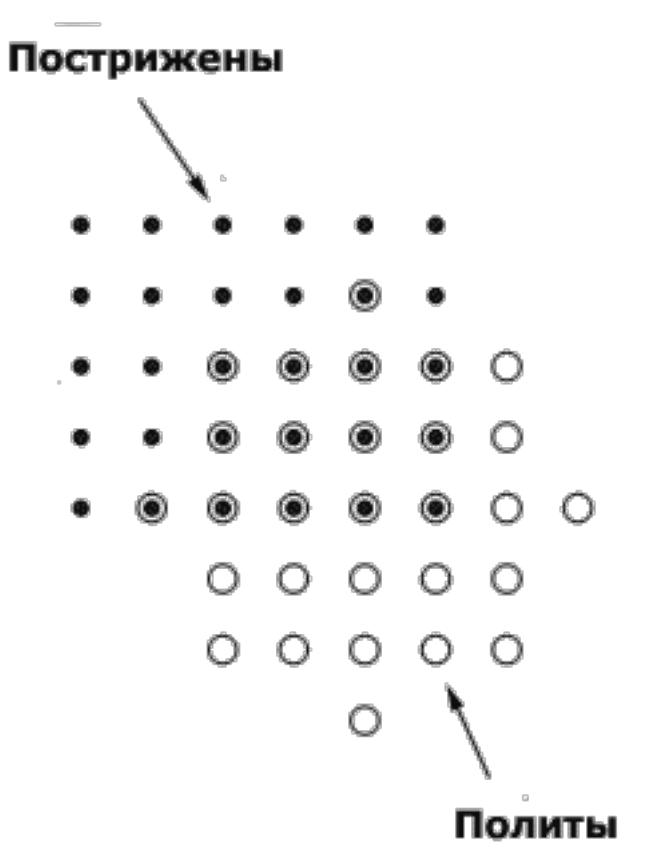
Входные данные

Первая строка входного файла INPUT.TXT содержит четыре целых числа x_1, y_1, x_2, y_2 ($-100\,000 \leq x_1 < x_2 \leq 100\,000$; $-100\,000 \leq y_1 < y_2 \leq 100\,000$). Во второй строке записаны три целых числа x_3, y_3, r ($-100\,000 \leq x_3, y_3 \leq 100\,000$; $1 \leq r \leq 100\,000$)

Выходные данные

В выходной файл OUTPUT.TXT необходимо вывести одно целое число – число пучков травы, которые были и пострижены, и политы.

Пример



№	INPUT.TXT	OUTPUT.TXT
1	0 0 5 4 4 0 3	14

Задача №532 Трамвай

(Время: 3 сек. Память: 16 Мб Сложность: 74%)

С окраины в центр города каждое утро по одному маршруту едут в трамвае N человек. За долгое время поездок они достаточно хорошо узнали друг друга. Чтобы никому не было обидно, они захотели решить, кто из них и между какими остановками маршрута должен сидеть, а кто должен стоять. Все остановки пронумерованы от 1 до P .

Один из пассажиров оказался знатоком теории математического моделирования. Он предложил рассмотреть значение суммарного удовлетворения пассажиров. Для каждого i -го пассажира он оценил две величины – a_i и b_i . Если в течение одного переезда между остановками пассажир сидит, то к суммарному удовлетворению прибавляется a_i , если же он стоит, то прибавляется b_i .

Всего в трамвае M сидячих мест. Вставать и садиться пассажиры могут мгновенно на любой остановке. Кроме того, некоторые пассажиры предпочитают ехать стоя, даже если в трамвае есть свободные места (для них $a_i < b_i$).

Требуется написать программу, которая вычисляет значение максимально достижимого суммарного удовлетворения, если для каждого i -го пассажира известны величины a_i и b_i , а также номера остановок, на которых он садится и выходит из трамвая.

Входные данные

Первая строка входного файла INPUT.TXT содержит разделенные пробелом три целых числа N, M и P – число пассажиров, число сидячих мест и число остановок на маршруте соответственно ($1 \leq N, M, P \leq 100\,000$; $2 \leq P$).

Каждая из следующих N строк содержит информацию об очередном пассажире в виде четырех целых чисел a_i, b_i, c_i, d_i , где первые два числа определяют вклад в параметр счастья, третье – номер остановки, на которой пассажир садится в трамвай, и последнее – номер остановки, на которой он выходит из трамвая ($-10^6 \leq a_i, b_i \leq 10^6$; $1 \leq c_i < d_i \leq P$).

Выходные данные

В выходной файл OUTPUT.TXT необходимо вывести одно целое число — максимальное суммарное удовлетворение, которого могут добиться пассажиры.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	4 2 4 10 -10 2 3 -1 -3 1 4 6 -6 1 3 7 4 2 4	28

Пояснение к примеру

Максимальное суммарное довольство достигается следующим образом:

- На первой остановке входят и садятся второй и третий пассажиры;
- На второй остановке входят первый и четвертый пассажиры, второй уступает место первому;
- На третьей остановке встают и выходят первый и третий пассажиры, второй и четвертый садятся на их места;
- На четвертой остановке выходят второй и четвертый пассажиры.

Задача №533 Треугольники - 3

(Время: 2 сек. Память: 16 Мб Сложность: 66%)

Петя достаточно давно занимается в математическом кружке, поэтому он уже успел освоить не только правила выполнения простейших операций, но и такое достаточно сложное понятие как симметрия. Для того, чтобы получше изучить симметрию Петя решил начать с наиболее простых геометрических фигур – треугольников. Он скоро понял, что осевой симметрией обладают так называемые равнобедренные треугольники. Поэтому теперь Петя ищет везде такие треугольники.

Напомним, что треугольник называется равнобедренным, если его площадь положительна, и у него есть хотя бы две равные стороны.

Недавно Петя, зайдя в класс, увидел, что на доске нарисовано n точек. Разумеется, он сразу задумался, сколько существует троек из этих точек, которые являются вершинами равнобедренных треугольников.

Требуется написать программу, решающую указанную задачу.

Входные данные

Входной файл INPUT.TXT содержит целое число N ($3 \leq N \leq 1500$). Каждая из последующих строк содержит по два целых числа – x_i и y_i – координаты i -ой точки. Координаты точек не превосходят 10^9 по абсолютной величине. Среди заданных точек нет совпадающих.

Выходные данные

В выходной файл OUTPUT.TXT выведите ответ на задачу.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	3 0 0 2 2 -2 2	1
2	4 0 0 1 1 1 0 0 1	4

Задача №534
Клавиатура - 2

(Время: 1 сек. Память: 16 Мб Сложность: 25%)

Всем известно, что со временем клавиатура изнашивается, и клавиши на ней начинают залипать. Конечно, некоторое время такую клавиатуру еще можно использовать, но для нажатий клавиш приходится использовать большую силу.

При изготовлении клавиатуры изначально для каждой клавиши задается количество нажатий, которое она должна выдерживать. Если знать эти величины для используемой клавиатуры, то для определенной последовательности нажатых клавиш можно определить, какие клавиши в процессе их использования сломаются, а какие – нет.

Требуется написать программу, определяющую, какие клавиши сломаются в процессе заданного варианта эксплуатации клавиатуры.

Входные данные

Первая строка входного файла INPUT.TXT содержит целое число N ($1 \leq N \leq 100$) – количество клавиш на клавиатуре. Вторая строка содержит n целых чисел – c_1, c_2, \dots, c_N , где c_i ($1 \leq c_i \leq 100000$) – количество нажатий, выдерживаемых i -ой клавишей. Третья строка содержит целое число K ($1 \leq K \leq 100000$) – общее количество нажатий клавиш, и последняя строка содержит K целых чисел p_j ($1 \leq p_j \leq N$) – последовательность нажатых клавиш.

Выходные данные

В выходной файл OUTPUT.TXT необходимо вывести N строк, содержащих информацию об исправности клавиш. Если i -ая клавиша сломалась, то i -ая строка должна содержать слово "yes" (без кавычек), если же клавиша работоспособна – слово "no".

Пример

№	INPUT.TXT	OUTPUT.TXT
1	5 1 50 3 4 3 16 1 2 3 4 5 1 3 3 4 5 5 5 5 5 4 5	yes no no no yes

Задача №535

Неправильное сложение

(Время: 1 сек. Память: 16 Мб Сложность: 41%)

Володя написал программу, которая складывает в столбик два числа. К сожалению, он не разобрался, как правильно переносить единицу из одного разряда в следующий. Поэтому программа стала выполняться следующим образом. Сначала она складывает последние цифры обоих чисел и записывает результат, как в случае, если он однозначный, так и в случае, если он двузначный. Затем программа складывает предпоследние цифры обоих чисел и результат сложения приписывает слева к результату предыдущего сложения. Далее процесс повторяется для всех разрядов. Если в одном числе цифр меньше, чем в другом, то программа размещает нули в соответствующих разрядах более короткого числа.

Федя хочет доказать Володе, что его способ сложения не обладает свойством ассоциативности. В частности, Федя утверждает, что существуют три числа, для которых важен порядок, в котором их складывают (при этом разрешается складывать числа в любом порядке, например можно сначала сложить первое число и последнее, а затем прибавить к ним среднее). Федя привел даже пример трех таких чисел.

Требуется написать программу, которая поможет Феде и Володе определить, верно ли утверждение, что, складывая заданные три числа в разном порядке, можно получить разные суммы.

Входные данные

Входной файл INPUT.TXT содержит в одной строке три целых числа a, b и c ($1 \leq a, b, c \leq 10^6$). Все числа в строке разделены пробелом.

Выходные данные

В первую строку выходного файла OUTPUT.TXT необходимо вывести слово YES, если данные три числа можно сложить разными способами и получить разные суммы. В противном случае, необходимо вывести слово NO.

В последующих строках необходимо вывести все возможные суммы, которые можно получить, складывая числа a, b и c . Суммы следует выводить по одной на строке и в порядке их возрастания.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	30 239 566	YES 7945 71215
2	643 733 553	NO 18129

Задача №536

Числа - 2

(Время: 1 сек. Память: 16 Мб Сложность: 45%)

Решая задачу по информатике, Вова в очередной раз допустил ошибку. Он снова вывел в выходной файл числа, забыв разделить их пробелами. Увидев полученный результат, Вова сначала огорчился, а потом задумался над следующим вопросом: сколько существует различных последовательностей неотрицательных целых чисел, таких что, если выписать их без пробелов, то получится тот же результат, что и у него. Он вспомнил также, что его программа смогла вывести

не произвольные числа, а только те, что не превосходят С и не имеют ведущих нулей.

Чтобы ответить на поставленный вопрос, Вова решил написать программу, которая позволит ему найти число различных последовательностей неотрицательных целых чисел, в каждой из которых любое число не превосходит С. Он понимал, что такое число могло быть достаточно большим, поэтому ограничился поиском только последних К цифр этого числа.

Требуется написать программу, которая покажет Вове, как можно правильно решить поставленную им задачу.

Входные данные

Первая строка входного файла INPUT.TXT содержит три целых числа – N, С и К ($1 \leq N \leq 50000$, $1 \leq C \leq 10^8$, $1 \leq K \leq 18$). Во второй строке этого файла содержится результат работы Вовиной программы, состоящий из N цифр.

Выходные данные

В выходной файл OUTPUT.TXT выведите последние К цифр искомого количества последовательностей без ведущих нулей.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	3 11 2 111	3
2	19 9 1 0123456789876543210	1
3	1 8 3 9	0

Задача №537 Перестановки - 3

(Время: 1 сек. Память: 16 Мб Сложность: 64%)

Задано множество из N различных натуральных чисел. Перестановку элементов этого множества назовем К-перестановкой, если для любых двух соседних элементов этой перестановки их наибольший общий делитель не менее К. Например, если задано множество элементов $S = \{6, 3, 9, 8\}$, то перестановка $\{8, 6, 3, 9\}$ является 2-перестановкой, а перестановка $\{6, 8, 3, 9\}$ – нет.

Перестановка $\{p_1, p_2, \dots, p_N\}$ будет лексикографически меньше перестановки $\{q_1, q_2, \dots, q_N\}$, если существует такое натуральное число i ($1 \leq i \leq N$), для которого $p_j = q_j$ при $j < i$ и $p_i < q_i$.

В качестве примера упорядочим все К-перестановки заданного выше множества в лексикографическом порядке. Например, существует ровно четыре 2-перестановки множества S: $\{3, 9, 6, 8\}$, $\{8, 6, 3, 9\}$, $\{8, 6, 9, 3\}$ и $\{9, 3, 6, 8\}$. Соответственно, первой 2-перестановкой в лексикографическом порядке является множество $\{3, 9, 6, 8\}$, а четвертой – множество $\{9, 3, 6, 8\}$.

Требуется написать программу, позволяющую найти М-ую К-перестановку в этом порядке.

Входные данные

Входной файл INPUT.TXT в первой строке содержит три натуральных числа – N ($1 \leq N \leq 16$), М и К ($1 \leq M, K \leq 10^9$). Вторая строка содержит N различных натуральных чисел, не

превосходящих 10^9 . Все числа в строках разделены пробелом.

Выходные данные

В выходной файл OUTPUT.TXT необходимо вывести М-ую К-перестановку заданного множества или -1 , если такой нет.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	4 1 2 6 8 3 9	3 9 6 8
2	4 4 2 6 8 3 9	9 3 6 8
3	4 5 2 6 8 3 9	-1

Задача №538 Радио

(Время: 1 сек. Память: 16 Мб Сложность: 52%)

Как известно, при распространении радиоволн возникает интерференция, поэтому если рядом расположены две радиопередающие станции, вещающие на одной и той же частоте, то качество радиопередач резко снижается.

Радиостанция «Радио Информатика» планирует транслировать свои программы в стране Флатландия. Министерство связи Флатландии выдало радиостанции лицензию на вещание на двух различных частотах.

Владельцы радиостанции имеют возможность транслировать свои радиoprogramмы с использованием n радиовышек, расположенных в различных точках страны. Для осуществления трансляции на каждой радиовышке требуется установить специальный передатчик – трансмиттер. Каждый передатчик можно настроить на одну из двух частот, выделенных радиостанции. Кроме частоты вещания, передатчик характеризуется также своей мощностью. Чем мощнее передатчик, тем на большее расстояние он распространяет радиоволны. Для простоты, предположим, что передатчик мощности R распространяет радиоволны на расстояние, равное R километрам.

Все передатчики, установленные на вышках, должны, согласно инструкции министерства, иметь одну и ту же мощность. Чтобы программы радиостанции могли приниматься на как можно большей территории, мощность передатчиков должна быть как можно большей. С другой стороны, необходимо, чтобы прием передач был качественным на всей территории Флатландии. Прием передач считается качественным, если не существует такого участка ненулевой площади, на который радиоволны радиостанции «Радио Информатика» приходят на одной частоте одновременно с двух вышек.

Требуется написать программу, которая определяет, какую максимальную мощность можно было установить на всех передатчиках, позволяющую выбрать на каждом передатчике такую одну из двух частот передачи, чтобы прием был качественным на всей территории Флатландии.

Входные данные

Первая строка входного файла INPUT.TXT содержит число N – количество вышек ($3 \leq N \leq 1200$). Последующие N строк содержат по два целых числа — координаты вышек. Координаты заданы в километрах и не превышают 10^4 по модулю. Все точки, в которых расположены вышки,

различны. Все числа в строках разделены пробелом.

Выходные данные

В первой строке выходного файла OUTPUT.TXT выводится вещественное число — искомая мощность передатчиков. Во второй строке выводятся N чисел, где i-е число должно быть равно 1, если соответствующий передатчик должен вещать на первой частоте, и 2, если на второй. Ответ должен быть выведен с точностью, не меньшей 10^{-8} .

Пример

№	INPUT.TXT	OUTPUT.TXT
1	4	0.70710678118654752
	0 0	1 2 2 1
	0 1	
	1 0	
	1 1	

Задача №539 Торт

(Время: 1 сек. Память: 16 Мб Сложность: 6%)

На свой день рождения Петя купил красивый и вкусный торт, который имел идеально круглую форму. Петя не знал, сколько гостей придет на его день рождения, поэтому вынужден был разработать алгоритм, согласно которому он сможет быстро разрезать торт на N равных частей. Следует учесть, что разрезы торта можно производить как по радиусу, так и по диаметру.

Помогите Пете решить эту задачу, определив наименьшее число разрезов торта по заданному числу гостей.

Входные данные

Входной файл INPUT.TXT содержит натуральное число N – число гостей, включая самого виновника торжества ($N \leq 1000$).

Выходные данные

В выходной файл OUTPUT.TXT выведите минимально возможное число разрезов торта.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	2	1
2	3	3

Задача №540 Таблица

(Время: 1 сек. Память: 16 Мб Сложность: 45%)

Рассмотрим прямоугольную таблицу размером $n \times m$. Занумеруем строки таблицы числами от 1 до n, а столбцы – числами от 1 до m. Будем такую таблицу последовательно заполнять числами следующим образом.

Обозначим через a_{ij} число, стоящее на пересечении i -ой строки и j -ого столбца. Первая строка таблицы заполняется заданными числами – $a_{11}, a_{12}, \dots, a_{1m}$. Затем заполняются строки с номерами от 2 до n . Число a_{ij} вычисляется как сумма всех чисел таблицы, находящихся в «треугольнике» над элементом a_{ij} . Все вычисления при этом выполняются по модулю r .

				a_{ij}		

Более точно, значение a_{ij} вычисляется по следующей формуле:

$$a_{i,j} = \left(\sum_{k=1}^{i-1} \sum_{\substack{f=i-k \\ 1 \leq f \leq m}} a_{i-k,f} \right) \bmod r$$

Например, если таблица состоит из трех строк и четырех столбцов, и первая строка состоит из чисел 2,3,4,5, а $r = 40$ то для этих исходных данных таблица будет выглядеть следующим образом (взятие по модулю показано только там, где оно приводит к изменению числа):

2	3	4	5
$5 = 2 + 3$	$9 = 2 + 3 + 4$	$12 = 3 + 4 + 5$	$9 = 4 + 5$
$23 = 2 + 3 + 4 + 5 + 9$	$0 = (2 + 3 + 4 + 5 + 5 + 9 + 12) \bmod 40$ $40 = 40 \bmod 40$	$4 = (2 + 3 + 4 + 5 + 9 + 12 + 9) \bmod 40$ $= 44 \bmod 40$	$33 = 3 + 4 + 5 + 12 + 9$

Требуется написать программу, которая по заданной первой строке таблицы ($a_{11}, a_{12}, \dots, a_{1m}$), вычисляет последнюю строку, как описано выше.

Входные данные

Первая строка входного файла INPUT.TXT содержит числа n, m и r ($2 \leq n, m \leq 2000, 2 \leq r \leq 10^9$) – число строк и столбцов таблицы соответственно, а так же число, по модулю которого надо посчитать ответ. Следующая строка содержит m целых чисел – первую строку таблицы: $a_{11}, a_{12}, \dots, a_{1m}$. Все a_{1i} неотрицательны и не превосходят 10^9 .

Выходные данные

В первой строке выходного файла OUTPUT.TXT необходимо вывести m чисел – последнюю строку таблицы: $a_{n1}, a_{n2}, \dots, a_{nm}$.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	2 3 10 1 2 3	3 6 5
2	3 3 10 1 1 1	8 0 8
3	3 4 40 2 3 4 5	23 0 4 33

Задача №541

Две строки

(Время: 1 сек. Память: 16 Мб Сложность: 47%)

Циклическим сдвигом строки $s_1s_2\dots s_n$ на k позиций назовем строку $s_{k+1}s_{k+2}\dots s_ns_1\dots s_k$. Например, циклическим сдвигом строки «abcde» на две позиции является строка «cdeab». В этой задаче далее будут рассматриваться только строки, состоящие из десятичных цифр от 0 до 9. Произвольной такой строке, первый символ которой не является нулем, можно сопоставить число, десятичной записью которого она является. Строкам, которые начинаются с нуля, никакое число сопоставляться не будет. Например, строке 123 сопоставляется число сто двадцать три, а строке 0123 не сопоставляется никакое число.

Пусть заданы две строки: s и t . Обозначим как S набор всех циклических сдвигов строки s , а как T – набор всех циклических сдвигов строки t . Например, если $s = \text{«1234»}$, то S содержит строки «1234», «2341», «3412», «4123». Обозначим также как $\text{NUM}(A)$ набор чисел, соответствующих строкам из набора A .

Требуется написать программу, которая по строкам s и t определит, максимальное число, представимое в виде разности $(x - y)$, где x принадлежит $\text{NUM}(S)$, а y принадлежит $\text{NUM}(T)$. Например, если $s = \text{«25»}$, $t = \text{«12»}$, то $\text{NUM}(S)$ содержит числа 25 и 52, $\text{NUM}(T)$ – числа 12 и 21; их попарными разностями будут: $25 - 12 = 13$, $25 - 21 = 4$, $52 - 12 = 40$, $52 - 21 = 31$. Из этих разностей максимальным числом является 40.

Входные данные

Первая строка входного файла INPUT.TXT содержит строку s , вторая строка входного файла – строку t . Обе строки непустые. Они содержат только цифры, из которых хотя бы одна не является нулем. Строки имеют длину не более 3000 символов.

Выходные данные

В выходной файл OUTPUT.TXT выведите искомое число без ведущих нулей.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	25 12	40
2	100 1	99

Задача №542

Бит-реверс

(Время: 1 сек. Память: 16 Мб Сложность: 25%)

Целое положительное число m записывается в двоичной системе счисления, разряды (в этой записи) переставляются в обратном порядке и число переводится в десятичную систему счисления. Получившееся число принимается за значение функции $B(m)$.

Требуется написать программу, которая для заданного m вычислит $B(m)$.

Входные данные

Входной файл INPUT.TXT содержит натуральное число m ($m \leq 10^9$).

Выходные данные

В выходной файл OUTPUT.TXT выведите значение $B(m)$.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	4	1
2	6	3

Задача №543 Монеты - 2

(Время: 1 сек. Память: 16 Мб Сложность: 31%)

В N корзинах находятся золотые монеты. Корзины пронумерованы числами от 1 до N . Во всех корзинах, кроме одной, монеты весят по w граммов. В одной корзине монеты фальшивые и весят $w-d$ граммов. Волшебник берет 1 монету из первой корзины, 2 монеты из второй корзины, и так далее, и, наконец, $N-1$ монету из $(N-1)$ -й корзины. Из N -й корзины он не берет ничего. Он взвешивает взятые монеты и сразу указывает на корзину с фальшивыми монетами.

Требуется написать программу, которая выполняет это волшебство.

Входные данные

Входной файл INPUT.TXT содержит четыре целых числа: N , w , d и P – суммарного веса отобранных монет ($2 \leq N \leq 8000$, $1 \leq d < w \leq 30$).

Выходные данные

В выходной файл OUTPUT.TXT выведите номер корзины с фальшивыми монетами.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	10 25 8 1109	2
2	10 25 8 1125	10
3	8000 30 12 959879400	50

Задача №544 Мячик на лестнице

(Время: 1 сек. Память: 16 Мб Сложность: 42%)

На вершине лесенки, содержащей N ступенек, находится мячик, который начинает прыгать по ним вниз, к основанию. Мячик может прыгнуть на следующую ступеньку, на ступеньку через одну или через две. То есть, если мячик лежит на 8-ой ступеньке, то он может переместиться на 5-ую, 6-ую или 7-ую.

Требуется написать программу, которая определит число всевозможных "маршрутов" мячика с вершины на землю.

Входные данные

Входной файл INPUT.TXT содержит число N ($0 < N \leq 70$).

Выходные данные

Выходной файл OUTPUT.TXT должен содержать искомое число.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	1	1
2	4	7

Задача №545 Задача Пифагора

(Время: 1 сек. Память: 16 Мб Сложность: 48%)

Пифагор заказал ремесленнику изготовить несколько прямоугольных треугольников из ценных пород дерева для использования на занятиях по геометрии, но ремесленник перепутал размеры, и треугольники получились не прямоугольные. Чтобы не выбрасывать испорченный ценный материал, ремесленник решил переделать получившиеся треугольники в прямоугольные, постаравшись максимизировать их площади.

Требуется написать программу, которая по размерам сторон треугольника находит максимальную площадь прямоугольного треугольника, который можно вырезать из этого треугольника.

Входные данные

Входной файл INPUT.TXT содержит в первой строке три числа, разделенных пробелами – длины сторон треугольника. Все числа вещественные, больше 0 и меньше 1000.

Выходные данные

В выходной файл OUTPUT.TXT выведите максимальную площадь прямоугольного треугольника, получаемого из заданного треугольника, с точностью не хуже, чем 10^{-5} .

Пример

№	INPUT.TXT	OUTPUT.TXT
1	10.0 10.0 10.0	25.00000

Задача №546 Печать буклета

(Время: 1 сек. Память: 16 Мб Сложность: 29%)

Когда происходит печать документа, обычно первая страница печатается первой, вторая - второй, третья - третьей и так далее до конца. Но иногда при создании буклета на одном листе печатаются четыре страницы: две - на лицевой стороне и две - на обратной. Когда вы сложите все листы по порядку и согнете их пополам, страницы будут идти в правильном порядке, как у обычной книги. Например, 4-страничный буклет должен быть напечатан на одном листе бумаги: лицевая сторона должна содержать сначала страницу 4, потом - 1, обратная - 2 и 3.

Если в буклете число страниц не кратно четырем, то в конце можно добавить несколько пустых страниц, но так, чтобы количество листов бумаги при этом было минимально возможным.

Требуется написать программу, которая по заданному числу страниц в буклете генерирует порядок его печати.

Входные данные

Во входном файле INPUT.TXT записано количество страниц в буклете - натуральное число, не превышающее 500.

Выходные данные

В выходной файл OUTPUT.TXT необходимо выдать порядок печати данного буклета - последовательность команд, каждая из которых располагается в отдельной строке и состоит из четырех чисел. Числа разделяются пробелом и обозначают следующее: номер листа, на котором происходит печать, сторону: 1 - если печать происходит на лицевой стороне, и 2 - если на обратной, два оставшихся числа - номера страниц буклета, которые должны быть напечатаны с левой и правой сторон.

Пустая страница задается числом 0. Если целая сторона должна быть оставлена пустой, команду для ее печати выводить не обязательно.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	4	1 1 4 1 1 2 2 3

Задача №547 Период дроби

(Время: 1 сек. Память: 16 Мб Сложность: 35%)

Рассмотрим дробь $1/n$, $n > 1$. Как известно, цифры в её десятичной записи начиная с некоторого места повторяются. Минимальную по длине повторяющуюся (без промежутков) часть называют периодом. Минимальную по длине часть после запятой, которая не входит ни в один период, называют предпериодом. Например:

- $n=3$: $1/3=0,(3)$, предпериод пустой "", его длина 0, период '3', его длина 1;
- $n=5$: $1/5=0,2(0)$, предпериод '2', длина 1, период '0', длина 1;
- $n=28$: $1/28=0,03(571428)$, предпериод '03', длина 2, период '571428', длина 6.

Требуется написать программу, которая по заданному n найдет длину предпериода и длину периода у дроби $1/n$.

Входные данные

Входной файл INPUT.TXT содержит одно натуральное число n ($2 \leq n \leq 10^6$).

Выходные данные

Выходной файл OUTPUT.TXT должен содержать два числа – сначала длина предпериода, потом длина периода дроби $1/n$. Числа отделить друг от друга пробелом.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	3	0 1
2	5	1 1
3	28	2 6

Задача №548 Минимальное число

(Время: 1 сек. Память: 16 Мб Сложность: 25%)

Требуется написать программу, которая из цифр двух натуральных чисел создает наименьшее возможное число, сохраняя при этом порядок следования цифр в этих числах.

Входные данные

Входной файл INPUT.TXT содержит два натуральных числа, записанных в двух строках. Числа больше нуля и меньше 10²⁵⁵.

Выходные данные

В единственную строку выходного файла OUTPUT.TXT нужно вывести наименьшее возможное число, удовлетворяющее условию задачи.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	125 34	12345

Задача №549 Факториальная система

(Время: 1 сек. Память: 16 Мб Сложность: 40%)

Известно, что любую рациональную дробь вида p/q можно представить в виде $(0 < p < q)$:

$$\frac{p}{q} = \frac{a_2}{2!} + \frac{a_3}{3!} + \dots + \frac{a_n}{n!}$$

где $0 \leq a_i < i$, $i=2, 3, \dots, n$ для некоторого конечного n . Такое представление называется факториальным.

Требуется написать программу, которая по заданной рациональной дроби находит ее факториальное представление.

Входные данные

Во входном файле INPUT.TXT записаны через пробел два натуральных числа p и q ($p < q < 1000$).

Выходные данные

В выходной файл OUTPUT.TXT нужно в первой строке вывести число n - количество значащих разрядов плюс 1 в факториальном представлении дроби. В следующих строках располагаются неотрицательные целые числа a_2, a_3, \dots, a_n . Известно, что $n < 1000$.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	1 5	5 0 1 0 4

Задача №550 День программиста

(Время: 1 сек. Память: 16 Мб Сложность: 13%)

День программиста отмечается в 255-й день года (при этом 1 января считается нулевым днем). Требуется написать программу, которая определит дату (месяц и число григорианского календаря), на которую приходится День программиста в заданном году.

В григорианском календаре високосным является:

- год, номер которого делится нацело на 400
- год, номер которого делится на 4, но не делится на 100

Входные данные

В единственной строке входного файла INPUT.TXT записано целое число от 1 до 9999 включительно, которое обозначает номер года нашей эры.

Выходные данные

В единственную строку выходного файла OUTPUT.TXT нужно вывести дату Дня программиста в формате DD/MM/YYYY, где DD — число, MM — номер месяца (01 — январь, 02 — февраль, ..., 12 — декабрь), YYYY — год в десятичной записи.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	2000	12/09/2000
2	2009	13/09/2009

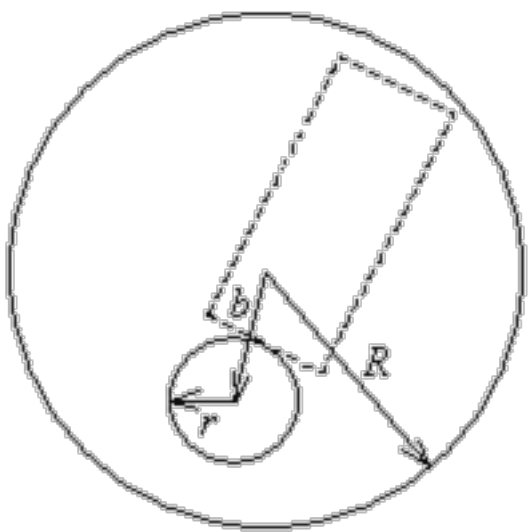
Задача №551 Заяц и дерево

(Время: 1 сек. Память: 16 Мб Сложность: 29%)

После строительства на выбранной поляне частокола для обороны от хищников Заяц обнаружил, что внутри ограды оказалось огромное засохшее дерево, которое может помешать строительству домика. Зайцу хочется спилить это дерево “под самый корешок” таким образом, чтобы дерево при падении не повредило ограду.

Заяц обрубил все ветки, и от дерева остался цилиндр высотой h и радиусом r . Ограда имеет

форму окружности с радиусом R и по высоте больше диаметра дерева. Центр дерева находится на расстоянии b от центра ограды. Заяц может повалить дерево в любую нужную ему сторону. После падения нижний край поваленного дерева касается (как касательная) окружности основания дерева, как показано на рисунке.



Требуется написать программу, которая определит, сможет ли заяц повалить спиленное дерево, не повредив ограду.

Входные данные

В единственной строке входного файла INPUT.TXT записано четыре целых числа через один пробел: R, r, h и b ($1 \leq r \leq r+b, R \leq 100, 1 \leq h \leq 1000$).

Выходные данные

В выходной файл OUTPUT.TXT нужно вывести слово YES, если существует направление падения дерева, при котором ограда останется неповрежденной, и NO в противном случае.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	90 3 50 10	YES

Задача №552 Зоопарк

(Время: 1 сек. Память: 16 Мб Сложность: 45%)

В городском зоопарке содержатся животные n разных видов. Для участия в международной выставке «Три твари» зоопарк должен представить трех животных различных видов.

Требуется написать программу, которая вычислит число способов выбрать трех животных для участия в выставке.

Например, если в зоопарке два медведя, тигр, лев и пингвин, то есть семь способов выбрать трех животных:

1. первый медведь, тигр и лев;
2. первый медведь, тигр и пингвин;
3. первый медведь, лев и пингвин;
4. второй медведь, тигр и лев;
5. второй медведь, тигр и пингвин;
6. второй медведь, лев и пингвин;
7. тигр, лев и пингвин.

Входные данные

Входной текстовый файл INPUT.TXT содержит в первой строке натуральное число n – количество видов животных в городском зоопарке ($1 \leq n \leq 1000$). Во второй строке через пробел записаны n натуральных чисел – количество животных соответствующего вида. Число животных каждого вида не превышает 1000.

Выходные данные

Выходной текстовый файл OUTPUT.TXT должен содержать одно число – количество способов выбрать трех животных для международной выставки.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	4 2 1 1 1	7
2	3 100 100 100	1000000

Задача №553 Объединение блоков

(Время: 1 сек. Память: 16 Мб Сложность: 43%)

Изделие изготавливают из n блоков, каждый из которых имеет два технологических параметра – m_i и k_i . Известно, что $k_i = m_{i+1}$, $i = 1, 2, \dots, n-1$. При этом условии два последовательных блока i и $i+1$ можно объединять в один новый, который будет иметь технологические параметры – m_i и k_{i+1} , и на это потребуется $m_i * k_{i+1}$ технологических операций. Новый блок можно опять объединять с одним из соседних и так далее. Меняя порядок сборки блоков можно добиться уменьшения количества технологических операций.

Поясним это на примере трех блоков: 34 и 29, 29 и 4, 4 и 15. Если собрать вначале 2 и 3 блок, а затем присоединить собранное к первому, то потребуется $29 * 15 + 34 * 15 = 435 + 510 = 945$ операций. Если собрать вначале блок из 1 и 2 исходных блоков, а затем присоединить 3 блок, то потребуется $34 * 4 + 34 * 15 = 136 + 510 = 646$ операций.

Требуется написать программу, которая найдет минимальное число технологических операций для изготовления изделия.

Входные данные

Входной файл INPUT.TXT содержит в первой строке число n – количество блоков ($1 \leq n \leq 100$). Последующие n строк содержат пары чисел (разделенных пробелом) – технологические параметры блоков. Технологические параметры – целые неотрицательные числа, не превышающие 100.

Выходные данные

Выходной текстовый файл OUTPUT.TXT должен содержать одно число – минимальное число технологических операций.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	3 34 29 29 4 4 15	646

Задача №554 Пицца

(Время: 1 сек. Память: 16 Мб Сложность: 20%)

Пицца – любимое лакомство Васи, он постоянно покупает и с удовольствием употребляет различные сорта этого великолепного блюда. Однажды, в очередной раз, разрезая круглую пиццу на несколько частей, Вася задумался: на какое максимальное количество частей можно разрезать пиццу за N прямых разрезов?

Помогите Васе решить эту задачу, определив максимальное число не обязательно равных кусков, которые может получить Вася, разрезая пиццу таким образом.

Входные данные

Входной файл INPUT.TXT содержит натуральное число N – число прямых разрезов пиццы ($N \leq 1000$).

Выходные данные

В выходной файл OUTPUT.TXT выведите ответ на задачу.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	2	4
2	3	7

Задача №555

Вычисление сложности программы

(Время: 1 сек. Память: 64 Мб Сложность: 76%)

Аня решила стать чемпионкой мира по программированию. Для этого ей, кроме всего прочего, потребовалось научиться вычислять время работы программ. Во многих программах основное время работы занимают вложенные циклы. Поэтому Аня хочет научиться узнавать, сколько операций будет выполнено внутри последовательности вложенных циклов.

Пока Аня еще только учится, поэтому все программы выглядят как последовательность вложенных циклов, причем все циклы выполняются от некоторого числа или переменной до некоторого другого числа. Ей требуется найти, сколько раз будет выполнена операция внутри самого вложенного цикла. Для этого она создала счетчик, проинициализировала его нулем, и увеличивает его на единицу каждый раз, входя в самый вложенный цикл.

Заметьте, что Аня пока не знает, что в качестве верхней границы в цикле может служить переменная. В качестве нижней границы может выступать как переменная, так и фиксированное значение. Если в каком-то цикле левая граница больше, чем правая, то цикл не будет выполняться вообще.

Так как программы у Ани иногда работают очень долго, то она попросила своего друга узнать ответ, когда же и он не смог этого посчитать, она попросила об этом вас. Помогите ей.

Напишите программу, которая находит, какое значение будет в счетчике в конце работы Аниной программы, если счетчик хранился в переменной целого 32-битного беззнакового типа.

Входные данные

В первой строке входного файла INPUT.TXT находится число $1 \leq N \leq 3\,000$ - количество циклов. Далее в N строках написано по два целых числа в каждой. В (i+1)-й строке входного файла записаны левая и правая граница i-го цикла: L_i и R_i ($1 \leq L_i \leq 3\,000$, $0 \leq R_i \leq 3\,000$). При этом

если $L_i < 0$, то это обозначает, что цикл начинается со значения переменной $(-L_i)$ -го цикла, в противном случае цикл начинается со значения L_i .

Выходные данные

В выходной файл OUTPUT.TXT выведите значение, которое будет в счетчике в конце работы программы.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	2 1 100 -1 100	5050
2	2 4 6 -1 3	0
3	4 1 256 1 256 1 256 1 256	0
4	4 1 1000 1 1000 1 1000 1 1000	3567587328

Задача №556 Есть ли жизнь на Марсе?

(Время: 1 сек. Память: 16 Мб Сложность: 37%)

- Ты врешь, Коля! На Марсе жизни нет! Кто тебе такую чушь сказал?
- Петя. А ему сказал Саша.
- Да от Пети я в жизни правдивого слова не слышал! Ему что ни скажут, он все переврет. Да и Саше откуда знать?
- А Саше рассказал про это Владимир Алексеевич, наш учитель биологии.
- Ну, Владимиру Алексеевичу-то можно верить... Только вряд ли он так сказал, это либо Саша, либо Петя придумал. А может, это ты меня разыгрываешь?..
- Минуточку, ребята, - вмешался подошедший к спорящим учитель математики, Глеб Тимофеевич, - давайте подойдем к проблеме формально. Допустим, что все диалоги - Владимира Алексеевича с Сашей, Саши с Петей и Пети с Колей - действительно имели место. Пронумеруем ребят числами 1, 2 и 3. Предположим также, что каждый из ребят независимо друг от друга передал полученную информацию относительно жизни на Марсе верно с вероятностью p_i , а соврал с вероятностью $q_i = 1 - p_i$ для $i = 1, 2, 3$. Вероятности – это вещественные числа от нуля до единицы; событие, имеющее вероятность 0, никогда не произойдет, событие же с вероятностью 1 произойдет без всякого сомнения. Зная, что Коля после этого объявил, что жизнь на Марсе все-таки есть, найдите по данным p_i вероятность того, что так действительно сказал Владимир Алексеевич.

- А как искать эту вероятность? И что значит независимо друг от друга? – растерялись ребята.
- Независимость означает, что действие одного из ребят никак не отражается на том, как поступят другие. К примеру, Пете неважно, соврал ли Саша - в любом случае он передаст сказанное Сашей правильно с вероятностью ровно p_2 . Задача несложная, и можно рассмотреть все восемь возможных случаев. Первый случай - все ребята говорили правду, и вероятность этого случая равна $p_1 \cdot p_2 \cdot p_3$. В этом случае жизнь на Марсе, без сомнения, есть - Владимиру Алексеевичу мы верим, а ребята передали его слова правильно. Вторым случаем, когда соврал только Саша, имеет место с вероятностью $q_1 \cdot p_2 \cdot p_3$, и в этом случае жизни на Марсе нет. Далее переберем остальные шесть случаев, каждый раз перемножая соответствующие вероятности, а потом просуммируем вероятности тех случаев, в которых слова учителя переданы правильно. То, что вероятности для отдельных ребят в каждом случае надо перемножить - это и есть формальное определение независимости. Ну, в скольких случаях будет передано именно то, что говорил Владимир Алексеевич?
- В одном ...
- А вот и нет. Например, если Петя и Коля соврали, а Саша сказал правду, то истина, дважды исказившись, дойдет до нас в неизменном виде. И вообще, четное количество отрицаний, примененных к утверждению, дает само утверждение. В нашей задаче случаев с четным количеством отрицаний - четыре, и итоговая вероятность равна $p_1 \cdot p_2 \cdot p_3 + q_1 \cdot q_2 \cdot p_3 + q_1 \cdot p_2 \cdot q_3 + p_1 \cdot q_2 \cdot q_3$.
- То есть если Петя и Коля точно соврут, а Саша точно скажет правду, то от Коли мы услышим в точности то, что говорил учитель?
- Совершенно верно. А теперь решите-ка задачу для общего случая, когда ребят не трое, а n . Первому, кто решит - пятерка на следующей контрольной!

Входные данные

Входной файл INPUT.TXT содержит целое число n ($1 \leq n \leq 100$). Во второй строке через пробел записаны n вещественных чисел - это числа p_1, p_2, \dots, p_n ($0 \leq p_i \leq 1$). Числа даны с не более чем шестью десятичными знаками после запятой.

Выходные данные

В выходной файл OUTPUT.TXT выведите одно вещественное число, округленное до шести знаков после запятой - вероятность существования жизни на Марсе.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	3 1 0.1 0.9	0.18

Задача №557 Матрицы

(Время: 1 сек. Память: 16 Мб Сложность: 42%)

Аня недавно узнала, что такое квадратная матрица размерности n . Это таблица $n \times n$ с целыми числами в ячейках. Число, стоящее на пересечении i -ой строки и j -ого столбца матрицы A , кратко обозначается $A[i, j]$. Матрицы можно умножать, и Аня быстро освоила, как запрограммировать эту операцию с помощью циклов. Результатом умножения двух матриц A и B будет матрица C ,

элементы которой определяются следующим образом:

$$C[i, j] = \sum_{k=1}^n A[i, k]B[k, j]$$

Матрицы ей понадобились для конкретной задачи, в которой надо узнать определенный элемент произведения нескольких матриц. Это уже достаточно сложная задача для Ани, но она усложняется тем, что все вычисления ведутся по модулю некоторого простого числа p , то есть если при арифметических операциях получается число, большее, либо равное p , оно заменяется на остаток при делении на p .

Помогите Ане вычислить нужный ей элемент.

Входные данные

В первой строчке входного файла INPUT.TXT стоят два числа: m - количество матриц, n - размер каждой из матриц ($1 \leq m \leq 200, 1 \leq n \leq 200$). В следующей строчке содержатся номер строки и столбца, интересующего Аню элемента $1 \leq a \leq n, 1 \leq b \leq n$. В третьей строке содержится простое число $p \leq 1000$. Далее следует описание m матриц. Описание каждой матрицы состоит из n строк. В каждой из строк содержится n неотрицательных целых чисел, меньших p . Соседние числа в строке разделены пробелом, а перед каждой матрицей пропущена строка.

Выходные данные

В выходной файл OUTPUT.TXT выведите нужный Ане элемент произведения матриц.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	3 2	20
	1 2	
	2 3 9	
	1 2	
	3 4	
	4 2	
	1 3	
	1 2	
	2 1	

Задача №558

Упаковка

(Время: 1 сек. Память: 16 Мб Сложность: 65%)

Вот уже неделю Ваня решает сложную задачу: по данному числу n необходимо расположить на плоскости n кругов с радиусами $1, 2, \dots, n$ таким образом, чтобы, во-первых, каждая пара кругов не пересекалась (возможно, касаясь), а во-вторых, все эти круги помещались в большой объемлющий круг как можно меньшего радиуса. В процессе решения Ваня заметил, что, расположив достаточное количество больших кругов, можно сразу начинать искать объемлющий круг, поскольку все оставшиеся маленькие круги можно поместить в промежутках, оставшихся между большими.

Теперь Ваня хочет по данному множеству радиусов оценить, как часто между тремя попарно касающимися кругами с радиусами из этого множества можно поместить четвертый круг. Для этого он ввел рейтинг упаковываемости P : для множества радиусов $R = \{r_1, r_2, \dots, r_n\}$ рейтинг $P(R)$ равен количеству таких упорядоченных четверок индексов (i, j, k, l) , что $r_i > r_j > r_k > r_l$ и между тремя попарно касающимися кругами радиусов r_i, r_j и r_k можно поместить круг радиуса r_l так, чтобы он не пересекался с ними, возможно, касаясь. Выражение «поместить между» означает, что центр четвертого круга должен лежать внутри треугольника с вершинами в центрах первых трех кругов.

Помогите Ване посчитать рейтинг упаковываемости данного множества.

Входные данные

Входной файл INPUT.TXT содержит целое число n ($1 \leq n \leq 250$). Во второй строке через пробел записаны n различных целых чисел r_1, r_2, \dots, r_n ($1 \leq r_i \leq 250$) - элементы множества R . Гарантируется, что R непусто.

Выходные данные

В выходной файл OUTPUT.TXT выведите одно число – рейтинг упаковываемости $P(R)$.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	5 2 1 5 10 20	1

Задача №559 Сосиска в тесте

(Время: 1 сек. Память: 16 Мб Сложность: 48%)

Аня еще учится в школе, однако она уже много слышала о трудностях студенческой жизни. Ее это очень волнует, так как она задумывается о своем будущем и понимает, что после окончания школы ей придется учиться в ВУЗе. Из многочисленных рассказов она узнала, что излюбленной пищей студентов являются сосиски. А так как студенты часто испытывают недостатки в средствах, сосиски часто приходится делить. Однако никто не доверяет делению «на глаз», всегда найдется кто-нибудь, утверждающий, что его обделили.

Сосиска представляет собой практически плоскую фигуру (таковы уж студенческие столовые), которую можно описать следующим образом. Рассмотрим некоторый горизонтальный отрезок на плоскости, а также некоторый круг. Тогда, если центр круга провести по всем точкам отрезка от начальной до конечной, то любая точка, лежавшая, хотя бы в какой-то момент в круге, принадлежит сосиске. Можно описать сосиску и другим способом. Рассмотрим тот же отрезок и два круга одинакового радиуса с центрами в его крайних точках. Тогда точка принадлежит сосиске ровно в том случае, если она лежит на некотором отрезке, у которого каждый из концов лежит в одном из кругов.

Подумав, Аня понимает, что при наличии линейки можно легко отмерить нужное расстояние от левого края сосиски и вертикально отрезать необходимую часть. Но если при этом отмерять одинаковые расстояния, то те, кому достанутся крайние куски, получают меньше. Помогите Ане разместить разрезы в нужных местах сосиски, чтобы все получили поровну.

Входные данные

Входной файл INPUT.TXT содержит в первой строке два числа: l - расстояние между центрами крайних кругов и r - радиус каждого из них. Оба числа вещественные, положительные и не превосходят 10 000 и располагаются на первой строке входного файла. На второй строке находится n – количество студентов, претендующих на сосиску ($2 \leq n \leq 1000$).

Выходные данные

В выходном файле OUTPUT.TXT должно содержаться $n-1$ строка. В i -ой строке должно содержаться расстояние от левого края сосиски до правого края i куска (куски нумеруются слева направо). Все числа должны быть вычислены с точностью до 6 десятичных знаков после запятой.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	1 1 2	1.500000
2	1 1 4	0.856810 1.500000 2.143190

Задача №560 Остановки

(Время: 1 сек. Память: 16 Мб Сложность: 64%)

Представьте себе, что вы - один из проектировщиков железной дороги. Вам нужно построить N остановок, размеры которых довольно велики, но все равны между собой и равны некоторому числу l . Стоимость постройки меняется в зависимости от рода поверхности, по которой предстоит строить платформу. При этом остановки не должны располагаться от границ дороги и друг от друга дальше, чем на R метров, иначе каким-то жителям области придется далеко ходить. Также остановки не должны располагаться от границ дороги и друг к другу ближе, чем на r метров, иначе электрички не будут успевать разгоняться между остановками, и будут медленно ездить. Руководство хочет потратить на постройку остановок минимальное количество денег. Вам известны стоимости постройки метра платформы на каждом из L метров всей дороги. Стоимость i -го метра - a_i (при $1 \leq i \leq L$). Вычислите минимально возможную стоимость расстановки всех остановок.

Входные данные

В первой строке входного файла INPUT.TXT содержатся целые числа N, l и L ($1 \leq N \leq 100, 1 \leq l \leq 100, 1 \leq L \leq 20\,000$). Во второй строке содержится два целых числа: r и R ($0 \leq r \leq R \leq 1000$). В третьей строке содержится L целых чисел a_i , разделенных пробелами ($1 \leq a_i \leq 10000$).

Выходные данные

В выходной файл OUTPUT.TXT выведите целое число - искомый ответ. Если решения не существует, то следует вывести «No solution.» .

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	3 1 5 0 1	6

	3 7 1 5 2	
2	2 3 9 1 2 8 1 2 3 7 4 3 1 11	14
3	1 1 1 10 10 239	No solution.

Задача №561

Башни

(Время: 1 сек. Память: 16 Мб Сложность: 71%)

Башней называется выражение вида $a_0^{a_1^{a_2^{...^{a_k}}}}$, где $k \geq 1$, т.е. последовательное возведение в степень чисел $a_0, ..., a_k$. Отметим, что операция возведения в степень выполняется справа налево, т.е. выражение a^{b^c} вычисляется как $a^{(b^c)}$.

Аня недавно изучила алгоритмы сортировок, и теперь она умеет сортировать целые числа, вещественные числа, и даже строки. Узнав об этом, Андрюша предложил ей написать программу для сортировки башен. Как же это сделать?

Аня привыкла учиться на примерах, и поэтому ей нужна ваша работающая программа. Напишите программу, которая по заданным во входном файле башням вычисляла бы порядок, в котором нужно их поставить, чтобы они оказались расположены по возрастанию.

Входные данные

В первой строке входного файла INPUT.TXT задается число башен N ($1 \leq N \leq 50000$). Далее следуют N строк, каждая из которых содержит одну башню в формате $k_i\ a_{i0}\ a_{i1}\ \dots\ a_{ik_i}$, здесь все числа разделены пробелами. Каждое из a_{ij} - целое число в пределах от 1 до 99, кроме того, $1 \leq k_i \leq 9$.

Известно, что среди башен во входном файле нет равных, кроме того, любые две башни отличаются достаточно сильно, чтобы их можно было различить. Например, числа 10^{10} и $10^{10} + 1$ отличаются достаточно сильно, а $10^{20} + 10^5$ и 10^{20} - недостаточно. Заметьте, что значения у башен могут быть весьма велики - например, даже число $2^{2^{2^2}} = 2^{65536}$ не помещается ни в какой вещественный тип.

Выходные данные

В выходной файл OUTPUT.TXT выведите перестановку номеров башен b_1, b_2, \dots, b_N в таком порядке, что если взять сначала башню с номером b_1 , потом с номером b_2 , и т.д., то они окажутся расположенными в порядке возрастания.

Пример

№	INPUT.TXT	OUTPUT.TXT
	10 4 2 2 2 2 2 1 2 2 1 3 2	2 4 3 6 7 5 9 10 1 8

1	1	2	3		
	3	2	2	2	2
	2	2	2	2	
	1	3	3		
	3	3	3	3	3
	2	4	3	3	
	2	2	3	4	

Задача №562

Слабая К-связность

(Время: 1 сек. Память: 16 Мб Сложность: 47%)

Ане, как будущей чемпионке мира по программированию, поручили очень ответственное задание. Правительство вручает ей план постройки дорог между N городами. По плану все дороги односторонние, но между двумя городами может быть больше одной дороги, возможно, в разных направлениях. Ане необходимо вычислить минимальное такое K, что данный ей план является слабо K-связным.

Правительство называет план слабо K-связным, если выполнено следующее условие: для любых двух различных городов можно проехать от одного до другого, нарушая правила движения не более K раз. Нарушение правил - это проезд по существующей дороге в обратном направлении. Гарантируется, что между любыми двумя городами можно проехать, возможно, несколько раз нарушив правила.

Входные данные

В первой строке входного файла INPUT.TXT записаны два числа $2 \leq N \leq 300$ и $1 \leq M \leq 10^5$ - количество городов и дорог в плане. В последующих M строках даны по два числа - номера городов, в которых начинается и заканчивается соответствующая дорога.

Выходные данные

В выходной файл OUTPUT.TXT выведите минимальное K, такое, что данный во входном файле план является слабо K-связным.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	3 2 1 2 1 3	1
2	4 4 2 4 1 3 4 1 3 2	0

Задача №563

Задача про XOR

(Время: 1 сек. Память: 16 Мб Сложность: 69%)

В рамках подготовки к чемпионату мира Кирилл придумал Ане задачу. Он написал N знаковых

32-битных чисел и попросил вычислить значение некоторого выражения S. Пусть a_1, \dots, a_N - все эти числа. Тогда выражение это

$$S = (a_1 \text{ xor } a_2 \text{ xor } \dots \text{ xor } a_n) \text{ xor } (b_1 \text{ xor } b_2 \text{ xor } \dots \text{ xor } b_{n-1}),$$

где

$$b_i = F(a_i, a_{i+1}) \text{ xor } F(a_i, a_{i+2}) \text{ xor } \dots \text{ xor } F(a_i, a_n).$$

В этой формуле под знаком xor понимается побитовое «исключающее или», а $F(a, b) = x - 1$, где x - максимальная степень двойки, на которую делится нацело $a-b$, если $a \neq b$, и $F(a, b) = -1$, если $a = b$. Все операции xor выполняются слева направо, если скобки не указывают иной порядок.

Аня, как большая специалистка в области циклов, быстро написала требуемую программу, однако программа работала слишком долго. Чтобы лучше разобраться в этом вопросе, она попросила вас написать программу, которая бы укладывалась в отведенное время. Помогите ей это сделать.

Входные данные

В первой строке входного файла INPUT.TXT содержится число N ($1 \leq N \leq 10^5$). В следующих N строках содержится N 32-битных знаковых целых чисел a_i по одному на строке.

Выходные данные

В выходной файл OUTPUT.TXT выведите значение выражения.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	3 1 2 3	1
2	2 1 1	-1
3	3 1 2 4	6

Задача №564 Забор - 2

(Время: 1 сек. Память: 16 Мб Сложность: 29%)

Времена Советского Союза уже давно стали историей. И к рассказам о событиях, происходивших тогда, только в России могут относиться не как к анекдотам. Попробуйте мысленно перенестись в те времена.

Однажды новому управлению строительного комбината выделили землю для дачных участков. Земли было выделено много, но тут возникла следующая проблема. На складах комбината хранилось некоторое количество секций для возведения заборов. Фрагменты имели достаточно большую длину, но их явно не хватало, чтобы огородить выделенную землю и поделить ее на

участки (а попытавшись приобрести новую часть забора подходящей длины, любой сотрудник управления явно привлек бы внимание соответствующих органов). К тому же фрагменты были выполнены таким образом, что собрать каждый из них можно было только в один прямой отрезок.

Поэтому было решено, что каждый из членов управления может выбрать для себя три фрагмента забора и с помощью них огородить себе участок. Директор комбината (который, конечно же, делает выбор первым) поручил вам помочь ему выбрать такие фрагменты забора, чтобы его участок имел максимально возможную площадь.

Входные данные

В первой строке входного файла INPUT.TXT записано натуральное число n , количество фрагментов на складе ($n \leq 100$). На второй строке содержатся n натуральных чисел, не превосходящих 1000 - длины фрагментов забора.

Выходные данные

В выходной файл OUTPUT.TXT выведите максимальное значение площади участка, который может соорудить директор и номера фрагментов, из которых может получиться максимальный участок в произвольном порядке. Ответ следует выводить с точностью, не меньшей, чем 6 знаков после запятой. В случае, если создать участок ненулевой площади нельзя, выведите -1.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	3 1 1 1	0.4330127018922193 1 2 3
2	4 1 2 4 7	-1

Задача №565 Сушка

(Время: 1 сек. Память: 16 Мб Сложность: 59%)

Тетя Люба только что постирала все белье и теперь перед ней стоит непростая задача - как его высушить, чтобы ни одна вещь не успела испортиться. Сразу после стирки, i -я постиранная вещь имеет влажность w_i . Если она сушится на веревке, то за минуту ее влажность уменьшается на 1, а если на батарее - то на r (если влажность была меньше r , то она становится равной 0). Причем веревок у тети Любы много (хватает для одновременной сушки всех вещей), а батарея только одна, причем такая маленькая, что на ней нельзя сушить две вещи одновременно. i -я вещь испортится, если не высохнет за время d_i . Помогите тете Любе составить план, когда какую вещь повесить на батарею.

Входные данные

Первая строка входного файла INPUT.TXT содержит целые числа n ($1 \leq n \leq 10^5$) - количество мокрых вещей, и r ($1 \leq r \leq 10^9$). Следующие n строк содержат описания постиранных вещей – пары чисел w_i и d_i ($1 \leq w_i, d_i \leq 10^9$).

Выходные данные

В выходной файл OUTPUT.TXT выведите план сушки в виде пар целых чисел t_i и k_i , где t_i -

время в минутах от начала сушки, а k_i - номер вещи, которую нужно повесить на батарею в этот момент. Выводите пары в порядке увеличения t_i . Пар не должно быть больше 10^5 . Не выводите числа больше 10^9 . Если высушить все вещи невозможно, выведите слово «Impossible».

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	3 3 2000 1000 2000 2000 2500 1500	0 3 500 1 1000 3
2	3 3 2000 1000 2000 1000 2000 1000	Impossible

Задача №566 Халява

(Время: 1 сек. Память: 16 Мб Сложность: 61%)

Гриша очень любит газировку PupsiCola. Однажды он узнал, что, собрав несколько крышек со звездочками, можно получить футболку. Гриша нашел А крышек с одной звездочкой, В крышек с двумя звездочками и С крышек с тремя звездочками. На футболку можно обменять набор крышек, общее количество звездочек на которых не меньше К.

Помогите Грише узнать, сколько футболок он может получить.

Входные данные

Входной файл INPUT.TXT содержит целые числа А, В, С и К ($0 \leq A, B, C \leq 100, 1 \leq K \leq 1000$).

Выходные данные

В выходной файл OUTPUT.TXT выведите максимальное количество футболок, которые может получить Гриша.

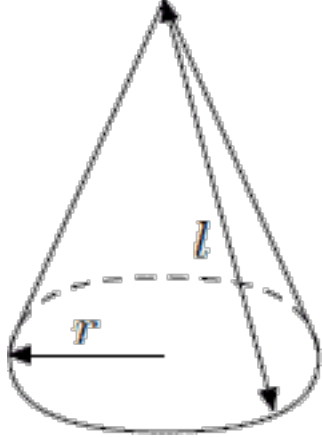
Примеры

№	INPUT.TXT	OUTPUT.TXT
1	2 2 2 4	3
2	0 0 4 4	2

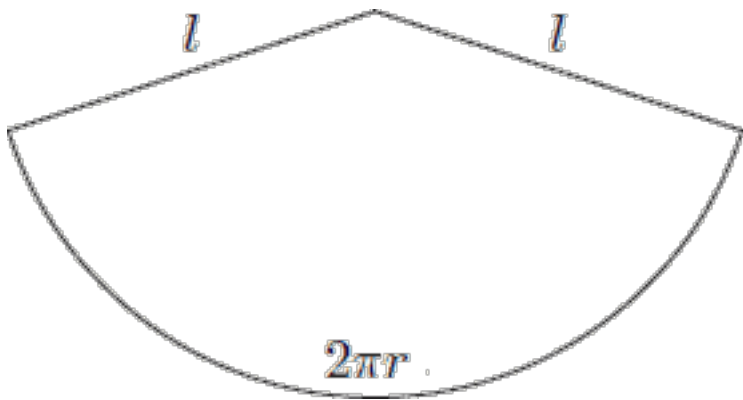
Задача №567 Упаковка подарка

(Время: 1 сек. Память: 16 Мб Сложность: 83%)

Для упаковки подарка Роме нужно сделать из фольги конус с нижним основанием радиуса r и длиной образующей l .



Для этого ему нужен кусок фольги такой формы:



У Ромы есть рулон фольги шириной d , от которого он может отрезать кусок произвольной длины. Помогите ему узнать минимальную длину куска, из которого можно вырезать нужную фигуру.

Входные данные

Во входном файле INPUT.TXT содержатся три целых числа: r , l и d (все числа не превосходят 10^6 , $l > r$). Гарантируется, что существует способ вырезать нужную фигуру.

Выходные данные

В выходной файл OUTPUT.TXT выведите минимально возможную длину куска фольги, из которого Рома сможет сделать нужную упаковку. Выведите не менее 6 знаков после точки.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	1 2 2	4

Задача №568
Дороги - 2

(Время: 1 сек. Память: 16 Мб Сложность: 53%)

Мэр города Гадюкино решил проверить состояние дорог после только что проведенного капитального ремонта. Для этого он хочет проехать по каждой дороге в обоих направлениях. Помогите мэру составить кратчайший маршрут, проходящий по каждой дороге в каждом направлении хотя бы один раз.

В городе Гадюкино n перекрестков и m дорог, каждая из которых соединяет два различных перекрестка. Между двумя перекрестками может быть не более одной дороги. Известно, что по дорогам от каждого перекрестка можно доехать до любого другого.

Входные данные

Входной файл INPUT.TXT содержит целые числа n и m ($1 \leq n \leq 10^4$, $1 \leq m \leq 10^5$), и далее m пар целых чисел a_i и b_i - номера перекрестков, которые соединяет i -я дорога.

Выходные данные

В выходной файл OUTPUT.TXT выведите число s - минимальную длину пути и далее $s+1$ число - номера перекрестков в том порядке, в котором их нужно проезжать.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	3 3 1 2 2 3 1 3	6 1 2 3 1 3 2 1

Задача №569 Зарплата - 2

(Время: 1 сек. Память: 16 Мб Сложность: 43%)

В одном государственном учреждении работают n сотрудников. Их фактические зарплаты равны d_i , но по закону всем сотрудникам положено платить одинаково, поэтому им всем официально выплачивают среднюю зарплату d , такую, чтобы общая сумма оставалась такой же, а потом сотрудники сами перераспределяют полученные деньги.

Короче, творится полный бардак. И чтобы этот бардак уменьшить, Самый Главный Начальник решил использовать недавно принятый закон «о материальной помощи», который позволяет любому сотруднику часть своей зарплаты (целое число рублей от 1 до $d-1$) передавать другому сотруднику в виде материальной поддержки. Однако по закону сотрудник, таким образом, может «помогать» только одному «малоимущему».

Например, если Петя и Вася получают по 100 рублей и Петя напишет заявление на передачу 30% своей зарплаты Васе, то Петя будет получать 70 рублей, а Вася - 130.

Теперь Самый Главный Начальник хочет узнать: кто, кому и сколько должен передавать, чтобы в результате все получали ровно столько, сколько нужно. Помогите ему это сделать.

Входные данные

Входной файл INPUT.TXT содержит целые числа n и d ($1 \leq n \leq 10^5$, $1 \leq d \leq 10^9$), и далее n целых чисел d_i ($1 \leq d_i \leq 10^9$). Сумма всех d_i равна $n \cdot d$.

Выходные данные

В выходной файл OUTPUT.TXT выведите n пар целых чисел a_i и b_i , означающих, что сотрудник i передает сотруднику a_i часть своей зарплаты в размере b_i рублей. Если сотрудник i ничего никому не передает, выведите вместо a_i и b_i два нуля.

Примеры

--	--	--

№	INPUT.TXT	OUTPUT.TXT
1	3 200 100 300 200	2 100 0 0 0 0
2	3 200 10 300 290	3 190 0 0 2 100

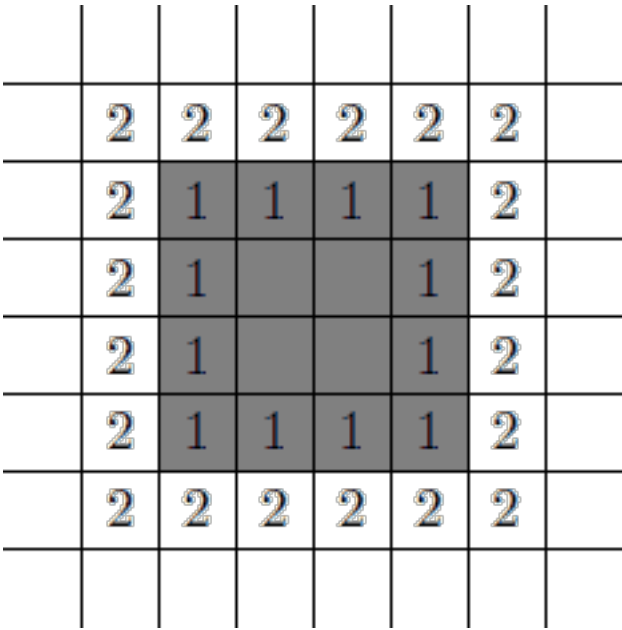
Задача №570
Квадрат

(Время: 1 сек. Память: 16 Мб Сложность: 33%)

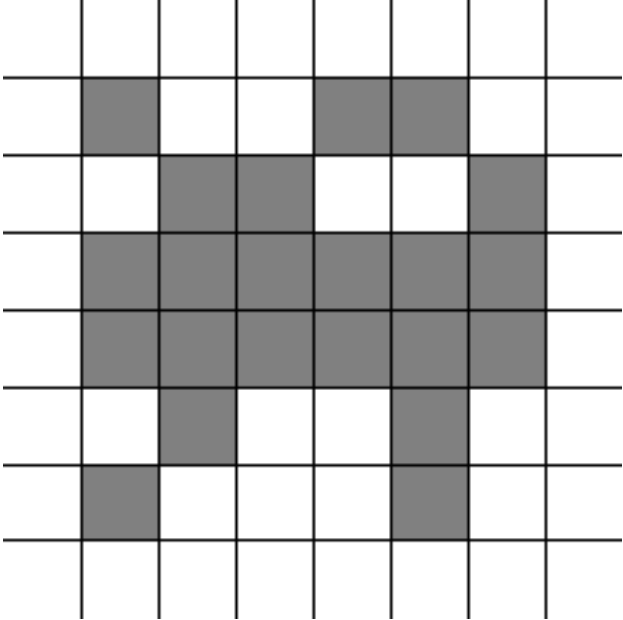
На сайте сотового оператора ВеерLine сделали защиту от роботов, рассылающих SMS-сообщения: прежде, чем отправить SMS, пользователь должен написать, какую фигуру он видит в специальном окошке: квадрат или круг. Причем, для усиления защиты, в рисунок внесены небольшие помехи.

Коле срочно нужно разослать всем друзьям сообщение, поэтому он просит Вас написать программу, распознающую изображение.

Экспериментально установлено, что система рисует квадрат с помехами следующим образом: сначала на белом фоне рисуется черный квадрат $k \times k$ клеток ($k \geq 3$), затем некоторые клетки на границе квадрата (на рисунке обозначены цифрой 1) закрашиваются белым, а некоторые клетки, граничащие с квадратом (на рисунке обозначены цифрой 2), закрашиваются черным.



Например, квадрат 4 x 4 после нанесения помех может выглядеть так:



Входные данные

Первая строка входного файла INPUT.TXT содержит целые числа n и m - размеры экрана ($1 \leq n, m \leq 1000$). Следующие n строк, по m символов в каждой, содержат описание картинки. Черные клетки обозначены символом «*», а белые - символом «.».

Выходные данные

В выходной файл OUTPUT.TXT выведите «SQUARE», если заданная картинка может быть квадратом после преобразований, описанных в условии, иначе выведите слово «CIRCLE».

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	10 10***..... ..***..... ..***.....	SQUARE
2	10 10*..... ..*****.. ..*****.. ..*****.. ..*****.. ..*.*.*.. ...*.....	SQUARE
	10 10	CIRCLE

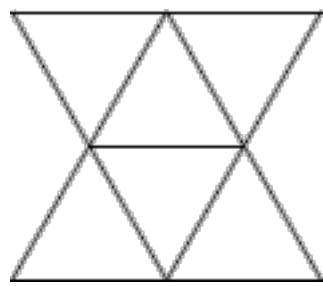
3	<pre>..****..... .*****... *****.**. *****. *****. .*****... ..****... ...*.....</pre>	
4	<pre>3 3</pre>	CIRCLE

Задача №571
Треугольники - 4

(Время: 5 сек. Память: 64 Мб Сложность: 86%)

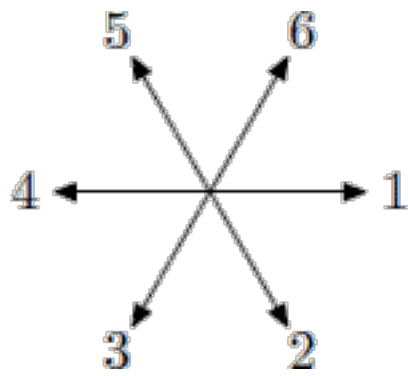
У Сени есть шоколадка, составленная из нескольких прилегающих друг к другу плиточек в форме правильных треугольников. Его брат Женя нашел эту шоколадку и решил сделать ее треугольной, съев все лишнее (ведь треугольные шоколадки намного вкуснее). Сколькими способами он может это сделать?

Например, из такой шоколадки:



можно сделать треугольную шоколадку со стороной 1 шестью способами или шоколадку со стороной 2 двумя способами. Итого восемь способов.

Входные данные



Форма шоколадки задается ее границей в порядке обхода по часовой стрелке. Первая строка входного файла INPUT.TXT содержит число n - количество отрезков на границе ($1 \leq n \leq 5000$). Далее n чисел от 1 до 6, задающих направление движения по границе (см. рисунок).

Выходные данные

В выходной файл OUTPUT.TXT выведите одно число - количество способов.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	8 1 1 3 2 4 4 6 5	8

Задача №572

Технология программирования

(Время: 1 сек. Память: 16 Мб Сложность: 60%)

Толик придумал новую технологию программирования. Он хочет уговорить друзей использовать ее. Однако все не так просто. i -й друг согласится использовать технологию Толика, если его авторитет будет не меньше a_i (авторитет выражается целым числом). Как только он начнет ее использовать, к авторитету Толика прибавится число b_i (попадаются люди, у которых $b_i < 0$). Помогите Толику наставить на путь истинный как можно больше своих друзей.

Входные данные

На первой строке входного файла INPUT.TXT содержатся два числа: n ($1 \leq n \leq 1000$) – количество друзей у Толика, и первоначальный авторитет Толика . Следующие n строк содержат пары чисел a_i и b_i . Все числа целые, по модулю не больше 10^6 .

Выходные данные

В выходной файл OUTPUT.TXT выведите число m - максимальное число друзей, которых может увлечь Толик, и затем m чисел - номера друзей в том порядке, в котором их нужно агитировать.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	5 1 1 3 6 -5 6 -4 2 2 2 -1	4 1 4 3 5

Задача №573

НЛО

(Время: 1 сек. Память: 16 Мб Сложность: 58%)

В маленьком городке М начала действовать служба контроля за незаконными полетами НЛО. Первая задача службы - выяснить, сколько НЛО действует в окрестности города.

Агенты службы опросили множество свидетелей и составили список случаев встречи с НЛО, произошедших за одни сутки, с указанием места и времени наблюдения.

Теперь аналитики хотят понять, сколько же на самом деле было НЛО. Из данных разведки известна максимальная скорость, с которой может лететь НЛО. Аналитики просят вас узнать, какое минимальное количество НЛО могли наблюдать свидетели.

Входные данные

На первой строке входного файла INPUT.TXT содержатся целые числа n и v - количество случаев наблюдения и максимальная скорость НЛО ($1 \leq n \leq 100, 1 \leq v \leq 10000$). Следующие n строк содержат описания случаев встречи с НЛО в формате «ЧЧ:ММ х у», где ЧЧ:ММ – время встречи, x и y - координаты места, в котором наблюдался НЛО (для простоты будем считать, что все встречи происходили на плоскости). Координаты по модулю не превышают 1000. Скорость выражена в км/ч, координаты - в км.

Выходные данные

В выходной файл OUTPUT.TXT выведите одно число - минимальное возможное количество НЛО.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	4 1 12:00 0 0 13:10 0 1 14:00 1 0 15:00 1 1	2

Задача №574 Анаграммы

(Время: 1 сек. Память: 16 Мб Сложность: 26%)

Строка $S1$ называется анаграммой строки $S2$, если она получается из $S2$ перестановкой символов. Даны строки $S1$ и $S2$. Напишите программу, которая проверяет, является ли $S1$ анаграммой $S2$.

Входные данные

Первая строка входного файла INPUT.TXT содержит строку $S1$, вторая - $S2$. Обе строки состоят только из прописных букв латинского алфавита. Строки не пусты и имеют длину не больше 100000 символов.

Выходные данные

В выходной файл OUTPUT.TXT выведите YES, если $S1$ является анаграммой $S2$, и NO - в противном случае.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	ABAA ABBA	NO
2	ABBA BABA	YES

Задача №575 Строительство

(Время: 1 сек. Память: 16 Мб Сложность: 41%)

Два числа называются *взаимно простыми*, если их наибольший общий делитель равен 1. *Наибольшим общим делителем* двух чисел а и b называется наибольшее целое положительное число, на которое делятся и а, и b.

Входные данные

Входной файл INPUT.TXT содержит целое положительное число N ($1 \leq N \leq 10^5$).

Выходные данные

В выходной файл OUTPUT.TXT выведите одно число – результат гадания.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	13	12
2	6	2

Задача №577 Таблица умножения

(Время: 1 сек. Память: 16 Мб Сложность: 30%)

Большой любитель математики Вова решил повесить у себя в комнате таблицу умножения. После некоторых раздумий он обнаружил, что обычная таблица умножения 10 на 10 уже не популярна в наши дни. Он решил повесить у себя в комнате таблицу N на M. Представив себе эту таблицу, Вова задался вопросом - сколько раз в ней встречается каждая из цифр от 0 до 9?

И прежде чем нарисовать эту таблицу Вова попросил вас написать программу, которая даст ответ на его вопрос.

Как известно, в таблице умножения на пересечении строки i и столбца j записано число $i \cdot j$.

Входные данные

Входной файл INPUT.TXT состоит из единственной строки, на которой через пробел записаны два натуральных числа N и M ($1 \leq N, M \leq 1000$).

Выходные данные

Выходной файл OUTPUT.TXT должен состоять из десяти строк. На строке i выведите количество раз, которое Вове придется нарисовать цифру i - 1.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	10 10	28 24 27 15 23 15 17 8

		15
		6

Задача №578

Система счисления

(Время: 1 сек. Память: 16 Мб Сложность: 37%)

Вчера на уроке математики Саша узнал о том, что иногда полезно использовать вместо десятичной системы счисления какую-нибудь другую.

Однако, учительница не объяснила, почему в системе счисления по основанию b в качестве цифр выбирают числа от 0 до $b - 1$.

Немного подумав, Саша понял, что можно выбирать и другие наборы цифр. Например, вместо троичной системы счисления можно рассмотреть систему счисления, где вместо обычных цифр 0, 1, 2 есть цифры 1, 2 и 3.

Саша заинтересовался вопросом, а как перевести число n в эту систему счисления? Например, число 7 в этой системе записывается как 21, так как $7 = 2 \cdot 3 + 1$, а число 22 записывается как 211, так как $22 = 2 \cdot 9 + 1 \cdot 3 + 1$.

Входные данные

Входной файл INPUT.TXT содержит натуральное число n , $1 \leq n \leq 2 \cdot 10^9$.

Выходные данные

В выходной файл OUTPUT.TXT выведите число n записанное в указанной системе счисления.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	7	21
2	22	211

Задача №579

Модуль суммы

(Время: 1 сек. Память: 16 Мб Сложность: 25%)

Дана последовательность целых чисел. Требуется найти подпоследовательность заданной последовательности с максимальным модулем суммы входящих в нее чисел. Напомним, что модуль целого числа x равняется x , если $x \geq 0$ и $-x$, если $x < 0$.

Входные данные

Первая строка входного файла INPUT.TXT содержит натуральное число n ($1 \leq n \leq 10000$) - длину последовательности. Во второй строке записаны n целых чисел, по модулю не превосходящих 10000.

Выходные данные

В первой строке выходного файла OUTPUT.TXT выведите длину l выбранной вами подпоследовательности. Во второй строке должны быть записаны l различных чисел, разделенных

пробелами - номера выбранных членов последовательности.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	5 -1 4 -1 6 -7	2 2 4

Задача №580 Поднос

(Время: 1 сек. Память: 16 Мб Сложность: 34%)

Школьная столовая хочет заказать новые треугольные подносы. Естественно, был организован тендер. Основным критерием отбора предложений является условие того, что имеющиеся в столовой круглые тарелки должны помещаться на новые подносы.

Директор школы дал вам задание написать программу, проверяющую, помещается ли тарелка данного размера на поднос с данными длинами сторон.

Входные данные

В первой строке входного файла INPUT.TXT находятся 3 целых положительных числа - длины сторон подноса. Гарантируется, что поднос с такими сторонами существует. Во второй строке входного файла содержится радиус тарелки - целое положительное число. Все числа во входном файле не превосходят 100. Радиус тарелки и стороны подноса указаны в одних и тех же единицах измерения.

Выходные данные

Выведите в выходной файл OUTPUT.TXT сообщение YES, если тарелка помещается на поднос, и NO - в противном случае.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	1 1 1 1	NO
2	3 4 5 1	YES

Задача №581 Атака летающих тарелок

(Время: 1 сек. Память: 16 Мб Сложность: 36%)

Вы работаете в фирме, занимающейся разработкой компьютерных игр. Сейчас вы занимаетесь разработкой новой компьютерной игры "Атака летающих тарелок". По сюжету игры на планету Зумла приземляются летающие тарелки, и их надо уничтожить. Игрок управляет лазерной пушкой. Для того, чтобы произвести выстрел он указывает две точки на поверхности Зумлы (которая в игре считается плоской), через которые должен проходить лазерный луч (который является прямой).

Вы должны написать программу, определяющую, какие летающие тарелки были уничтожены выстрелом.

Входные данные

Первая строка входного файла INPUT.TXT содержит целое число n ($1 \leq n \leq 30000$) - число приземлившихся летающих тарелок. Вторая строка содержит числа x_1, y_1, x_2, y_2 - координаты точек, через которые проходит лазерный луч. Далее идут n строк, каждая из которых содержит описание одной летающей тарелки в формате $x_i y_i r_i$, где x_i, y_i - координаты центра, r_i - радиус тарелки. Все числа целые и не превосходят по модулю 10000. Радиусы летающих тарелок - целые и положительные. Летающие тарелки могут касаться и пересекаться.

Выходные данные

В первую строку выходного файла OUTPUT.TXT выведите количество уничтоженных летающих тарелок. Во вторую строку выведите номера уничтоженных летающих тарелок в возрастающем порядке. Тарелка считается уничтоженной, если она имеет, хотя бы одну общую точку с лазерным лучом.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	2 0 0 1 1 2 2 100 1000 1000 1	2 1 2

Задача №582 Кубик

(Время: 1 сек. Память: 16 Мб Сложность: 47%)

Все мы в детстве играли в кубики. Были у нас и кубики с цифрами, и кубики с буквами. Были также и разноцветные кубики. Маленький Андрюша тоже любит играть в кубики. У него есть несколько наборов кубиков, причем все кубики из одного набора раскрашены одинаково, а кубики из разных наборов - по-разному.

На столе у Андрюши лежат два кубика. Помогите ему определить, принадлежат они одному набору или нет. Стол у Андрюши стеклянный, поэтому он видит цвета всех граней кубика. Кубики принадлежат одному набору, если один из них можно комбинацией поворотов вокруг осей, проходящих через середины противоположных граней, перевести в другой.

Входные данные

Первая строка входного файла INPUT.TXT содержит описание первого кубика в формате: цвет передней грани, цвет задней грани, цвет верхней грани, цвет нижней грани, цвет левой грани, цвет правой грани. Во второй строке находится описание второго кубика в таком же формате. Цвета граней кодируются числами. Все числа во входном файле - целые, положительные и не превосходят 100.

Выходные данные

В выходной файл OUTPUT.TXT выведите «YES», если первый и второй кубики принадлежат одному набору, и «NO» - в противном случае.

Примеры

--	--	--

№	INPUT.TXT	OUTPUT.TXT
1	1 2 3 4 5 6 1 2 3 4 5 6	YES
2	1 2 3 4 5 6 1 1 1 1 1 1	NO
3	1 2 3 4 5 6 5 6 3 4 2 1	YES

Задача №583

Рисование

(Время: 1 сек. Память: 16 Мб Сложность: 65%)

При обучении школьников младших классов программированию часто используется язык «Лого», позволяющий рисовать на экране картинки хвостом виртуальной черепашки.

Рассмотрим упрощенную версию этого языка, в которой разрешается подавать черепашке следующие команды: переместиться вперед на некоторое количество сантиметров, рисуя за собой линию, повернуть налево на 90 градусов и повернуть направо на 90 градусов.

Рассмотрим замкнутую фигуру, нарисованную таким образом. Требуется проверить, верно ли, что любой вертикальный и любой горизонтальный отрезок, соединяющий две точки фигуры, полностью содержится внутри этой фигуры.

Входные данные

Первая строка входного файла INPUT.TXT содержит число n - количество команд черепашки, использованных для рисования фигуры ($7 \leq n \leq 40\,000$). Следующие n строк содержат команды. Команды задаются следующим образом:

- "f x" - переместиться вперед на x сантиметров (x - натуральное число);
- "l" - повернуть налево;
- "r" - повернуть направо.

Известно, что последовательность команд начинается с команды перемещения, не выполняется подряд две команды поворота или две команды перемещения, последняя команда также является командой перемещения. Гарантируется, что ломаную можно поместить в некоторый прямоугольник, длины сторон которого не превосходят 10^9 .

Выходные данные

В выходной файл OUTPUT.TXT выведите «TRUE», если нарисованная фигура удовлетворяет описанному выше свойству и «FALSE», если не удовлетворяет.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	11 f 1 r f 2 r f 2 r	TRUE

f 1	
r	
f 1	
l	
f 1	

Задача №584

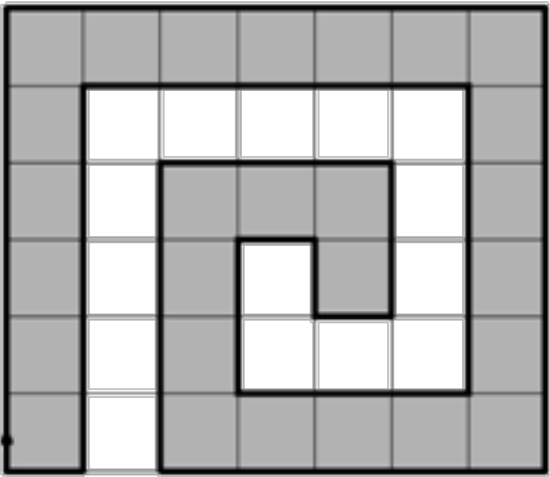
Спираль - 2

(Время: 1 сек. Память: 16 Мб Сложность: 59%)

При обучении школьников младших классов программированию часто используется язык «Лого», позволяющий рисовать на экране картинки хвостом виртуальной черепашки.

Рассмотрим упрощенную версию этого языка, в которой разрешается подавать черепашке следующие команды: переместиться вперед на некоторое количество сантиметров, рисуя за собой линию, повернуть налево на 90 градусов и повернуть направо на 90 градусов.

Вам предлагается написать на этом языке программу, изображающую контур спирали.



Спираль - это множество клеток, которое строится по следующим правилам. Рассмотрим прямоугольник, состоящий из m х n единичных клеток. Закрасим некоторые из его клеток в следующей последовательности.

Начнем красить левый столбец снизу вверх. Когда продвижение невозможно, то есть либо следующая клетка лежит вне прямоугольника, либо соприкасается с закрашенной ранее, направление движения поворачивается по часовой стрелке на 90 градусов. Если и после этого поворота дальнейшее движение невозможно, процесс заканчивается. Множество закрашенных таким образом клеток и образует спираль.

Входные данные

Входной файл INPUT.TXT содержит два целых числа m и n ($1 \leq m, n \leq 20\,000$).

Выходные данные

В выходной файл OUTPUT.TXT выведите последовательность команд, в результате выполнения которой будет нарисован контур ломаной. Напомним формат, в котором следует выводить результат. Первая строка должна содержать k - количество команд. Последующие k строк задают либо длину отрезка d , который надо провести, в формате $f\ d$, либо направление поворота - l для поворота налево и r для поворота направо. Команды рисования отрезка и поворота должны чередоваться, начинаясь и заканчиваясь командой рисования отрезка. Считайте, что начало первого отрезка совпадает с левым нижним углом прямоугольника. При этом контур должен быть выведен в направлении обхода по часовой стрелке, и никакая его часть ненулевой длины не должна быть нарисована дважды.

Пример

№	INPUT.TXT	OUTPUT.TXT
	6 7	31 f 6

1	r
	f 7
	r
	f 6
	r
	f 5
	r
	f 4
	r
	f 3
	r
	f 2
	r
	f 1
	r
	f 1
	l
	f 1
	l
	f 2
	l
	f 3
	l
	f 4
	l
	f 5
	l
	f 5
	r
	f 1

Задача №585

Египетские знаки

(Время: 1 сек. Память: 16 Мб Сложность: 72%)

Прошлым летом школьник Вася побывал в Египте. И лишь сейчас он вспомнил, что во время своей поездки он успел сфотографировать много интересностей. Перебирая эти фотографии, больше всего он заинтересовался видами пирамид в Гизе. Пирамиды, помимо всяких иероглифов, содержали удивительные знаки - 1, 2, 3, 4, 5, 6, 7, 8, 9, 0. Внимательно присмотревшись Вася заметил, что каждая строка таких странных символов является степенью двойки и более того первая строка начинается последовательностью символов 1, вторая - 2, ..., сто двадцатая - 120, и т.д. Но все было бы хорошо, если бы Вася пользовался современными фотоаппаратами, поэтому его старенькая мыльница плохо сфотографировала наиболее освещенные части пирамиды, поэтому все символы разобрать невозможно и определить показатель степени двойки, соответствующий таким строкам, затруднительно.

Вася задумался о восстановлении символов и, наконец, решил попросить кого-нибудь написать программу (Вася учится в 6 классе и не знает языков программирования), которая бы определила показатель степени двойки, которая записана в N-ой строке.

Входные данные

Входной файл INPUT.TXT содержит единственное число $N \leq 10^7$.

Выходные данные

В выходной файл OUTPUT.TXT выведите минимальное натуральное число K такое, что 2^K в десятичной записи начинается с числа N , или если Вася что-то напутал и такого числа нет - выведите -1.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	12	7
2	134	27
3	82	209

Пояснение

Пример №1: $2^7=128$

Пример №2: $2^{27}=134217728$

Задача №586 Законопроект

(Время: 1 сек. Память: 16 Мб Сложность: 62%)

В связи с изменением политической обстановки в мире и во избежание наступления экономического кризиса в стране, президент Флатландии на очередном заседании кабинета министров поставил перед правительством первоочередную задачу о подготовке пакета реформ. Сразу же после заседания был созван специальный комитет, который подготовил k различных законопроектов, каждый из которых описывал один из возможных путей проведения реформ. Перед парламентом Флатландии (который состоит из n депутатов) встала нелегкая задача выбора одного из этих законопроектов.

Согласно конституции Флатландии, все обсуждения законопроектов в парламенте проходят в одном чтении. После того, как депутаты ознакомятся с проектом закона, проводится голосование. Каждый из депутатов может проголосовать либо за принятие закона, либо против. При этом, закон принимается, если за него проголосовало как минимум l депутатов.

Известно, что законопроекты, подготовленные специальным комитетом, будут обсуждаться на заседаниях парламента последовательно, начиная с проекта номер 1 и заканчивая проектом номер k . При этом, если в результате обсуждения законопроекта номер i этот законопроект будет принят, то оставшиеся законопроекты (с номерами от $i + 1$ до k) не будут обсуждаться вообще. Если же законопроект с номером i будет отклонен, то парламент перейдет к обсуждению следующего законопроекта (с номером $i + 1$). Если в результате последовательного голосования парламент отвергнет все проекты, то процесс проведения реформ отложится на неопределенное время (в этом случае страну, скорее всего, ожидает глубокий кризис).

Известно также, что каждый депутат Флатландии кратко ознакомился с каждым законопроектом еще до первого заседания парламента и составил для себя список всех проектов в порядке своих предпочтений. Обозначим номер j -го законопроекта в списке предпочтений i -го депутата, как $a_{i,j}$.

Тогда это будет означать, что i -й депутат во время голосований будет пытаться добиться принятия проекта с номером $a_{i,1}$, а если это не возможно, то проекта с номером $a_{i,2}$, если и этого никак не добиться, то i -й депутат будет пытаться протолкнуть закон номер $a_{i,3}$ и так далее. При

этом никакой депутат не хочет допустить кризиса в стране, и, поэтому, в первую очередь будет пытаться предотвратить тот случай, когда все законопроекты будут отвергнуты парламентом. Все депутаты флатландского парламента знают не только свои предпочтения, но и предпочтения всех своих коллег. К тому же они достаточно умны и будут действовать оптимально (то есть будут пытаться добиться принятия закона, как можно более предпочтительного с их точки зрения).

Президент Флатландии попросил Вас, как своего главного советника, предсказать результаты работы парламента (а именно то, какой из вариантов закона о реформах будет в итоге принят).

Входные данные

В первой строке входного файла INPUT.TXT записаны три целых числа: n ($1 \leq n \leq 1000$), l ($1 \leq l \leq n$) и k ($1 \leq k \leq 1000$). Далее следует n строк, каждая из которых содержит k чисел (j -е число в i -й строке равно $a_{i,j}$).

Выходные данные

В выходной файл OUTPUT.TXT выведите номер законопроекта, который будет принят парламентом, или слово "Crisis" (без кавычек), если все они будут отклонены.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	3 2 4 3 4 1 2 2 3 4 1 4 1 2 3	1
2	3 2 4 3 1 4 2 1 3 4 2 4 2 3 1	3

Задача №587 Экспедиция на Морс

(Время: 1 сек. Память: 16 Мб Сложность: 60%)

Для исследования плоского космоса была организована сверхсекретная научная экспедиция. Специально для нее был сконструирован космический корабль Z-239, а в качестве первой планеты для исследования была выбрана Зумла.

На Z-239 установлено n двигателей на антиматерии, каждый из которых характеризуется вектором силы тяги, которую он создает. Для перемещения корабля необходимо включить некоторые двигатели, при этом общая сила тяги будет равна векторной сумме тяг включенных двигателей.

При старте корабля его можно ориентировать произвольным образом, после чего включить некоторые (возможно, все) двигатели.

Очередной целью экспедиции была выбрана планета Морс. Разумеется, руководитель экспедиции хочет добраться туда как можно быстрее. Чем больше модуль суммарной силы тяги, тем быстрее полетит корабль. Какие двигатели нужно включить, чтобы добраться на Морс как можно быстрее?

Входные данные

Первая строка входного файла INPUT.TXT содержит натуральное число n - количество двигателей ($1 \leq n \leq 30\,000$). В $i+1$ -ой строке записана пара чисел - модуль силы тяги i -ого двигателя, и угол в градусах между вектором его силы тяги и направлением на Полярную звезду перед поворотом корабля (вещественное число от 0 до 360 градусов). Модуль силы тяги каждого двигателя не превосходит 10 000.

Выходные данные

В первой строке выходного файла OUTPUT.TXT выведите количество l двигателей, которые надо включить. Во строке должны быть записаны l различных чисел, разделенных пробелами - номера включаемых двигателей в любом порядке.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	3 5.00 0.00 3.00 90.00 4.00 270.00	2 1 3
2	2 10.63 228.81 10.63 48.81	1 2

Задача №588 Число

(Время: 1 сек. Память: 16 Мб Сложность: 39%)

Скажем, что число в b -ой системе счисления описывает себя, если оно - b -значное, его первая цифра равна числу единиц в нем, вторая цифра равна числу двоек, и т.д., $b-1$ -ая цифра равна числу цифр $b-1$, а последняя цифра равна числу нулей. Примером такого числа для $b = 5$ является 12002.

Задано число b . Требуется найти число в системе счисления по основанию b , которое описывает себя.

Входные данные

Входной файл INPUT.TXT содержит единственное целое число b ($2 \leq b \leq 1000$).

Выходные данные

В выходной файл OUTPUT.TXT выведите любое описывающее себя число в системе счисления с основанием b , или -1, если такого числа не существует. Каждую цифру числа выводите на отдельной строке в десятичной записи.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	5	1 2 0 0 2
2	2	-1

Задача №589
Физика

(Время: 1 сек. Память: 16 Мб Сложность: 84%)

В трехмерном пространстве с прямоугольной декартовой системой координат находятся n абсолютно упругих шаров. Для каждого из них известны: масса m_i , радиус r_i , координаты x_i, y_i, z_i и вектор скорости $v_i = (v_{ix}, v_{iy}, v_{iz})$ в начальный момент времени. Необходимо рассчитать их координаты и скорости по прошествии T секунд от начального момента времени. Шары взаимодействуют только при соударениях, других взаимодействий между ними нет.

Учтите, что шары могут сталкиваться, и их столкновение описывается законами сохранения энергии и импульса, то есть сохраняется величина $\sum_{i=1}^n m_i \cdot (v_{ix}^2 + v_{iy}^2 + v_{iz}^2)$, а также вектор $\sum_{i=1}^n m_i \cdot v_i$.

Заметим, что при столкновении шаров их скорости могут измениться только на вектор, параллельный прямой, соединяющей их центры в момент столкновения.

Входные данные

Первая строка входного файла INPUT.TXT содержит натуральное число n ($1 \leq n \leq 50$) - количество шаров. Следующие n строк входного файла содержат описание начального состояния шаров, $(i+1)$ -ая строка содержит 8 разделенных пробелами вещественных чисел, не более чем с тремя знаками после запятой: $m_i, r_i, x_i, y_i, z_i, v_{ix}, v_{iy}, v_{iz}$.

Последняя строка содержит целое число T ($0 \leq T \leq 100$) - время, состояние системы по прошествии которого надо рассчитать. Все проекции скоростей заданы в метрах в секунду, все радиусы и координаты центров - в метрах, все массы - в килограммах, а время T задано в секундах. Гарантируется, что входные данные таковы, что в каждом столкновении участвуют ровно 2 шара. Все числа во входном файле не превосходят 100 по абсолютной величине. Начальное положение шаров таково, что они не касаются друг друга и не пересекаются. Массы и радиусы всех шаров строго положительны.

Выходные данные

В выходной файл OUTPUT.TXT выведите n строк. На i -ой строке выведите 6 вещественных чисел с точностью не меньше чем 3 знака после десятичной точки: x -координату центра i -ого шара, y -координату центра i -ого шара, z -координату центра i -ого шара, проекцию его скорости на ось Ox , проекцию его скорости на ось Oy , проекцию его скорости на ось Oz по прошествии T секунд. Координаты выводите в метрах, проекции скоростей - в метрах в секунду.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	3 1.0 1.0 0.0 0.0 0.0 1.00 1.0 1.0 2.0 2.0 10.0 10.0 10.0 -1.0 -1.0 -1.0 3.0 3.0 20.0 20.0 20.0 -0.5 -0.5 -0.5 100	-155.643 -155.643 -155.643 -1.667 -1.667 -1.667 -47.028 -47.028 -47.028 -0.667 -0.667 -0.667 26.566 26.566 26.566 0.167 0.167 0.167
	2	-1.000 0.000 0.000 -1.000 0.000 0.000

2	1.0	1.0	0	0	0	0	0	0	2.000	0.000	0.000	0.000	0.000	0.000
	1.0	1.0	10	0	0	-1	0	0						
	9													

Задача №590

Зазеркалье

(Время: 1 сек. Память: 16 Мб Сложность: 42%)

Дано множество из N точек в декартовой системе координат. Требуется определить, является ли это множество центрально-симметричным.

Входные данные

В первой строке входного файла INPUT.TXT записано натуральное число N, не превышающее 1000. В следующих N строках записаны по три целых числа x, y, z, обозначающие координаты точки. Все точки попарно различны. Все координаты не превышают по модулю 2*10⁹.

Выходные данные

В единственную строку выходного файла OUTPUT.TXT нужно вывести Yes, если множество является центрально-симметричным и No в противном случае.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	2 0 0 0 5 7 3	Yes
2	3 0 0 5 5 0 0 0 5 0	No

Задача №591

Транспортная реформа

(Время: 1 сек. Память: 16 Мб Сложность: 70%)

Далекое королевство состоит из n городов, соединенных m дорогами, при этом из каждого города можно доехать в каждый по дорогам (по каждой из дорог можно ездить в обоих направлениях). Некоторые дороги находятся в хорошем состоянии, другие же требуют ремонта.

Содержание каждой из дорог обходится казне в 239239 дублонов в год. В связи с новой транспортной реформой, направленной на уменьшение ненужных расходов, было решено оставить на государственном содержании как можно меньше дорог, но по-прежнему должна остаться возможность из каждого города добраться в любой другой по государственным дорогам. Заметим, что при этом на государственном содержании останется ровно n - 1 дорога.

При этом, возможно, в число оставленных дорог войдут и требующие ремонта. Сэкономленные в связи с этой реформой деньги планируется направить на ремонт дорог, оставленных на государственном содержании.

В королевстве есть две конкурирующие фирмы, занимающиеся починкой дорог. Поскольку король не желает быть обвиненным в поддержке одной из этих фирм, он хочет иметь возможность

распределить дороги, оставленные на государственном содержании и требующие ремонта, поровну между ними. В связи с этим требуется оставить на государственном содержании четное число дорог, требующих ремонта.

Требуется составить соответствующий желанию короля список дорог, которые следует оставить на государственном содержании.

Входные данные

Первая строка входного файла INPUT.TXT содержит натуральные числа n ($2 \leq n \leq 30\,000$) и m ($1 \leq m \leq 100\,000$) - число городов и дорог в королевстве соответственно. Далее идут m строк - описание дорог. Дорога описывается тремя числами. Первые два из них - номера городов, которые она соединяет, третье равно 1, если дорога не требует ремонта, и 2, если требует. Города пронумерованы целыми числами от 1 до n . Ни одна дорога не соединяет город с самим собой, между любыми двумя городами не более одной дороги.

Выходные данные

В выходной файл OUTPUT.TXT выведите $n-1$ строку - номера государственных дорог (дороги нумеруются с 1 в том порядке, в котором они идут во входном файле). Если оставить на государственном содержании дороги указанным образом невозможно, то следует вывести -1.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	4 4	2
	1 2 1	3
	2 3 2	4
	3 4 1	
	4 1 2	

Задача №592 Небоскреб

(Время: 1 сек. Память: 16 Мб Сложность: 77%)

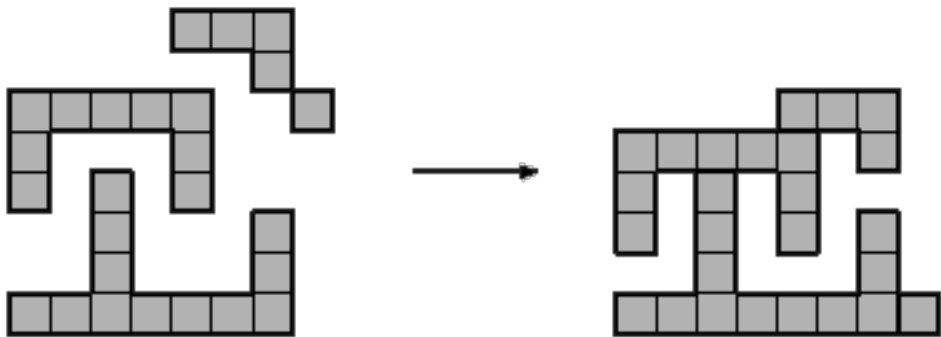
Спасатели разрабатывают новую систему действий при возникновении аварийных ситуациях на небоскребах. В этих случаях важно определять, какие части здания стоят устойчиво, а какие - нет.

На данный момент используется следующая двумерная модель. Строение описывается как прямоугольник, составленный из одинаковых блоков квадратной формы. Предполагается, что при аварии некоторые из блоков полностью разрушаются, а остальные остаются неповрежденными. Будем называть сегментом множество блоков, таких что из любого можно дойти до любого другого, если разрешается переходить из блока в блок, имеющий с ним общую сторону.

Считается, что сегмент стоит устойчиво, если один из его блоков соприкасается с нижней стороной прямоугольника.

Сегменты из блоков, которые стоят неустойчиво, проваливаются вниз до тех пор, пока какой-либо из блоков сегмента не будет соприкасаться с нижней стороной прямоугольника или с блоком устойчивого сегмента. После этого сегмент так же считается стоящим устойчиво.

По данным о том, какие блоки сохранились, требуется определить положение верхнего блока в каждом вертикальном столбце.



Входные данные

Первая строка входного файла INPUT.TXT содержит одно целое число m ($1 \leq m \leq 10\,000$). За ней следуют m строк, описывающих вертикальные столбцы. Описание производится в таком порядке: сначала записано число a_i оставшихся целыми фрагментов этого вертикального столбца, за которым следуют $2a_i$ чисел $l^{(1)}_i, r^{(1)}_i, l^{(2)}_i, r^{(2)}_i, \dots, l^{(a_i)}_i, r^{(a_i)}_i$, задающие нижние и верхние границы уцелевших фрагментов. При этом $1 \leq l^{(1)}_i \leq r^{(1)}_i, l^{(2)}_i \leq r^{(2)}_i, \dots, l^{(a_i)}_i \leq r^{(a_i)}_i \leq 10^6$, для всех допустимых i и j выполнено $r^{(j)}_i < l^{(j+1)}_i - 1$. Сумма всех a_i не превосходит 100 000.

Выходные данные

В выходной файл OUTPUT.TXT выведите m чисел - высоты, на которых находится самый верхний блок в соответствующем вертикальном столбце.

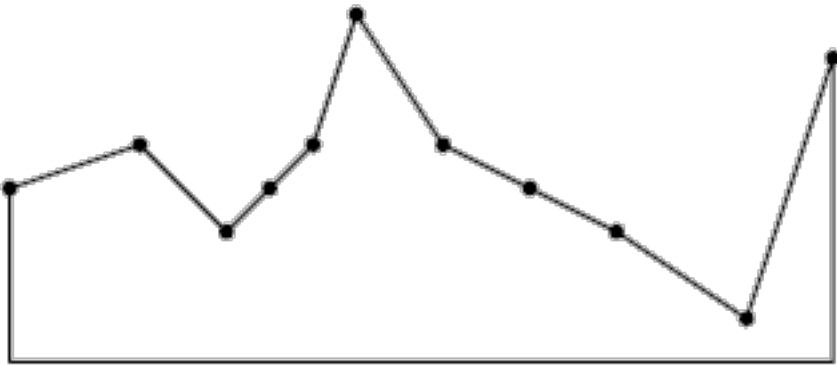
Пример

№	INPUT.TXT	OUTPUT.TXT
1	8 2 1 1 4 6 2 1 1 6 6 2 1 4 6 6 2 1 1 6 6 3 1 1 4 6 8 8 2 1 1 8 8 2 1 3 7 8 1 6 6	5 5 5 5 6 6 6 1

Задача №593 Башни - 2

(Время: 1 сек. Память: 16 Мб Сложность: 53%)

Для того, чтобы защититься от некоторых соседей, король решил построить стену, имеющую форму отрезка. С некоторыми соседями король находится в хороших отношениях, а некоторым готовится объявить войну. Король решил не загораживаться от друзей очень высокой стеной. Однако, стена, отделяющая его от врагов, должна быть достаточно высокой. Было решено, что для наблюдения за прилегающей территорией нужно построить башни. При этом, на участках между башнями высота стен должна изменяться равномерно.



После того, как стена и башни были построены, король заметил, что башни могут быть использованы для наблюдения за состоянием других башен. Однако, некоторые башни оказались очень высокими и загородили другие.

Для каждой башни король попросил вас выяснить, сколько других башен из нее видно.

Входные данные

В первой строке входного файла INPUT.TXT находится n ($2 \leq n \leq 2000$) - количество башен стены. В следующих n строках находятся натуральные числа x_i и h_i ($0 \leq x_i \leq 100000$, $1 \leq h_i \leq 10000$) - координата и высота i -ой башни. Все x_i различны.

Выходные данные

Выходной файл OUTPUT.TXT должен содержать n строк. В i -ой строке выведите количество башен, которые видно из башни номер i .

Пример

№	INPUT.TXT	OUTPUT.TXT
1	11	2
	0 4	5
	3 5	4
	5 3	4
	6 4	4
	7 5	10
	8 8	4
	10 5	4
	12 4	5
	14 3	3
	17 1	5
	19 7	

Задача №594 Треугольник - 2

(Время: 1 сек. Память: 16 Мб Сложность: 45%)

После окончания многолетней войны короли-победители решили разделить между собой захваченную территорию. Для того, чтобы избежать лишних споров, короли решили делить территорию следующим образом.

Каждому королю был предоставлен набор, состоящий из n отрезков l_1, \dots, l_n . Королям разрешается присоединять к их королевствам территорию, имеющую форму треугольника, составленного из имеющихся отрезков.

Помогите королю вашего королевства максимизировать площадь треугольника.

Входные данные

На первой строке входного файла INPUT.TXT задано число n ($3 \leq n \leq 100000$) - количество отрезков в наборе. В следующих n строках заданы длины отрезков в следующем формате.

Сначала указывается целое положительное число l - длина отрезка, а затем единица измерения (m, km, mile, uin, kairi, zhang, sen).

Напомним, что: 1mile = 1609m, 1km = 1000m, 1uin = 33m, 1kairi = 1852m, 1zhang = 3m и 1sen = 38m.

Длины всех отрезков не превосходят десяти километров. Гарантируется, что из заданных отрезков можно выбрать три отрезка так, что из них можно составить треугольник.

Выходные данные

В выходной файл OUTPUT.TXT выведите максимальную площадь (в таланг вахах) с точностью 10^{-6} и номера трех отрезков в любом порядке (если ответов несколько, выведите любой). Помните, что один таланг вах равен четырем квадратным метрам.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	7 1mile 1km 100m 20uin 1kairi 300zhang 40sen	291590.0233624191 7 1 5

Задача №595 Слова

(Время: 1 сек. Память: 16 Мб Сложность: 48%)

Для шифрования слов с ними можно производить множество различных операций. Например, интересна такая операция: первые несколько букв заданного слова приписываются к его концу в обратном порядке, после чего удаляются из начала слова. При этом слово $a_1a_2 \dots a_ka_{k+1} \dots a_n$ переходит в слово $a_{k+1} \dots a_na_ka_{k-1} \dots a_1$ (число k выбирается в диапазоне от 0 до n).

Для двух заданных слов требуется определить, можно ли применением описанной операции преобразовать первое слово во второе.

Входные данные

Входной файл INPUT.TXT состоит из двух строк - на первой из них записано исходное слово, а во второй - предполагаемый результат. Длины строк не превышают 50000 символов.

Выходные данные

В первой строке выходного файла OUTPUT.TXT выведите "Yes", если преобразование возможно, и "No", если нет. В случае положительного ответа во второй строке выведите k - длину перемещаемой части исходного слова k (из всех таких k выберите минимальный).

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	wpwdwpw wdwpwpw	Yes 2
	dwoddwd	No

2	dOdWdWd	
---	---------	--

Задача №596
Сотовая связь в большом городе

(Время: 1 сек. Память: 16 Мб Сложность: 35%)

В одном из больших городов нашей страны хорошо развит рынок сотовой связи. На рынке присутствуют несколько операторов, каждому из которых принадлежит некоторое количество базовых станций, с помощью которых организуется связь.

Каждая базовая станция характеризуется своими координатами (для простоты размерами базовых станций пренебрегаем) и радиусом надежной связи (если абонент находится на расстоянии, не превосходящем этот радиус, от базовой станции, то она может использоваться для работы с ним - передачи ему сигналов и приема сигналов от него).

Известный производитель сотовых телефонов Мокеа планирует оснастить свою новую модель сотового телефона функцией определения базовых станций, с которыми может работать абонент. Вам же предстоит написать программу, которая по местоположению абонента для каждого оператора определит, сколько базовых станций этого оператора могут работать с абонентом.

Входные данные

Первая строка входного файла INPUT.TXT содержит целое число n ($1 \leq n \leq 10000$) - количество базовых станций в городе. Далее идут описания этих базовых станций. Каждое описание занимает две строки. На первой расположено название оператора сотовой связи, которому принадлежит эта базовая станция, а на второй - три целых числа x, y, r ($-10000 \leq x, y \leq 10000, 1 \leq r \leq 10000$) - соответственно ее координаты и радиус надежной связи. Последняя строка входного файла содержит два целых числа x_a, y_a ($-10000 \leq x_a, y_a \leq 10000$) - координаты абонента.

Все координаты во входном файле даны в одной и той же декартовой прямоугольной системе координат. Названия операторов - это непустые строки длиной не более 50 символов, состоящие из цифр, строчных и прописных букв латинского алфавита. Прописные и строчные буквы латинского алфавита различаются (например, MPS и mps - два разных оператора).

Выходные данные

На первой строке выходного файла OUTPUT.TXT выведите число k - количество операторов сотовой связи, работающих в городе (разумеется, два оператора считаются разными, если их названия не совпадают). Далее выведите k строк. Каждая из этих строк должна содержать название оператора и количество базовых станций этого оператора, доступных абоненту. Первым должно идти название оператора, число базовых станций должно быть отделено от него одним пробелом. В этом списке операторы должны быть перечислены в том же порядке, в каком они встречаются во входном файле (см. примеры). Гарантируется, что $k \leq 100$.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	5 Megahorn 0 0 10 BeepLine 10 10 10 MPS 0 0 10	5 Megahorn 1 BeepLine 1 MPS 1 Ele2 0 SkyPink 0

	Ele2 0 0 1 SkyPink 100 100 10 5 5	
2	3 Megahorn 0 0 10 MPS 1 1 10 Megahorn 2 2 10 1 1	2 Megahorn 2 MPS 1

Задача №597

Внеземные гости

(Время: 1 сек. Память: 16 Мб Сложность: 10%)

Недавно на поле фермера Джона были обнаружены следы приземления летающих тарелок. Об этом даже писала газета Mew Yorc Courier.

Поле фермера Джона имеет форму круга радиусом r_1 . По сообщениям журналистов были обнаружены два следа от летающих тарелок, имевшие форму кругов. Один из них имел радиус r_2 , второй - радиус r_3 . Также сообщается, что они находились внутри поля фермера Джона и не пересекались (при этом, они, возможно, касались друг друга и/или границы поля).

Поскольку журналисты часто склонны преувеличивать масштабы событий, необходимо написать программу, которая будет проверять, могли ли иметь место события, описанные в газете.

Входные данные

Входной файл INPUT.TXT содержит три целых положительных числа - r_1, r_2, r_3 ($1 \leq r_1, r_2, r_3 \leq 10^9$).

Выходные данные

В выходной файл OUTPUT.TXT выведите слово YES, если информация, опубликованная в газете, может соответствовать правде, и слово NO - иначе.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	10 10 10	NO
2	10 3 4	YES

Задача №598

Друзья - 2

(Время: 1 сек. Память: 16 Мб Сложность: 51%)

Несколько человек решили поехать отдохнуть на природе, подышать свежим воздухом и т.п. Как это часто бывает, некоторые из них дружат друг с другом, а некоторые - нет. Для того, чтобы не испортить никому настроение, они решили разделиться на несколько групп. При этом, в каждой

группе должно быть не более 5 человек и они должны дружить друг с другом.

Найдите такое разбиение людей на группы, в котором размер наибольшей группы был бы максимальным (среди всех разбиений).

Входные данные

Первая строка входного файла INPUT.TXT содержит целое число n ($1 \leq n \leq 15$) - количество людей. Следующие n строк содержат по n чисел. Если i -ый и j -ый люди дружат, то j -ое число $i + 1$ -ой строки равно 1, иначе - 0.

Выходные данные

В первой строке выходного файла OUTPUT.TXT выведите число групп. Во второй строке выходного файла выведите n чисел (i -ое число - номер группы, в которой находится i -ый человек). Так как в любом случае количество групп не превзойдет n , нумеруйте группы целыми числами от 1 до n . Если решений несколько, то выведите любое.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	3 1 1 0 1 1 0 0 0 1	2 1 1 2
2	8 1	3 1 1 1 1 1 2 2 3

Задача №599 Телешоу

(Время: 2 сек. Память: 16 Мб Сложность: 66%)

В новом интеллектуальном телешоу участнику, проходящему в суперфинал, предлагается следующая игра: на каждом из n секторов большого барабана записывается буква латинского алфавита l_i . После минуты на размышления игрок указывает одну из позиций на барабане i . Его выигрыш вычисляется по такому правилу: для каждой позиции j меньшее из расстояний по и против часовой стрелке от i до j , измеренное в секторах, умножается на абсолютную величину разности номеров в алфавите букв l_i и l_j , после чего все такие величины суммируются.

А Вы можете написать программу, находящую способ получения наибольшего выигрыша?

Входные данные

Первая строка входного файла INPUT.TXT содержит натуральное число n ($1 \leq n \leq 100000$) - размер барабана. Во второй строке задаются разделенные пробелами строчные латинские буквы, записанные на барабане.

Выходные данные

В первой строке выходного файла OUTPUT.TXT выведите наибольший выигрыш, который можно получить при заданном расположении букв на барабане. Во второй строке выведите номер какого-нибудь из секторов, на который игрок должен для этого указать.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	4 r e a r	55 3

Задача №600 Распознавание языка

(Время: 1 сек. Память: 16 Мб Сложность: 28%)

Важным понятием теории формальных грамматик и автоматов является формальный язык. Неформально его можно определить как некоторое множество слов, где под словом понимается некоторая строка из символов.

В этой задаче необходимо проверить, принадлежит ли данное слово языку $\{0^n1^n2^n, n \geq 1\}$. В этот язык входят те и только те слова, которые имеют такую структуру: в них нулей столько же, сколько единиц, а единиц - столько же, сколько и двоек. При этом любой ноль находится ближе к началу слова, чем любая единица, а любая единица находится ближе к началу слова, чем любая двойка. Например, слово 001122 принадлежит этому языку, а слово 0000111122220 - не принадлежит.

Входные данные

Первая строка входного файла INPUT.TXT содержит целое положительное число n ($n \leq 10$) – количество слов, которые надо проанализировать. Далее идут n строк, каждая из которых содержит по одному слову. Слова имеют длину не более тридцати тысяч символов и состоят только из нулей, единиц и двоек. Каждое из слов состоит хотя бы из одного символа.

Выходные данные

Выходной файл OUTPUT.TXT должен содержать ровно n строк. Для каждого слова из входного файла выведите по одной строке, содержащей слово YES, если оно принадлежит указанному выше языку, и NO - иначе.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	3 001122 00011122222 000111222	YES NO YES
2	2 0000111122220 012	NO YES

Задача №601

Цветной лабиринт

(Время: 1 сек. Память: 16 Мб Сложность: 46%)

В одном из парков одного большого города недавно был организован новый аттракцион Цветной лабиринт. Он состоит из n комнат, соединенных m двунаправленными коридорами. Каждый из коридоров покрашен в один из s цветов, при этом от каждой комнаты отходит не более одного коридора каждого цвета. При этом две комнаты могут быть соединены любым количеством коридоров, но никакой коридор не соединяет комнату саму с собой.

Человек, купивший билет на аттракцион, оказывается в комнате номер один. Кроме билета, он также получает описание пути, по которому он может выбраться из лабиринта. Это описание представляет собой последовательность цветов $c_1 \dots c_k$. Пользоваться ей надо так: находясь в комнате, надо посмотреть на очередной цвет в этой последовательности, выбрать коридор такого цвета и пойти по нему. При этом если из комнаты нельзя пойти по коридору соответствующего цвета, то человеку приходится дальше самому выбирать, куда идти.

В последнее время в администрацию парка стали часто поступать жалобы от заблудившихся в лабиринте людей. В связи с этим, возникла необходимость написания программы, проверяющей корректность описания и пути, и, в случае ее корректности, сообщаемой номер комнаты, в которую ведет путь.

Описание пути некорректно, если на пути, который оно описывает, возникает ситуация, когда из комнаты нельзя пойти по коридору соответствующего цвета.

Входные данные

Первая строка входного файла INPUT.TXT содержит два целых числа n ($1 \leq n \leq 10000$) и m ($1 \leq m \leq 100000$) - соответственно количество комнат и коридоров в лабиринте. Следующие m строк содержат описания коридоров. Каждое описание содержит три числа u ($1 \leq u \leq n$), v ($1 \leq v \leq n$), c ($1 \leq c \leq 100$) - соответственно номера комнат, соединенных этим коридором, и цвет коридора. Следующая, $(m+2)$ -ая строка входного файла содержит длину описания пути - целое число k ($0 \leq k \leq 100000$). Последняя строка входного файла содержит k целых чисел, разделенных пробелами, - описание пути по лабиринту.

Выходные данные

В выходной файл OUTPUT.TXT выведите строку INCORRECT, если описание пути некорректно, иначе выведите номер комнаты, в которую ведет описанный путь. Помните, что путь начинается в комнате номер один.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	3 2 1 2 10 1 3 5 5 10 10 10 10 5	3
2	2 2 1 2 10 2 3 5 5 5 10 10 10 10	INCORRECT
	3 2	INCORRECT

3	1 2 10
	1 3 5
	4
	10 10 10 5

Задача №602

Точки на прямой

(Время: 1 сек. Память: 16 Мб Сложность: 32%)

На прямой отмечено n точек. Требуется найти такой отрезок длины l , на котором лежат m из отмеченных точек ($m \geq 2$), что величина l/m минимальна. Считается, что точки, совпадающие с одним из концов отрезка, лежат на нем.

Входные данные

Входной файл INPUT.TXT содержит количество точек n ($2 \leq n \leq 10000$). На второй строке записаны координаты этих точек x_i - целые числа, разделенные пробелами. При этом $|x_i| \leq 30000$ и $x_i < x_j$ при $i < j$.

Выходные данные

В выходной файл OUTPUT.TXT выведите координаты начала и конца найденного отрезка a и b ($a < b$). Если решений несколько, выведите любое.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	3 -2 -1 1	-2 -1

Задача №603

Поиск

(Время: 1 сек. Память: 16 Мб Сложность: 48%)

Обычно программы, предоставляющие возможность поиска заданных строк в текстовых файлах, недостаточно гибко обрабатывают различные пробельные символы. Например, если в некотором тексте слова «Internet» и «olympiad» разделены переводом строки, словосочетание «Internet olympiad» чаще всего не будет обнаружено в этом месте. В данной задаче пробельными символами мы будем считать пробелы, символы табуляции (код символа 9), а так же переводы строк. Любую последовательность идущих подряд непробельных символов будем называть словом.

Ваша программа должна производить обработку одного запроса на поиск словосочетания в тексте. Словосочетание будет задано как последовательность слов, состоящих из цифр и строчных и прописных букв латинского алфавита, каждые два из которых разделены пробелом. Будем считать, что некоторая последовательность символов, первый и последний из которой непробельные, является вхождением этого словосочетания в текст, если после замены каждого блока пробельных символов из этой последовательности на один пробел она совпадет с заданным словосочетанием с точностью до регистра букв. Для представления ответа перед каждым вхождением словосочетания в исходный текст следует поставить символ «@».

Входные данные

Первая строка входного файла INPUT.TXT, заканчивающаяся переводом строки, задает запрос.

Длина словосочетания не превосходит 100 символов. Последующие строки описывают сам текст, размер которого не превосходит 2000 символов. Файл заканчивается переводом строки.

Выходные данные

В выходной файл OUTPUT.TXT выведите результат применения к тексту описанной процедуры. Он должен отличаться от исходного текста только добавлением символов «@».

Пример

№	INPUT.TXT	OUTPUT.TXT
1	internet olympiad Internet Olympiads Everyone is welcome to participate in internet olympiads. Jury of internet olympiads	@Internet Olympiads Everyone is welcome to participate in @internet olympiads. Jury of @internet olympiads

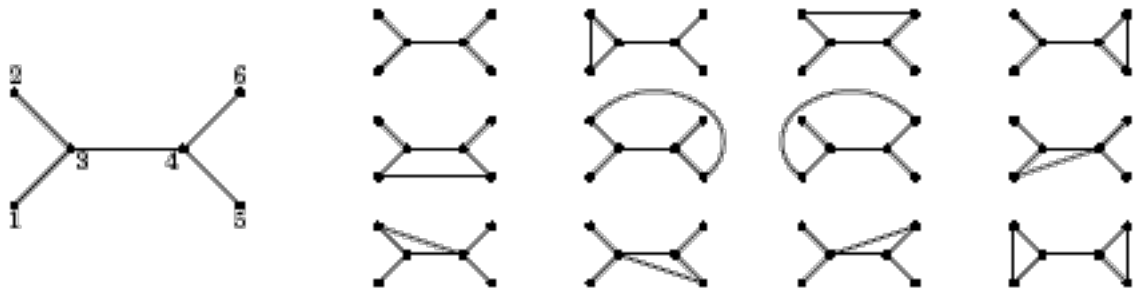
Задача №604 Кактусы

(Время: 1 сек. Память: 16 Мб Сложность: 68%)

Вершинный кактус - это связный неориентированный граф, каждая вершина которого лежит не более, чем на одном простом цикле.

Дерево можно превратить в кактус, добавив в него несколько ребер (поскольку дерево само является кактусом, то можно не добавлять ребер вообще). Вообще говоря, может существовать несколько способов превратить дерево в кактус. Количество способов сделать это назовем *кактусастостью* дерева.

Например, кактусастость дерева, изображенного на картинке слева, равна 12. Двенадцать кактусов, в которые оно может быть превращено, изображены справа.



Для заданного дерева требуется найти его кактусастость.

Входные данные

Первая строка входного файла INPUT.TXT содержит одно целое число n - количество вершин в дереве ($1 \leq n \leq 200$). Следующие $n-1$ строк задают ребра дерева.

Выходные данные

В выходной файл OUTPUT.TXT выведите единственное целое число - кактусастость заданного дерева.

Пример

--	--	--

№	INPUT.TXT	OUTPUT.TXT
1	6 1 3 2 3 3 4 4 5 4 6	12

Задача №605

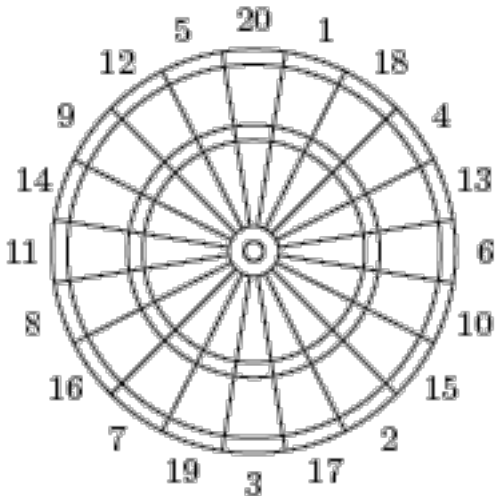
Дартс

(Время: 1 сек. Память: 16 Мб Сложность: 38%)

Игра в дартс очень популярна в Великобритании и Голландии. В игре принимают участие несколько игроков. Они по очереди бросают в мишень по три дротика.

В начале игры каждый игрок имеет некоторое количество очков, обычно 501. За каждое попадание дротика в мишень сумма игрока уменьшается на некоторое число, в зависимости от того, в какую часть мишени он попал. Первый, кто достигает нуля очков, считается победителем.

Внешний вид мишени показан на рисунке справа. Она разделена на 20 секторов, расположенных вокруг небольшого центрального круга. Этот круг, в свою очередь, делится на внутреннюю и внешнюю часть (иногда внутренняя часть называется «яблочко»). Попадание во внешнюю часть центрального круга оценивается 25 очков, а в «яблочко» - вдвое больше, то есть в 50 очков. Стоимость сектора равняется числу, которое на нем написано. Кроме того на мишени выделены два кольца - внешнее и внутреннее. Попадание в них оценивается соответственно в два и в три раза больше, чем в оставшуюся часть соответствующего сектора.



Существуют дополнительные правила для последней серии бросков, в которой игрок должен достичь нуля очков. В этой серии игроку придется бросить в мишень от одного до трех дротиков. Игрок должен достичь в точности нуля очков, получение отрицательной суммы считается ошибкой. Последний дротик должен быть «двойным», то есть попасть во внешнее кольцо какого-либо сектора или в «яблочко» - (оно считается удвоением внешней часть центрального круга).

Например, один из правильных способов закончить игру, имея 50 очков - бросить дротики в «18» и «D16».

Способы «D20», «10», или «20», «T10» не подходят: последний бросок не является удвоенным. Еще один возможный способ победить в этом случае - просто попасть в «яблочко» («Bull»). По количеству оставшихся очков, найдите все способы правильно закончить игру.

Входные данные

Входной файл INPUT.TXT содержит число n - количество оставшихся очков ($1 \leq n \leq 200$).

Выходные данные

В первой строке выходного файла OUTPUT.TXT выведите k - количество способов правильно завершить партию. Каждая из следующих k строк должна содержать описание одного правильного способа. При этом число от 1 до 20 отвечает попаданию в соответствующий сектор. Буква «D» перед числом обозначает попадание во внешнее (удваивающее) кольцо, а «T» - во внутреннее (утраивающее). Внешняя часть центрального круга обозначается как «25», а «яблочко» (bull eye) -

словом «Bull».

Пример

№	INPUT.TXT	OUTPUT.TXT
1	5	7 1 D1 D1 1 2 D1 1 D2 D1 1 D1 T1 D1 2 1 D1 3 D1

Задача №606 Треугольник - 3

(Время: 1 сек. Память: 16 Мб Сложность: 17%)

Даны длины трех отрезков. Требуется проверить: могут ли они являться сторонами треугольника.

Входные данные

Входной файл INPUT.TXT содержит 3 натуральных числа X Y Z – длины заданных отрезков. Длины отрезков записаны в одной строке через пробел и не превышают 1000.

Выходные данные

В выходной файл OUTPUT.TXT выведите YES, если отрезки могут быть сторонами треугольника и NO в противном случае.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	1 2 3	YES
2	1 1 5	NO

Задача №607 Домино в казино

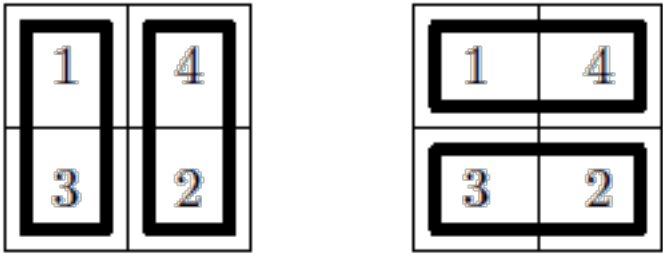
(Время: 1 сек. Память: 16 Мб Сложность: 72%)

Домино известно в качестве игры, в которую люди обычно играют во дворе, чтобы расслабиться после рабочего дня. Но так было лишь до того времени, пока Джон Бигбак не предоставил возможность играть в домино в своем казино «BUMP» (Bring Us Money, Please).

Обычная игра в домино не совсем подходит для казино, поэтому Джон создал свою собственную игру. Партия играется на прямоугольной доске размера m_n. В каждой ее клетке записано некоторое целое число.

У игрока есть k костей домино - прямоугольников 2x1. Он кладет их на доску так, чтобы не возникало наложений, и его выигрыш вычисляется как сумма произведений чисел, накрытых каждой из костей домино.

Например, существует два способа положить две кости домино на доску размера 2x2. Для доски, приведенной ниже, лучший способ положить домино показан слева - в этом случае сумма составляет $1 \times 3 + 4 \times 2 = 11$. Если игрок выберет способ, показанный справа, то сумма составит $1 \times 4 + 3 \times 2 = 10$, что меньше чем 11.



По заданному расположению чисел на доске и количеству костей домино, которыми располагает игрок, найдите наибольшую сумму, которую он может получить.

Входные данные

Первая строка входного файла INPUT.TXT содержит целые числа m , n и k ($1 \leq m \leq 16$, $1 \leq n \leq 100$, $1 \leq k \leq 200$). Следующие m строк содержат по n целых чисел каждая и описывают доску. Числа, записанные на доске, неотрицательны и по величине не превосходят 1000. Гарантируется, что существует хотя бы один способ разместить все кости домино на доске.

Выходные данные

В выходной файл OUTPUT.TXT выведите одно целое число - наибольшую сумму, которую может получить игрок.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	2 2 2 1 4 3 2	11

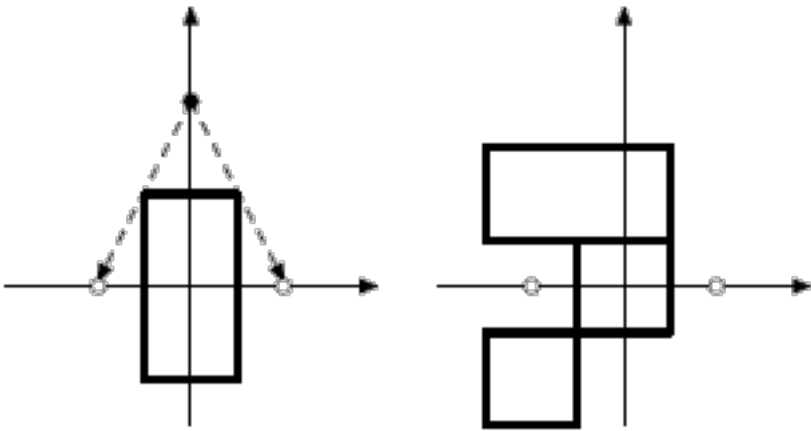
Задача №608 Про любовь

(Время: 1 сек. Память: 16 Мб Сложность: 70%)

Ваське нравится Машка. Она симпатичная, и он очень любит смотреть на нее. Но ему нравится и Ленка! Она тоже довольно симпатичная. Васька может смотреть на нее часами, когда она сидит на скамейке и читает книгу. Почему бы не подойти и поговорить с ней? Есть небольшая проблема. Васька - кот.

Но надо сказать, что это его не очень беспокоит. В общем, он даже привык к этому. Ему просто нравится смотреть на девочек. К сожалению, часто дома и другие препятствия не дают увидеть обеих девочек одновременно. А даже когда это можно сделать, найти подходящую точку довольно сложно. Васька просит вас, своего любимого хозяина, помочь ему.

Даны положения препятствий и точки, в которых находятся девочки. Найдите точку, из которой видны обе девочки, или установите, что такой точки нет. Конечно, Васька не может



забираться внутрь препятствий.

Входные данные

Первая строка входного файла INPUT.TXT содержит два целых числа x_1 и y_1 - координаты Машки. Следующая строка содержит x_2 и y_2 - координаты Ленки. Третья строка входного файла содержит n - количество препятствий ($0 \leq n \leq 10$).

Все препятствия являются прямоугольниками со сторонами, параллельными осям координат. Каждая из следующих n строк содержит четыре целых числа $x_{i,1}$, $y_{i,1}$, $x_{i,2}$ и $y_{i,2}$ - координаты левого нижнего и правого верхнего углов препятствий. Все координаты не превосходят 100 по абсолютному значению. Препятствия не пересекаются, но могут касаться друг друга. Если два препятствия касаются друг друга углами или сторонами, между ними нет зазора. В противном случае Васька может смотреть таким образом, что линия его взгляда касается угла или идет вдоль стороны препятствия.

Ни одна из девочек не находится внутри или на границе какого-то из препятствий. Девочки находятся в разных точках.

Выходные данные

В первой строке выходного файла OUTPUT.TXT выведите «YES» , если точка, из которой Васька может видеть обеих девочек, существует. В этом случае вторая строка должна содержать два вещественных числа - координаты точки, из которой должен смотреть Васька. Эта точка не должна быть внутри какого-либо препятствия, но может быть на его границе. Васька не должен находиться в точке, которая одновременно принадлежит углам двух зданий, не имеющих общей стороны. Координаты должны быть выведены с точностью не менее, чем 10^{-6} . Если искомой точки нет, выведите «NO» в единственную строку выходного файла.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	2 0 -2 0 1 -1 -2 1 2	YES 0.0 4.0
2	2 0 -2 0 3 -1 -1 1 1 -3 -3 -1 -1 -3 1 1 3	NO

Задача №609 Разбиения множества

(Время: 2 сек. Память: 16 Мб Сложность: 69%)

Рассмотрим множество, состоящее из первых n натуральных чисел: $N_n = \{1, 2, \dots, n\}$. Разбиение - это представление этого множества в виде объединения одного или нескольких непустых множеств. Примерами разбиений для $n = 5$ являются:

$$\{1, 2, 3, 4, 5\} = \{1, 2, 3\} \cup \{4, 5\}$$
$$\{1, 2, 3, 4, 5\} = \{1, 3, 5\} \cup \{2, 4\}$$

$\{1, 2, 3, 4, 5\} = \{1, 2, 3, 4, 5\}$
 $\{1, 2, 3, 4, 5\} = \{1\} \cup \{2\} \cup \{3\} \cup \{4\} \cup \{5\}$

Всего существует 52 разбиения множества N_5 . Заметим, что разбиения, отличающиеся только порядком объединяемых множеств, не различаются.

Разбиения множества N_n можно упорядочить лексикографически. Для того, чтобы определить этот порядок, вначале определим лексикографический порядок на подмножествах N_n . Будем говорить, что подмножество A множества N_n лексикографически меньше подмножества B множества N_n и писать $A < B$, если верно одно из следующих утверждений:

- найдется i , такое что i принадлежит A и не принадлежит B , для всех $j < i$, таких что j принадлежит A и B , и найдется $k > i$, такое что k принадлежит B ;
- A включено в B и $i < j$ для всех i из A и j из $B \setminus A$.

Очевидно, что введенное отношение является полным порядком на подмножествах множества N_n . Теперь определим каноническое представление разбиения, как представление, в котором объединяемые множества упорядочены лексикографически.

Разбиения упорядочиваются лексикографически следующим образом. Разбиение $N_n = A_1 \cup A_2 \cup \dots \cup A_k$ лексикографически меньше разбиения $N_n = B_1 \cup B_2 \cup \dots \cup B_l$, если существует такое i , что $A_1 = B_1, A_2 = B_2, \dots, A_{i-1} = B_{i-1}$ и $A_i < B_i$.

По разбиению множества N_n найдите следующее в лексикографическом порядке разбиение.

Входные данные

Входной файл INPUT.TXT содержит несколько описаний тестов. Каждое описание является каноническим представлением разбиения. Первая строка описания содержит n и k - количество элементов в разбиваемом множестве и количество частей в разбиении ($1 \leq n \leq 200$). Последующие k строк содержат элементы разбиения. Элементы каждого множества упорядочены по возрастанию. Описания тестов отделены друг от друга пустыми строками. Последняя строка входного файла содержит два нуля. Этот тест не должен обрабатываться. Сумма n по всем описаниям не превосходит 2000.

Выходные данные

В выходной файл OUTPUT.TXT для каждого теста выведите следующее в лексикографическом порядке разбиение. Если разбиение во входном файле является последним в лексикографическом порядке, выведите первое в лексикографическом порядке. Используйте тот же формат, что и во входном файле. Отделяйте разбиения друг от друга пустыми строками. Элементы каждого множества следует выводить в порядке возрастания.

Пример

№	INPUT.TXT	OUTPUT.TXT
	5 2 1 2 3 4 5	5 2 1 2 3 4 5
	5 2 1 3 5 2 4	5 4 1 4 2

данном районе.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	4 4	6
2	5 7	0
3	2 8	1
4	12 8	102283239429

Задача №611 Словарные квадраты

(Время: 1 сек. Память: 16 Мб Сложность: 50%)

Некоторые наборы из n слов длины n обладают интересным свойством - их можно расположить в клетках квадрата $n \times n$ так, что все слова набора можно прочитать как в вертикали, так и по горизонтали.

Примером такого набора слов является {"DATE", "FIND", "IDEA", "NEXT"}. Их можно расположить так:

F	I	N	D
I	D	E	A
N	E	X	T
D	A	T	E

Заметьте, что каждое слово можно прочитать как по горизонтали, так и по вертикали. Такие квадраты называются словарными квадратами, наибольший известный словарный квадрат в английском языке имеет размер 10×10 .

Рассмотрим еще один пример словарного квадрата:

C	R	A	B
R	A	R	E
A	R	T	S
B	E	S	T

Вам даны такие $2n$ слов, что из них можно построить два различных словарных квадрата размера $n \times n$. Ваша задача состоит в том, чтобы разбить эти слова на две группы, по n слов в каждой, и построить из слов каждой группы словарный квадрат.

Гарантируется, что все данные вам слова являются английскими словами (некоторые из них могут быть достаточно редкими словами, именами, или специальными терминами).

Входные данные

Первая строка входного файла INPUT.TXT содержит целое число n ($2 \leq n \leq 10$). Каждая из следующих $2n$ строк содержит слово, состоящее из заглавных букв латинского алфавита. Каждое слово содержит ровно n букв.

Выходные данные

В выходной файл OUTPUT.TXT выведите два словарных квадрата, построенных из данных

слов. Разделите квадраты пустой строкой.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	4	CRAB
	ARTS	RARE
	BEST	ARTS
	CRAB	BEST
	DATE	
	FIND	FIND
	IDEA	IDEA
	NEXT	NEXT
	RARE	DATE

Задача №612 Подстрока

(Время: 1 сек. Память: 32 Мб Сложность: 58%)

Рассмотрим слова, состоящие из букв «А», «В», «а» и «b». Скажем, что два слова эквивалентны, если одно может быть получено из другого с помощью следующих операций:

- удалить в любой позиции подстроку, равную Aa, aA, Bb или bB;
- добавить в любую позицию подстроку, равную Aa, aA, Bb или bB.

Например, слова abAaBBabbA и aAaBabaAbA эквивалентны:

abAaBBabbA → abBBabbA → aBabbA → aAaBabbA → aAaBabaAbA,

а слова abAB и baBA - нет.

Интересно отметить, что для произвольных слов X и Y найдется такое слово Z, эквивалентное X, которое содержит Y в качестве подстроки. Ваша задача - найти кратчайшее такое слово.

Входные данные

Первая строка входного файла INPUT.TXT содержит X. Вторая строка содержит Y. Оба слова непустые и каждое из них имеет длину не больше 2000.

Выходные данные

В выходной файл OUTPUT.TXT выведите одну строку, содержащую минимальное по длине слово, эквивалентное X, содержащее Y в качестве подстроки. Если решений несколько, выведите любое.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	abAaBBabbA AaBaba	aAaBabaAbA

Задача №613 Поле чудес - 2

Всем известно сверхпопулярное теле-шоу «Поле чудес».

В игре принимают участие n человек. Их цель - отгадать загаданное слово. Изначально известна только длина слова l , а буквы на специальном табло закрыты черными прямоугольниками. В свой ход игрок называет одну букву. Если эта буква встречается в слове, то все ее вхождения открываются, и игрок делает еще один ход. Если в слове нет такой буквы, то ход передается следующему игроку. От последнего игрока ход передается первому. Выигрывает тот игрок, который отгадывает последнюю букву.

Пусть, например, загадано секретное слово «CONTEST». Изначально игроки видят только «----
---». Если первый игрок скажет 'Е', то она открывается, и теперь табло выглядит как «----Е--». Первый игрок делает еще один ход - пусть, например, он называет 'А'. В слове нет такой буквы, поэтому ход передается следующему игроку. Если тот скажет 'Т', то игроки увидят «---ТЕ-Т», и так далее.

Пашин друг собирается поучаствовать в игре. Павлу интересно: каковы шансы на победу его друга. Паша оценил интеллектуальный потенциал каждого из игроков q_t . Вероятность того, что игрок t правильно отгадает букву в ситуации, когда еще не были названы i букв, неизвестны еще j разных букв слова, и на табло осталось k закрытых букв, вычисляется по следующей формуле:

$$p_{t,i,j,k} = \left(1 - q_t^{\frac{1}{k-j+1}}\right) \frac{j}{i} + q_t^{\frac{1}{k-j+1}} \left(1 - \frac{1}{i}\right)^{i-j}$$

Здесь мы будем считать, что $0^0 = 1$.

Вероятности угадывания каждой из все еще неизвестных букв слова одинаковы. По заданным n , очереди хода друга Павла r , загаданному слову и значениям интеллектуального потенциала игроков, определите вероятность того, что Пашин друг победит в игре.

Входные данные

Первая строка входного файла INPUT.TXT содержит n и r ($2 \leq n \leq 10, 1 \leq r \leq n$). На второй строке записано загаданное слово. Оно состоит из больших букв латинского алфавита, и его длина не превосходит 12. Третья строка содержит n вещественных чисел - значения интеллектуального потенциала игроков ($0 \leq q_t \leq 0.99$).

Выходные данные

В выходной файл OUTPUT.TXT выведите одно число - вероятность победы друга Павла. Ответ должен быть дан с точностью, не хуже, чем 10^{-8} .

Пример

№	INPUT.TXT	OUTPUT.TXT
1	3 1 CONTEST 0.7 0.2 0.1	0.4648222937

Задача №614
Скобки - 3

Рассмотрим скобочные последовательности с одним типом скобок. Для заданной скобочной

последовательности найдите количество ее подпоследовательностей, которые являются правильными скобочными последовательностями.

Например, для последовательности "((()())(" таких последовательностей восемь: "((()())()", "(()())", "((()())", "(()())", "(()())", "()" и "".

Входные данные

Входной файл INPUT.TXT содержит последовательность, состоящую не более чем из 300 круглых скобок.

Выходные данные

В выходной файл OUTPUT.TXT выведите количество различных правильных скобочных подпоследовательностей заданной последовательности.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	((()())()	8

Задача №615 Новый год

(Время: 1 сек. Память: 16 Мб Сложность: 71%)

Санта Клаус решил сделать новогодний подарок своим эльфам. Конечно же, Санта был бы не Санта, если бы его подарок был обыкновенным. Подарок просто замечательный - эльфы идут в кино!

Конечно же, если все эльфы пойдут в кино одновременно, это может показаться кому-нибудь подозрительным. Ну, вы понимаете, некоторые люди все еще не верят в эльфов. Так что каждый день ровно одна пара эльфов пойдет в кино. Конечно, идти в кино самому по себе неинтересно, гораздо интереснее пойти с другом или подругой. Таким образом, каждый день один мальчик-эльф и одна девочка-эльф вместе пойдут в кино.

Но эльфы, они такие разные. Каждый эльф четко знает, с какими эльфами и на какой фильм он или она хотели бы пойти. Поскольку у каждой пары эльфов свои взаимоотношения, для каждой пары, которая согласна вместе пойти в кинотеатр, известно какой именно фильм они хотели бы посмотреть.

Но на самом деле все еще хуже. Дело в том, что билеты в кино стоят денег. Конечно, Санта не беден, но ему нужно много денег на подарки детям. Так что собрав у эльфов пожелания, кто куда и с кем хотел бы пойти в кино, Санта решил удовлетворить некоторые пожелания таким образом, чтобы потратив как можно меньше денег, добиться того, что каждый эльф хотя бы один раз побывает в кино.

Помогите Санте, он пока не до конца освоился с компьютером.

Входные данные

Первая строка входного файла INPUT.TXT содержит два целых числа n и m - количество мальчиков-эльфов и девочек-эльфов, соответственно ($1 \leq n, m \leq 100$). Вторая строка содержит r - количество пожеланий, которое Санта получил от своих эльфов ($1 \leq r \leq 1500$). Следующие r строк содержат по три целых числа a_i , b_i и c_i каждая. Числа означают, что мальчик-эльф a_i хотел бы пойти в кино с девочкой-эльфом b_i , и билет на фильм, на который они хотели бы пойти, стоит c_i (1

$\leq c_i \leq 1000$). Каждый эльф хочет пойти в кино хотя бы с одним другим эльфом.

Выходные данные

На первой строке выходного файла OUTPUT.TXT выведите минимальную сумму, которую Санте придется потратить, чтобы сделать своим эльфам подарок. На второй строке выведите k - количество билетов, которое Санта должен купить. Наконец, третья строка должна содержать k целых чисел – номера пожеланий, которые следует удовлетворить.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	3 3	11
	7	4
	1 1 3	2 3 4 6
	1 2 2	
	1 3 4	
	2 1 3	
	2 2 9	
	3 1 2	
	3 3 11	

Задача №616 Отношения

(Время: 1 сек. Память: 16 Мб Сложность: 54%)

Бинарным отношением R на множестве X называется множество упорядоченных пар элементов из X . Если X конечно и содержит n элементов, то отношение можно задать как квадратную булеву матрицу размера $n \times n$.

В некоторых случаях бывает важно задать отношение в более компактной форме. Один из способов компактного задания отношений применяется, в частности, при описании грамматик операторного предшествования.

Рассмотрим две функции f и g , каждая из которых сопоставляет элементам X целые числа. Будем говорить, что эти функции описывают отношение R , если для любых x и y из X пара (x, y) принадлежит R тогда и только тогда, когда $f(x) \leq g(y)$.

По заданному отношению R , найдите способ описать его указанным образом с помощью двух функций f и g , либо выясните, что это невозможно сделать.

Входные данные

Первая строка входного файла INPUT.TXT содержит n - количество элементов в множестве $X = \{x_1, x_2, \dots, x_n\}$ ($1 \leq n \leq 1\,000$). Следующие n строк описывают отношение R . Каждая строка содержит n символов, j -й символ i -й из этих строк равен 1, если (x_i, x_j) принадлежит R , и 0 в противном случае.

Выходные данные

На первой строке выходного файла OUTPUT.TXT выведите "YES", если отношение можно описать указанным образом. В этом случае вторая строка должна содержать n целых чисел в диапазоне от -10^9 до 10^9 - значения функции f на элементах x_1, x_2, \dots, x_n , соответственно, а третья

строка должна описывать функцию g аналогичным образом. Если отношение нельзя закодировать описанным образом с помощью двух функций, выведите "NO" на первой строке выходного файла.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	3 111 110 100	YES 0 1 2 2 1 0
2	3 110 101 011	NO

Задача №617 Лады

(Время: 2 сек. Память: 16 Мб Сложность: 70%)

Напомним, что ладья - это шахматная фигура, которая ходит по вертикалям и горизонталям.

Будем называть расстановку k белых и l черных ладей на доске размера m x n мирной, если ни одна ладья не угрожает ладье другого цвета.

Для заданных m, n, k и l найдите количество мирных расстановок ладей на доске.

Входные данные

Входной файл INPUT.TXT содержит четыре целых числа: m, n, k и l ($2 \leq m, n \leq 10$, $1 \leq k, l \leq 1$, $k+l \leq m \cdot n$).

Выходные данные

В выходной файл OUTPUT.TXT выведите одно целое число - количество мирных расстановок k белых и l черных ладей на доске размера m x n.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	3 3 2 2	18

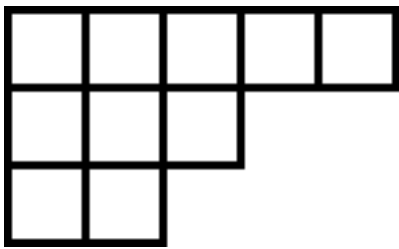
Задача №618 Диаграммы Юнга

(Время: 1 сек. Память: 16 Мб Сложность: 74%)

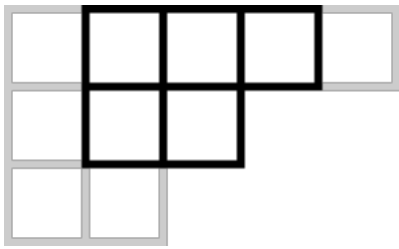
Диаграммы Юнга используются для того, чтобы изобразить разбиение числа на слагаемые. Разбиение числа n на слагаемые представляет собой сумму вида $n = m_1 + m_2 + \dots + m_k$, где $m_1 \geq m_2 \geq \dots \geq m_k$.

Диаграмма состоит из n квадратиков, организованных в виде k рядов, где k количество слагаемых в разбиении. Ряд, соответствующий числу m_i , содержит m_i квадратиков. Все ряды выровнены по левому краю и упорядочены от более длинного к более короткому.

Например, диаграмма Юнга, приведенная на рисунке, соответствует разбиению $10 = 5 + 3 + 2$.



Иногда можно вписать одну диаграмму Юнга в другую. Диаграмму X можно вписать в диаграмму Y , если можно удалить некоторые квадратики из диаграммы Y так, чтобы получилась диаграмма X . Отметим, что разрешается только удалять некоторые квадратики, вращать или отражать диаграмму не разрешается. Например, диаграмма для разбиения $5 = 3 + 2$ может быть вписана в диаграмму для разбиения $10 = 5 + 3 + 2$, как показано на рисунке.



С другой стороны, диаграмму для разбиения $8 = 4 + 4$ нельзя вписать в диаграмму для разбиения $10 = 5 + 3 + 2$. Для заданного n найдите такое разбиение n на слагаемые, что в соответствующую ему диаграмму Юнга можно вписать максимальное количество различных диаграмм. Например, в диаграмму для разбиения $10 = 5 + 3 + 2$ можно вписать 36 различных диаграмм. Однако это не максимальное значение. В диаграмму для разбиения $10 = 4 + 2 + 2 + 1 + 1$ можно вписать 41 диаграмму Юнга.

Входные данные

Входной файл INPUT.TXT содержит целое число n ($1 \leq n \leq 100$).

Выходные данные

На первой строке выходного файла OUTPUT.TXT выведите максимальное число диаграмм Юнга, которые можно вписать в некоторую диаграмму, соответствующую разбиению на слагаемые числа n . На второй строке выведите одно или более целых чисел - количество квадратиков в каждом из рядов оптимальной диаграммы.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	10	41 4 2 2 1 1

Задача №619 Бросание кубика

(Время: 1 сек. Память: 16 Мб Сложность: 42%)

Кубик, грани которого помечены цифрами от 1 до 6, бросают N раз. Требуется найти вероятность того, что сумма выпавших чисел будет равна Q .

Входные данные

Входной файл INPUT.TXT содержит натуральные числа N и Q ($N \leq 500, Q \leq 3000$).

Выходные данные

В выходной файл OUTPUT.TXT выведите единственное вещественное число – искомую вероятность, которая должна отличаться от истинного значения не более чем на 10^{-6} .

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	1 6	0.166667
2	1 7	0
3	4 14	0.112654
4	100 100	1.530647E-78

Задача №620 Бассейн реки

(Время: 1 сек. Память: 16 Мб Сложность: 67%)

Дана карта рек некоторого континента. Каждая река показана как ломаная линия, которая начинается у истока реки и заканчивается или в точке, где река впадает в другую, или устьем реки. Вершины ломаной - или точки поворота реки, или точки впадения притоков.

Будем рассматривать бассейн реки как выпуклый многоугольник минимальной площади, который содержит реку и все её притоки. Одна и та же территория может принадлежать бассейнам различных рек.

В качестве примера приведем континент с тремя реками. Координаты рек и площади бассейнов даны в таблице:

Название реки	x	y	Площадь бассейна реки без притоков	Рисунок
река 1	6	9	12,5	
	5	11		
	3	12		
	2	10		
	1	7		
река 2	7	9	1,5	
	5	7		
	5	5,5		
река 3	3	10	9,5	
	5	8		
	4	6		
	5	5,5		
	6	5		
	3	5		

Требуется найти максимальную площадь бассейна реки, расположенной на данном континенте.

Входные данные

Первая строка входного файла INPUT.TXT содержит число рек N. В следующих строках файла содержится N блоков, описывающих реки. Каждый блок номер i состоит:

- из одной строки с k_i - числом вершин ломаной, представляющей реку;
- k_i строк, содержащих пары вещественных чисел x_j и y_j ($1 \leq j \leq k_i$), разделённых пробелом, - координаты точек, описывающих реку.

Ограничения: $1 \leq N \leq 10$, сумма $k_i \leq 1000$, $-1000 \leq x_j, y_j \leq 1000$.

Выходные данные

В выходной файл OUTPUT.TXT выведите одно число - площадь наибольшего бассейна реки с двумя знаками после запятой.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	3 5 6 9 5 11 3 12 2 10 1 7 3 7 9 5 7 5 5.5 6 3 10 5 8 4 6 5 5.5 6 5 3 5	16.00
2	2 5 6 9 5 11 3 12 2 10 1 7 6 3 10 5 8 4 6 5 5.5 6 5 3 5	12.50

(Время: 1 сек. Память: 16 Мб Сложность: 46%)

Дана матрица A размером N×N, заполненная неотрицательными целыми числами. Расстояние между двумя элементами A_{ij} и A_{pq} определено как $|i - p| + |j - q|$. Требуется заменить каждый нулевой элемент матрицы ближайшим ненулевым. Если есть две или больше ближайших ненулевых ячейки, нуль должен быть оставлен.

Входные данные

В первой строке входного файла INPUT.TXT содержится натуральное число N ($N \leq 200$). Затем идут N строк по N чисел, разделённых пробелами и представляющих собой матрицу. Элементы матрицы не превосходят значения 10^6 .

Выходные данные

В выходной файл OUTPUT.TXT выведите N строк по N чисел, разделённых пробелами, - модифицированную матрицу.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	3	1 0 2
	0 0 0	1 0 2
	1 0 2	0 3 0
	0 3 0	

Задача №622 Прямоугольное деление

(Время: 1 сек. Память: 16 Мб Сложность: 53%)

Дано N прямоугольников со сторонами, параллельными осям координат. Требуется определить, на сколько частей эти прямоугольники разбивают плоскость (внутри частей не должно быть границ прямоугольников).

Входные данные

В первой строке входного файла INPUT.TXT содержится число прямоугольников N ($1 \leq N \leq 100$). Далее идут N строк, содержащих по 4 числа x_1, y_1, x_2, y_2 - координаты двух противоположных углов прямоугольника. Координаты представляют собой целые числа и по абсолютной величине не превосходят 10 000.

Выходные данные

В выходной файл OUTPUT.TXT выведите одно число - количество частей, на которые разбивается плоскость.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	3	6
	10 20 50 30	
	40 10 50 25	
	40 25 80 30	

Задача №623

Снова Фибоначчи

(Время: 1 сек. Память: 16 Мб Сложность: 23%)

Вам наверняка знакомы числа Фибоначчи: 1, 1, 2, 3, 5, 8, 13, 21... Они определяются рекуррентным соотношением: $F_n = F_{n-1} + F_{n-2}$, $F_0 = F_1 = 1$.

Требуется найти последнюю цифру n -го числа Фибоначчи.

Входные данные

Во входном файле INPUT.TXT содержится одно целое число n ($0 \leq n \leq 10^8$).

Выходные данные

В выходной файл OUTPUT.TXT необходимо вывести одно число - последнюю цифру числа F_n .

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	1	1
2	5	8

Задача №624

Электронная почта

(Время: 2 сек. Память: 16 Мб Сложность: 61%)

Современный мир немыслим без Интернета и электронной почты. Для того, чтобы людям было проще ориентироваться в потоке поступающих писем, были созданы специальные программы - почтовые клиенты. Фирма TIRLABS занимается разработкой почтового клиента The Bar!.

Недавно программисты компании завершили разработку очередной, 366239-ой, версии этого почтового клиента. Менеджеры по продажам и рекламе уже готовы всю рекламировать и продавать эту новую программу. Однако, генеральный директор компании TIRLABS считает, что любая программа должна быть подвергнута всестороннему тестированию, прежде чем она будет продаваться. «Да и не работающую программу, скорее всего, никто не купит!» - сказал он.

Одним из видов тестирования сложных программ является так называемое стресс-тестирование. При нем программа тестируется в экстремальных условиях, часто даже в таких, на какие она не рассчитана. Для тестирования The Bar! ver. 366239 был выбран такой метод: программа запускается на n компьютерах, стоящих в одной комнате, после чего с компьютеров друг на друга посылаются m писем. При этом никакие два события (отправление или прием письма) не происходят одновременно, а сеть настолько надежна, что все письма доходят до адресата. Адресат у каждого письма при этом только один.

Почтовый клиент The Bar! в процессе работы ведет протокол, в который заносятся идентификаторы отправленных и полученных писем в том порядке, в котором они были обработаны почтовым клиентом. При этом при оценке результатов тестирования в расчет принимаются только события, отраженные в этом протоколе.

Программа считается правильно работающей по результатам тестирования, если всем событиям, указанным в протоколах, можно сопоставить моменты времени таким образом, что никакие два события не происходят одновременно, и каждое из писем отправляется до того, как

получается. При этом, разумеется, внутри каждого из протоколов порядок событий должен остаться прежним.

Даны протоколы работы почтового клиента на каждом из компьютеров. Напишите программу, проверяющую, можно ли по результатам этого тестирования признать программу правильно работающей.

Входные данные

Входной файл INPUT.TXT содержит несколько наборов входных данных. Первая строка содержит t - число наборов входных данных. Оставшиеся строки входного файла содержат эти наборы.

Описание каждого набора начинается со строки, содержащей два целых числа n ($1 \leq n \leq 50000$) и m ($1 \leq m \leq 100000$) - количество компьютеров и отправленных писем соответственно. Далее следуют n строк, i -ая из которых содержит протокол работы почтового клиента на i -ом компьютере. Протокол работы состоит из целого числа k_i ($0 \leq k_i \leq 2m$) и k_i чисел $a_{i,j}$, описывающих события. Если $a_{i,j} > 0$, то j -ым по счету событием на i -ом компьютере была посылка письма с идентификатором $a_{i,j}$, если же $a_{i,j} < 0$, то j -ым по счету событием на i -ом компьютере было получение письма с идентификатором $-a_{i,j}$. Нулю $a_{i,j}$ равно быть не может.

Идентификатор письма - это целое число от 1 до 10^6 . Внутри одного набора входных данных все письма имеют различные идентификаторы. Гарантируется, что все письма, которые были отправлены, были кем-то приняты, то есть сумма всех k_i в одном наборе входных данных равна $2m$.

Сумма чисел n по всем наборам входных данных не превосходит 50000, сумма чисел m по всем наборам входных данных не превосходит 100000.

Выходные данные

В выходной файл OUTPUT.TXT для каждого набора входных данных выведите ровно одну строку. Эта строка должна содержать слово YES, если программу можно считать правильно работающей по результатам тестирования, и NO - в противном случае.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	2	YES NO
	2 2	
	2 1 -2	
	2 2 -1	
	2 2	
	2 -2 1	
	2 -1 2	
2	1	YES
	2 3	
	2 1 -1	
	4 239 -239 366 -366	

Задача №625 SMS - 2

(Время: 1 сек. Память: 16 Мб Сложность: 40%)

В наше время непросто найти человека, который ни разу в жизни не использовал мобильный телефон для отправления текстовых сообщений. Старые телефоны поддерживали только один способ набора текста, упрощенно описывающийся следующими правилами: Каждой из восьми кнопок от «2» до «9» ставится в соответствие несколько букв, а кнопка «1» отводится для знаков препинания. Пробел ставится нажатием кнопки «0». Для ввода первого из соответствующих кнопке символов, ее надо нажать один раз, для ввода второго - два раза и так далее. Если два подряд идущие символа (кроме пробела) сопоставлены одной кнопке, то после ввода первого из них можно, либо подождать, либо нажать на кнопку перемещения курсора (второй вариант оказывается быстрее). Для переключения регистра букв используется кнопка «#». Кроме того, для удобства, после ввода вопросительного и восклицательного знаков, а так же точки (если на этот момент включен нижний регистр), активируется режим «первой заглавной буквы». При этом следующая буква автоматически печатается заглавной, после чего опять включается нижний регистр. В начале набора включен режим «первой заглавной буквы».

В последнее время в Интернете стали появляться результаты различных исследований, доказывающих неэффективность обычной раскладки телефонной клавиатуры, в которой буквы сопоставляются цифрам в алфавитном порядке. Составьте программу, которая по заданному сопоставлению букв кнопкам, будет находить минимальное количество нажатий, необходимых для максимально быстрого ввода данного текста при условии, что на каждое нажатие уходит одинаковое количество времени.

Входные данные

Строки с первой по девятую входного файла INPUT.TXT задают символы, сопоставленные соответствующим кнопкам. Следующая строка содержит сообщение длиной от 1 до 1000 символов. Гарантируется, что в первой строке находятся символы «?», «!» и «.», заданные в определенном порядке; в последующих 8 строках расположены все строчные латинские символы от «a» до «z» без повторов; текст сообщения содержит только те символы, которые возможно напечатать.

Выходные данные

В выходной файл OUTPUT.TXT выведите единственное число - минимальное количество нажатий на кнопки, требующееся для наискорейшего ввода сообщения.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	.?! abc def ghi jkl mno pqrs tuv wxyz Hello. How do you do? i hope everything is fine See ya!	120

Задача №626 Преобразователь строк

(Время: 1 сек. Память: 16 Мб Сложность: 52%)

Преобразователь строк - это специальная компьютерная программа. Она получает на вход

строку S и набор правил преобразования строки. Каждое из правил имеет вид $c_1c_2 \rightarrow E$, где c_1 и c_2 - маленькие буквы латинского алфавита. Символом E здесь обозначена пустая строка. При этом каждый символ присутствует не более, чем в одном правиле.

Преобразователь строк работает по шагам. За один шаг он находит в текущей строке подстроку, совпадающую с правой частью одного из правил, после чего эта подстрока удаляется из текущей строки (этот процесс называется применением правила). Этот процесс продолжается до тех пор, пока существует правило, которое можно применить. Строка, которая остается к моменту, когда нельзя применить никакое правило, считается результатом T работы преобразователя строк.

Пусть, например, набор правил таков: $\{ab \rightarrow E, cd \rightarrow E\}$, а исходная строка $S = aabbccdba$. Тогда работа преобразователя будет выглядеть так: $aabbccdba \rightarrow abccdba \rightarrow ccdba \rightarrow cba$, и результатом T работы преобразователя будет строка cba.

Ваша задача состоит в том, чтобы написать программу, моделирующую работу преобразователя строк с заданным набором правил на заданной строке S.

Входные данные

Первая строка входного файла INPUT.TXT содержит целое число n ($0 \leq n \leq 13$) - количество правил. Далее идут n строк, каждая из которых описывает одно из правил. Гарантируется, что каждая из маленьких букв латинского алфавита присутствует не более, чем в одном правиле.

Последняя, (n+2)-ая строка входного файла содержит исходную строку S. Она не пуста и состоит только из маленьких букв латинского алфавита. Длина S не превосходит 100000.

Выходные данные

В выходной файл OUTPUT.TXT выведите строку T - результат работы преобразователя на строке S.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	2 ab cd aabbccdba	cba
2	0 abcdefghijklmnopqrstuvwxyz	abcdefghijklmnopqrstuvwxyz

Задача №627 Игра в слова

(Время: 1 сек. Память: 16 Мб Сложность: 68%)

Существует множество игр, которые называются «игра в слова». В одной из вариаций правила следующие: выбирается длинное слово, после чего два игрока пытаются вспомнить все слова, которые можно составить, используя некоторые буквы загаданного слова. После этого они по очереди называют придуманные слова (называть одно слово два раза запрещается). Если первый игрок назвал больше слов, то он побеждает, если у обоих игроков слова закончились одновременно, засчитывается ничья, в оставшемся случае побеждает второй.

Оказывается, в этой игре важно не только знать много слов, но и придерживаться правильной стратегии. Напишите программу, которая будет узнавать, кто победит, если оба игрока будут играть идеально, если первый игрок будет играть оптимально, а второй - наихудшим образом, и

если первый игрок будет поддаваться. При этом можно считать, что, придумав слова, игроки записывают их на бумагу, и каждый видит записи предыдущего. Если на ходу того игрока, который поддается, у него остаются не упоминавшиеся ранее слова, он обязан назвать одно из них.

Входные данные

В первой строке входного файла INPUT.TXT записано загаданное слово. Затем описываются слова, которые знает первый игрок - на отдельной строке целое число n_1 ($0 \leq n_1 \leq 10000$), за которым следуют n_1 слов по одному на строке. После этого задается число n_2 слов, известных второму игроку, и описываются сами эти слова в таком же формате. Все слова состоят из маленьких букв латинского алфавита, а их длины не превосходят 100 символов.

Выходные данные

В выходной файл OUTPUT.TXT выведите три числа по одному на строке - ответы для случаев, когда оба игрока играют оптимально и когда поддаются первый и второй игроки соответственно: номер выигрывающего игрока и 0 в случае ничьи.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	internationalization	0
	6	2
	zone	1
	oil	
	rent	
	impression	
	noir	
	trail	
	7	
	teal	
	creativity	
	rent	
	rain	
	oil	
	zealot	
	zone	

Задача №628 Clear World and Brothers

(Время: 2 сек. Память: 16 Мб Сложность: 46%)

Наконец в деревнях Виллорибо и Виллобаджо закончились праздники. Перемыта вся посуда! Этот процесс прошел так быстро и непринужденно, что братьями Карлионе было решено открыть сеть агентств «Clear World and Brothers», специализирующихся на профессиональном мытье посуды. В области Новая Берляндия, где и находятся знаменитые деревни, всего N деревень. Система координат введена так, что Виллорибо имеет координаты $(x_1, 0)$, а Виллобаджо - $(x_2, 0)$. Координаты всех деревень целые числа не превосходящие по модулю 10^6 . Вы работаете на мистера Берлионе старшего и ваша задача найти оптимальное расположение для регионального отделения «Clear World and Brothers», то есть сумма расстояний от агентства до всех деревень должна быть наименьшей и агентство обязательно должно располагаться на прямолинейном шоссе Виллорибо-Виллобаджо (возможно расположение не только внутри, но и на границе отрезка).

Входные данные

В первой строке входного файла INPUT.TXT записано натуральное число N ($2 \leq N \leq 15000$). Далее в N строках записаны пары координат всех вершин. Виллорибо и Виллобаджо первая и вторая деревня соответственно. Возможно, что сколько-то деревень расположены так близко, что их координаты совпадают.

Выходные данные

В выходной файл OUTPUT.TXT выведите абсциссу оптимального расположения агентства. Разрешается абсолютная или относительная погрешность не более 10^{-6} .

Пример

№	INPUT.TXT	OUTPUT.TXT
1	4 -10 0 10 0 3 1 1 -1	2.000000

Задача №629 Сочетания

(Время: 1 сек. Память: 16 Мб Сложность: 55%)

Петя выписал все сочетания из N первых латинских букв по K букв. В каждом сочетании он выписывал буквы в лексикографическом порядке. Сочетания он выписывал в лексикографическом порядке по одному в строке. Теперь он хочет узнать: какое слово записано в M -ой строке.

Входные данные

Во входном файле INPUT.TXT записаны целые числа N , K , M ($1 \leq N \leq 26$, $1 \leq K \leq N$). Гарантируется, что M не превосходит количества всех выписанных сочетаний.

Выходные данные

В выходной файл OUTPUT.TXT выведите M -ое выписанное сочетание.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	4 2 3	ad

Пояснение

Все сочетания в порядке их записи: ab, ac, ad, bc, bd, cd. Здесь 3м по счету сочетанием является ad.

Задача №630 Охрана

(Время: 1 сек. Память: 16 Мб Сложность: 42%)

На секретной военной базе работает N охранников. Сутки поделены на 10000 равных промежутков времени, и известно когда каждый из охранников приходит на дежурство и уходит с него. Например, если охранник приходит в 5, а уходит в 8, то значит, что он был в 6, 7 и 8-ой промежуток. В связи с уменьшением финансирования часть охранников решено было сократить. Укажите: верно ли то, что для данного набора охранников, объект охраняется в любой момент времени хотя бы одним охранником и удаление любого из них приводит к появлению промежутка времени, когда объект не охраняется.

Входные данные

В первой строке входного файла INPUT.TXT записано натуральное число K ($1 \leq K \leq 100$) – количество тестов в файле. Каждый тест начинается с числа N ($1 \leq N \leq 10000$), за которым следует N пар неотрицательных целых чисел A и B - время прихода на дежурство и ухода ($0 \leq A < B \leq 10000$) соответствующего охранника. Все числа во входном файле разделены пробелами и/или переводами строки.

Выходные данные

В выходной файл OUTPUT.TXT выведите K строк, где в М-ой строке находится слово Accepted, если М-ый набор охранников удовлетворяет описанным выше условиям. В противном случае выведите Wrong Answer.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	2 3 0 3000 2500 7000 2700 10000 2 0 3000 2700 10000	Wrong Answer Accepted

Задача №631 Отгадай число

(Время: 1 сек. Память: 16 Мб Сложность: 60%)

Двое играют в игру. Первый игрок (ведущий) загадал число от 1 до N. Второй хочет отгадать это число. Для достижения цели он задает вопросы. Каждый вопрос имеет вид: «Содержится ли загаданное число в наборе S?», где S - произвольный набор чисел от 1 до N. В случае положительного ответа второй игрок платит две конфеты ведущему, в противном случае - одну.

Какое наименьшее количество конфет должен иметь второй игрок, чтобы наверняка суметь отгадать число?

Входные данные

Входной файл INPUT.TXT содержит целое число N ($1 \leq N \leq 10^5$).

Выходные данные

В выходной файл OUTPUT.TXT выведите искомое количество конфет.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	6	5

2	2	2
---	---	---

Задача №632

Отрезки

(Время: 1 сек. Память: 16 Мб Сложность: 79%)

Дан прямоугольник на координатной плоскости с левым нижним углом в точке (0, 0), а правым верхним - в точке (W, H) и отрезки, параллельные осям координат. Отрезки задаются координатами своих концов. Эти отрезки разрезают прямоугольник на несколько частей (возможно, одну). Требуется определить их площади. Отрезки могут пересекаться, накладываться и вырождаться в точку. Все координаты - целые числа по модулю не превосходящие 10000.

Входные данные

В первой строке входного файла INPUT.TXT указываются числа W и H ($1 \leq W, H \leq 10000$). Во второй строке N ($0 \leq N \leq 50$) - количество отрезков. Далее в N строках через пробел указываются числа A_i, B_i, C_i, D_i - координаты концов i-го отрезка: (A_i, B_i) и (C_i, D_i) .

Выходные данные

Выходной файл OUTPUT.TXT должен содержать последовательность положительных чисел — площади областей, записанные в порядке не возрастания.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	3 3 3 1 3 1 1 1 2 4 2 2 0 2 8	5 2 1 1

Задача №633

ACM World Finals

(Время: 1 сек. Память: 16 Мб Сложность: 20%)

Некоторые из вас, наверно, знают, что ежегодно проводится чемпионат мира по программированию среди студентов (<http://acm.baylor.edu>). В финал этого соревнования проходят около 80 команд со всего мира.

Каждая команда состоит из трех человек и имеет название. Напишите программу, которая по краткому названию команды и фамилиям ее участников, строит полное название команды.

Полное название команды состоит из краткого названия команды и списка фамилий ее участников. Фамилии участников в списке должны быть упорядочены по алфавиту и отделены друг от друга запятыми. Название команды от фамилий участников должно быть отделено двоеточием. После каждого знака препинания должен стоять ровно один пробел.

Входные данные

Входной файл INPUT.TXT содержит ровно 4 строки. Первая из строк содержит название команды. Каждая из следующих трех строк содержит фамилию одного из членов команды. Длины строк не превышают 50 символов.

Выходные данные

Выходной файл OUTPUT.TXT должен содержать ровно одну строку, содержащую полное название команды.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	Dream Team Knuth Dijkstra Cormen	Dream Team: Cormen, Dijkstra, Knuth
2	Ivanovs Team Ivanov Ivanov Ivanov	Ivanovs Team: Ivanov, Ivanov, Ivanov
3	Team a aa aaa	Team: a, aa, aaa

Задача №634 Кубок CBOSS

(Время: 1 сек. Память: 16 Мб Сложность: 52%)

В 2239 году команде-победителю Открытого кубка CBOSS (<http://shade.msu.ru/~ejudge>) достался весьма нетрадиционный приз - поездка по k самым красивым городам России. Так как в России красивых городов достаточно много, то победителям было предложено выбрать k городов из списка, содержащего n городов.

Для удобства занумеруем эти города целыми числами от 1 до n. Для каждого города известно t_i - время, требующееся на осмотр его достопримечательностей. Также для каждой пары (i, j) , $1 \leq i, j \leq n$ известно $a_{i,j}$ - время, которое требуется на проезд из i-ого города в j-ый. При этом может оказаться, что $a_{i,j} \neq a_{j,i}$, но $a_{i,i}$ всегда равно нулю.

У студентов, входящих в команду-победитель, не так много времени на посещение красивых городов, ведь скоро у них сессия. Поэтому они хотят выбрать k городов и посетить их в таком порядке, чтобы затраты времени были минимальны. Разумеется, посещать один город несколько раз им неинтересно. Также они не хотят приезжать в город, не осматривая при этом его достопримечательности.

Напишите программу, находящую нужные k городов и порядок, в котором их нужно посетить.

Входные данные

Первая строка входного файла INPUT.TXT содержит целые числа n и k ($1 \leq n \leq 7$, $1 \leq k \leq n$). Каждая из последующих n строк входного файла содержит по n целых чисел каждая: j-ое число (i + 1)-ой строки входного файла - это время, требуемое на проезд из i-ого города в j-ый ($a_{i,j}$). Последняя строка входного файла содержит n целых чисел t_1, \dots, t_n .

Все числа во входном файле не превосходят 100. Все времена неотрицательны.

Выходные данные

В первой строке выходного файла OUTPUT.TXT выведите минимальное время, которое потребуется для посещения k городов с учетом осмотра достопримечательностей. Во второй строке выходного файла выведите номера городов в порядке посещения, гарантирующем такое время.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	4 3 0 3 2 1 8 0 6 5 1 2 0 4 5 6 7 0 1 2 3 4	10 3 1 4
2	4 4 0 3 2 1 8 0 6 5 1 2 0 4 5 6 7 0 1 2 3 4	18 3 1 4 2

Задача №635 Интернет-олимпиады

(Время: 1 сек. Память: 16 Мб Сложность: 28%)

Многим известно о проекте «Интернет-олимпиады по информатике», расположенном в сети Интернет по адресу <http://neerc.ifmo.ru/school/io/>, где проводятся онлайн-олимпиады для школьников в двух номинациях: базовой и усложненной. Ваша задача состоит в том, чтобы написать программу, обрабатывающую результаты Интернет-олимпиады, то есть определяющую: какие команды в какой номинации будут участвовать в следующей олимпиаде.

Правила перехода команд из одной номинации в другую таковы. Задачи усложненной номинации решают следующие команды:

- участвовавшие в предыдущей олимпиаде в усложненной номинации и решившие хотя бы одну задачу;
- участвовавшие в предыдущей олимпиаде в базовой номинации, решившие хотя бы одну задачу, и при этом решившие либо столько же задач, сколько команда-победитель, либо строго больше задач, чем команда, занявшая медианное место среди команд, решивших хотя бы одну задачу (место с номером $k/2$, где k - число команд, участвовавших в базовой номинации, решивших хотя бы одну задачу, округление производится вниз).

Все остальные команды участвуют в следующей олимпиаде в базовой номинации. В этой задаче мы считаем, что никакие новые команды не регистрируются для участия в следующей олимпиаде.

Напомним также, что при подведении итогов одной олимпиады команды сортируются по убыванию количества решенных задач, а при равенстве количества решенных задач - по возрастанию штрафного времени.

Входные данные

Первая строка входного файла INPUT.TXT содержит два целых числа: n и m - соответственно, количество команд, участвовавших в базовой и усложненной номинациях ($1 \leq n, m \leq 1000$).

После этого идут n строк, описывающих результаты команд, участвовавших в базовой

номинации. Каждая из этих строк содержит три целых числа, разделенных пробелами, - id, s, t - соответственно, уникальный идентификатор команды, количество решенных задач и штрафное время. Количество решенных задач - целое число от 0 до 12, а штрафное время - целое число от 0 до 20000. Идентификатор команды - это целое число от 1 до 2000.

После этого идут m строк, описывающих результаты усложненной номинации в таком же формате.

Выходные данные

В первой строке выходного файла OUTPUT.TXT выведите число k команд, которые допускаются до участия в усложненной номинации в следующей олимпиаде. Вторая строка должна содержать k целых чисел - идентификаторы этих команд в возрастающем порядке.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	6 3 1 1 45 2 4 678 3 5 1000 4 5 894 5 2 343 6 3 555 1998 0 0 1999 1 34 2000 3 366	4 3 4 1999 2000

Задача №636 Java Challenge

(Время: 2 сек. Память: 16 Мб Сложность: 47%)

Все участники олимпиад знают, что во время соревнования на счету каждая секунда. Иногда даже время, которое в суете затрачивается на переключение между окнами может оказаться критичным. В таких соревнованиях, как Java Challenge, количество окон может быть довольно большим (Java Challenge - это соревнование, проходящее в рамках финала чемпионата мира по программированию среди студентов. Оно состоит в разработке искусственного интеллекта для управления виртуальным роботом).

В данной задаче мы будем считать, что этот процесс выполняется следующим способом. В системе хранится циклический список открытых окон. При нажатии определенной комбинации клавиш k раз можно перейти в этом списке на k позиций в одну или в другую сторону. Кроме того, окна, относящиеся к каждому приложению так же организованы в циклический список. По этому списку также можно перемещаться в обе стороны, для перемещения на k позиций так же требуется k нажатий клавиш. При этом после своей активизации окно перемещаются в позицию перед первым элементом общего списка окон. Напишите программу, которая для каждого из окон будет определять минимальное количество нажатий клавиш, которое нужно затратить для его активизации. До начала выполнения операции активным является первое окно.

Входные данные

Первая строка входного файла INPUT.TXT содержит целое число n ($1 \leq n \leq 50000$) - количество открытых окон. Следующие n строк описывают окна в том порядке, в котором они идут в списке. Для каждого из окон задается номер приложения, которому соответствует это окно, и его

номер в циклическом списке окон этого приложения.

Выходные данные

На единственной строке выходного файла OUTPUT.TXT для каждого окна выведите минимальное количество нажатий клавиш, которое надо затратить для его активации.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	9	0 1 2 2 2 1 3 2 1
	3 2	
	2 2	
	2 3	
	1 3	
	2 1	
	3 1	
	4 1	
	1 2	
	1 1	

Задача №637 NEERC

(Время: 1 сек. Память: 16 Мб Сложность: 17%)

В полуфинале студенческого чемпионата мира по программированию NEERC (<http://neerc.ifmo.ru>) участвуют команды из n институтов. Участники для проведения соревнований распределяются по k залам, каждый из которых имеет размеры, достачные для размещения всех команд от всех институтов. При этом по правилам соревнований в одном зале может находиться не более одной команды от института.

Многие институты уже подали заявки на участие в полуфинале. Оргкомитет полуфинала хочет допустить до участия максимально возможное количество команд. При этом, разумеется, должна существовать возможность рассадить их по залам без нарушения правил.

Напишите программу, определяющую максимальное количество команд, которые можно допустить до участия в полуфинале.

Входные данные

Первая строка входного файла INPUT.TXT содержит число n - число институтов, подавших заявки. Вторая строка входного файла содержит n чисел a_1, \dots, a_n (a_i - это количество команд, заявленных от института номер i). Последняя строка входного файла содержит число k - количество залов, в которых проходят соревнования.

Все числа во входном файле целые, положительные и не превосходят 10000.

Выходные данные

В выходной файл OUTPUT.TXT выведите одно целое число - ответ на задачу.

Примеры

№	INPUT.TXT	OUTPUT.TXT

1	3 1 2 4 3	6
2	3 1 2 4 4	7

Задача №638

Всероссийская олимпиада по информатике

(Время: 1 сек. Память: 16 Мб Сложность: 37%)

2127 год. Прошло уже много лет с тех пор, как состоялась первая Всероссийская олимпиада по информатике. Как и многие другие соревнования, наши олимпиады теперь проводятся в несколько дней. Теперь даже задача выбора подходящего времени для олимпиады представляет определенные трудности. Ведь на разных планетах, входящих в состав Российской Федерации используются разные способы отсчета времени: длина месяца, количество дней в неделе и те дни, по которым невозможно проведение олимпиады, могут различаться. Возникла необходимость написания программы, которая поможет решить эту задачу. И тогда в жюри вспомнят, что уже сейчас мы предвидели такую ситуацию и предложили вам решить подобную задачу.

В качестве первого шага найдите количество способов выбрать время проведения олимпиады.

Входные данные

В первой строке входного файла INPUT.TXT содержатся два целых числа n и l ($1 \leq l \leq n \leq 100000$) - количество дней месяца и продолжительность олимпиады соответственно. Во второй строке задаются количество дней в неделе w , количество дней, запрещенных еженедельно, d_w и день недели, на который приходится первый день месяца s ($1 \leq s \leq w \leq n$, $0 \leq d_w \leq w$). Третья строка содержит d_w номеров дней недели (например, выходных), в которые нельзя проводить олимпиаду. В четвертой строке записано количество дней месяца d_m , не подходящих для проведения олимпиады по причинам отличным от еженедельного распорядка (например, такими днями являются государственные праздники). Последняя строка содержит d_m целых чисел - номера этих дней. Дни месяца так же нумеруются начиная с 1. Заметим, что некоторые дни могут быть запрещенными сразу по обеим причинам.

Выходные данные

В выходной файл OUTPUT.TXT выведите единственное целое число - количество способов выбрать l подряд идущих дней, в которые возможно проведение олимпиады.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	31 3 7 1 7 7 2 1 9	15

Некоторые из вас, наверное, слышали о сайте <http://www.topcoder.com>, на котором часто проводятся различные соревнования по программированию.

В некоторых из них участникам предлагаются три задачи, каждая из которых оценивается в некоторое количество баллов. В зависимости от того, насколько долго участник решал задачу, количество полученных им за нее баллов уменьшается. Как и в большинстве других соревнований, выигрывает участник, набравший наибольшее число баллов. Участники, набравшие одинаковое число баллов, считаются выступившими одинаково и их порядок в таблице итоговых результатов не важен. Из-за некоторых особенностей этих соревнований для предотвращения жульничества участники разделены в группы по 20 человек, называемые комнатами.

Ваша задача заключается в том, чтобы написать программу, которая по итоговым результатам в каждой комнате выводила бы суммарные итоговые результаты.

Входные данные

Первая строка входного файла INPUT.TXT содержит целое число n ($1 \leq n \leq 100$) - число комнат. Далее следуют n описаний итоговых результатов в комнатах.

Результаты в i -ой комнате заданы в следующем формате. Первая строка содержит целое число n_i ($1 \leq n_i \leq 20$)- количество участников в i -ой комнате. Следующие n_i строк содержат информацию о выступлениях участников. $j+1$ -ая строка описания результатов в i -ой комнате содержит информацию об участнике, занявшем в i -ой комнате j -ое место: разделенные одним пробелом вещественное число $total^i_j$ ($-5000 \leq total^i_j \leq 10000$) и строку $name^i_j$ - соответственно количество набранных участником баллов и его имя. Имя участника имеет длину от 1 до 25 и может содержать только буквы латинского алфавита, цифры и символ подчеркивания. При этом первый символ имени не является цифрой. Все вещественные числа заданы с двумя знаками после десятичной точки.

Гарантируется, что в каждой комнате участники упорядочены по невозрастанию набранных ими баллов.

Выходные данные

На первой строке выходного файла OUTPUT.TXT выведите N - суммарное число участников. На следующих N строках выведите информацию о выступлении участников. $(k+1)$ -ая строка описания суммарных результатов должна содержать информацию об участнике, занявшем k -ое место: разделенные одним пробелом вещественное число $total_k$ с двумя знаками после десятичной точки и строку $name_k$ - соответственно количество набранных участником баллов и его имя.

Не забудьте, что участники должны быть упорядочены по невозрастанию набранных ими баллов.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	2	11
	6	909.94 Savior
	909.94 Savior	867.15 Ying
	439.51 tywok	448.12 natori
	130.52 LimberG	439.51 tywok
	0.00 BryanChen	195.32 aubergineanode
	0.00 angsa	130.52 LimberG
	-75.00 The_Hedgehog	0.00 angsa

5	0.00 shalinmangar
867.15 Ying	0.00 BryanChen
448.12 natori	-25.00 Excilus
195.32 aubergineanode	-75.00 The_Hedgehog
0.00 shalinmangar	
-25.00 Excilus	

Задача №640

Test-The-Best

(Время: 1 сек. Память: 16 Мб Сложность: 50%)

Конкурс Test-the-best (<http://www.test-the-best.by>), в котором участвуют лучшие программисты из Беларуси, России и других стран, проводит очный тур. Стараясь не отставать от времени, организаторы решили позаботиться о безопасности. В последнее время на рынке техники есть много аппаратуры, позволяющей осуществлять видеонаблюдение.

Широко распространены программы, позволяющие автоматически обрабатывать полученные результаты. Однако имея дело с участниками этих соревнований, на существующие разработки полагаться небезопасно. Поэтому у оргкомитета возникла необходимость написания собственной системы анализа изображений. Перед вами поставлена задача сравнения двух черно-белых изображений на клетчатой сетке. Изображения считаются одинаковыми, если множества черных пикселей в них могут быть получены друг из друга поворотом на 90, 180, или 360 градусов и, возможно, симметрией относительно вертикальной оси.

Входные данные

Входной файл INPUT.TXT содержит описания двух изображений в следующем формате: первая строка содержит два целых числа n и m ($1 \leq n, m \leq 500$) - высоту и ширину изображения соответственно. Затем следуют n строк, содержащих по m символов: «#» обозначает черный пиксель, «.» - белый.

Выходные данные

В выходной файл OUTPUT.TXT выведите одно слово: «Yes», если изображения одинаковы и «No» в противном случае.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	<pre> 7 8###... ..#..... 6 10#..... ...#..... ...##..... </pre>	Yes

	
2	1 1 # 1 1 .	No

Задача №641

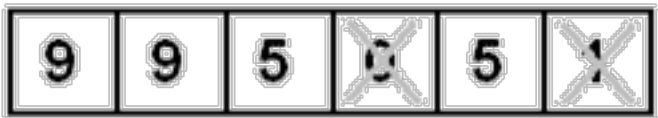
Странная лотерея

(Время: 1 сек. Память: 16 Мб Сложность: 25%)

В честь успешного окончания первой четверти родители Пети и Вовы подарили им один лотерейный билет. Как обычно, ребята всерьез не восприняли данный подарок ввиду довольно скудной вероятности выигрыша по их мнению. Но какова было удивление братьев, когда они узнали из средств массовой информации, что номер их билета является выигрышным.

После продолжительных минут радости ребята бросились узнавать сумму выигрыша, но как оказалось, узнать это не так уж просто. Дело в том, что сумма выигрыша отчасти определяется самим владельцем или владельцами (если их несколько) следующим образом. На обратной стороне билета есть определенная секция, стерев слой защитного покрытия, можно увидеть целое положительное число N. После чего, каждый из владельцев билета должен зачеркнуть ровно по одной цифре данного числа N, полученное число и является суммой выигрыша.

Ниже приведен пример с N равным 995051.



Максимальное число, которое может быть получено из данного N посредством вычеркивания двух цифр является 9955. Помогите Пете и Вове в этой непростой, но очень актуальной для них задачей.

Входные данные

В единственной строке входного файла INPUT.TXT находится число N ($100 \leq N < 10^{250}$). Число не содержит лидирующих нулей. Строка входного файла заканчивается переводом строки и не содержит символов отличных от "0"-"9".

Выходные данные

В выходной файл OUTPUT.TXT выведите одно число – максимальное значение, которое может быть получено из N посредством вычеркивания из него ровно двух цифр.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	102	2
2	995051	9955

Задача №642

Кризисный бизнес

(Время: 1 сек. Память: 16 Мб Сложность: 20%)

Петр Васильевич Колошин никогда не был пугливым человеком и всегда отличался спокойствием и прозорливостью, особенно в сфере мировых политических и экономических процессов. Однако, не смотря ни на что, Петр Васильевича очень недооценил последствия мирового финансового кризиса и как следствие был уволен пару недель назад с должности сетевого администратора одной большой и серьезной организации.

Не смотря ни на что, Петр Васильевич не отчаялся и решил начать свое дело. Тщательно проанализировав бизнес-климат в своем регионе, Петр Васильевич пришел к выводу, что наиболее целесообразным будет открыть новый таксопарк. Первое с чего решил начать новоиспеченный бизнесмен – это закупить автомобили. За все время работы Петр Васильевичу удалось накопить сумму S , которую он готов потратить на закупку машин.

В городе, в котором живет Петр Васильевич, есть только один автосалон. Известно, что в этом автосалоне выставлено на продажу N автомобилей, причем установлено, что стоимость i -го автомобиля равняется A_i . Вашей задачей является помочь еще неопытному бизнесмену Петру Васильевичу приобрести максимальное количество автомобилей, потратив сумму не более S .

Входные данные

В первой строке входного файла INPUT.TXT находится два целых положительных числа разделенные одиночным пробелом – это числа N ($1 \leq N \leq 100$) и S ($1 \leq S \leq 10^9$) соответственно.

Вторая строка содержит ровно N чисел A_i ($1 \leq A_i \leq 10^9$), которые описывают стоимость соответствующих автомобилей. Все числа в строке разделены одиночными пробелами.

Выходные данные

В выходной файл OUTPUT.TXT выведите одно целое число – максимальное количество автомобилей, которые сможет приобрести Петр Иванович на сумму не более чем S .

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	5 30 15 5 11 10 12	3
2	6 18 5 10 1 2 1 20	4

Задача №643 Временной ключ

(Время: 1 сек. Память: 16 Мб Сложность: 20%)

Практически каждый уважающий себя программист знает, что для полного успеха зачастую мало написать программный продукт, его также успешно надо уметь продать, и тем более также успешно защитить от взлома, а соответственно и от несанкционированного распространения.

Многие годы основным способом защиты программного обеспечения от незаконного распространения было использование, так называемого, активационного ключа. Вся проблема заключалась и заключается в том, что зачастую используется статический ключ, то есть активационный ключ для конкретного программного продукта не зависит ни от каких параметров и всегда является неизменным.

Знаменитая компания "Gold&Silver Soft" решилась на революционный шаг – было решено разработать принципиально новый способ динамической генерации активационного ключа. В

данном алгоритме ключ зависит от времени и меняется каждую минуту, что существенно затрудняет взлом.

Будем считать, что активационным ключом является обычное целое положительное число. В данной версии алгоритма значение ключа на следующей минуте целиком и полностью зависит от значения ключа в текущий момент. Если в данный момент ключ равен N , то через минуту он будет равен $N + S(N)$, где $S(N)$ – это число, называемое контрольной суммой числа N и равняется количеству единиц в двоичной записи числа N . То есть если $N = 6$, то в следующую минуту значение ключа будет равно 8, если быть точнее, то $N' = N + S(N) = 6 + S(6) = 6_{10} + 110_2 = 8$.

Будем считать, что на данный момент времени значение ключа равно N , вашей задачей является вычислить значение ключа через одну минуту.

Входные данные

В первой и единственной строке входного файла INPUT.TXT находится одно натуральное число – N ($1 \leq N \leq 2 \cdot 10^9$).

Выходные данные

В выходной файл OUTPUT.TXT выведите одно число – значение активационного ключа на следующей минуте, учитывая, что на данный момент времени значение ключа равно N .

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	2	3
2	6	8
3	27	31

Задача №644 Временной ключ-2

(Время: 1 сек. Память: 16 Мб Сложность: 80%)

Практически каждый уважающий себя программист знает, что для полного успеха зачастую мало написать программный продукт, его также успешно надо уметь продать, и тем более также успешно защитить от взлома, а соответственно и от несанкционированного распространения.

Многие годы основным способом защиты программного обеспечения от незаконного распространения было использование, так называемого, активационного ключа. Вся проблема заключалась и заключается в том, что зачастую используется статический ключ, то есть активационный ключ для конкретного программного продукта не зависит ни от каких параметров и всегда является неизменным.

Знаменитая компания "Gold&Silver Soft" решилась на революционный шаг – было решено разработать принципиально новый способ динамической генерации активационного ключа. В данном алгоритме ключ зависит от времени и меняется каждую минуту, что существенно затрудняет взлом.

Будем считать, что активационным ключом является обычное целое положительное число. В данной версии алгоритма значение ключа на следующей минуте целиком и полностью зависит от значения ключа в текущий момент. Если в данный момент ключ равен N , то через минуту он будет равен $N + S(N)$, где $S(N)$ – это число, называемое контрольной суммой числа N и равняется количеству единиц в двоичной записи числа N . То есть если $N = 6$, то в следующую минуту

значение ключа будет равно 8, если быть точнее, то $N' = N + S(N) = 6 + S(6) = 6_{10} + 110_2 = 8$.

Будем считать, что на данный момент времени значение ключа равно N , вашей задачей является вычислить значение ключа через K минут.

Входные данные

В первой и единственной строке входного файла INPUT.TXT находятся два натуральных числа – N ($1 \leq N \leq 2 \cdot 10^9$) и K ($1 \leq K \leq 2 \cdot 10^9$).

Выходные данные

В выходной файл OUTPUT.TXT выведите одно число – значение активационного ключа через K минут, учитывая, что на данный момент времени значение ключа равно N .

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	1 1	2
2	1 10	22
3	2 3	7

Задача №645 Красивая стена

(Время: 1 сек. Память: 16 Мб Сложность: 25%)

Однажды великий художник Гигабайт подарил королю Байтландии одно из своих лучших полотен. Король, впечатленный произведением Гигабайта, в знак благодарности подарил ему K плиток из очень ценной разновидности мрамора размером 10×10 квадратных сантиметров каждая.

Художник решил украсить этими плитками одну из стен своего дома. Он задумал выложить из них прямоугольник высотой N плиток и шириной W плиток. Художник понимает, что число вариантов для выбора N и W достаточно велико. Из всех возможных вариантов он хочет выбрать самый красивый. Ваша задача – помочь ему с выбором!

Для определения степени красоты художник решил учитывать два параметра:

1. Насколько выбранный прямоугольник будет близок к квадрату. Значение этого параметра равно модулю разности чисел N и W (т.е. $|N-W|$).
2. Сколько плиток останется невостребованными после украшения стены. Значение этого параметра равно разности чисел K и $N \cdot W$ (т.е. $K - N \cdot W$, где $K \geq N \cdot W$).

Степень красоты вычисляется как сумма значений двух описанных выше параметров. Например, имея 11 плиток, можно выбрать прямоугольник 3×3 , степень красоты равна $0+2 = 2$. Также можно выбрать прямоугольник 2×5 , тогда степень красоты равна $3+1 = 4$. Считается, что чем меньше степень красоты, тем красивее прямоугольник.

Ваша задача – написать программу, которая по заданному числу K находит размеры самого красивого прямоугольника.

Входные данные

Единственная строка входного файла INPUT.TXT содержит одно целое число K – количество подаренных королем плиток ($1 \leq K \leq 10^6$).

Выходные данные

Единственная строка выходного файла OUTPUT.TXT должна содержать натуральные числа N и W соответственно, которые определяют размеры самого красивого прямоугольника. Числа должны быть разделены одиночным пробелом. Если решений несколько, выведите любое из них.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	11	3 3
2	2	1 2
3	20	4 5

Задача №646 Сладкие забавы

(Время: 1 сек. Память: 16 Мб Сложность: 50%)

Маленькие сладкоежки Сережа и Юлия очень любят конфеты. Родители детей это знают, и потому у них не возникает вопросов о том, что ребятишкам дарить в день рождения.

А тут подвернулся особый случай – обоим ребятишкам исполняется по 10 лет, и потому родители подсказали приглашенным гостям, чему дети больше всего обрадуются на их общем юбилее.

Гостей в день рождения собралось достаточно много – N человек. Гость, пришедший i -м по счету, подарил детям коробку, содержащую A_i конфет. Коробки конфет были как совсем маленькие, так и невероятно большие. Коробки были с прозрачными крышками и было видно, сколько там конфет.

Чтобы все было честно, дети решили поделить коробки так, чтобы каждому досталось не менее K конфет. Однако дети обнаружили, что сделать это можно многими способами. Ваша задача – определить количество различных способов честного дележа конфет, учитывая, что коробки не вскрываются и конфеты поштучно не делятся, и каждая коробка должна достаться только одному из ребятишек. Два варианта деления конфет считаются различными, если существует коробка конфет, которая в данных вариантах принадлежит разным детям.

Входные данные

Первая строка входного файла INPUT.TXT содержит два натуральных числа N и K соответственно ($1 \leq N \leq 50$; $1 \leq K \leq 10000$). Числа в строке разделены одиночным пробелом.

Вторая строка содержит N натуральных чисел A_i ($1 \leq i \leq N$, $1 \leq A_i \leq 10^9$), разделенных одиночными пробелами, где число A_i – это количество конфет в коробке, подаренной гостем, пришедшим i -м по счету.

Выходные данные

Единственная строка выходного файла OUTPUT.TXT должна содержать одно целое число — количество способов честно поделить конфеты между детьми.

Примеры

№	INPUT.TXT	OUTPUT.TXT

1	4 3 1 2 3 2	8
2	5 67 10 22 30 41 50	6
3	3 6 4 1 5	0

Пояснение

Для первого примера варианты разбиения (номера коробок, доставшихся, например, Сереже): 3, 1 2, 1 3, 1 4, 2 3, 2 4, 3 4, 1 2 4.

Задача №647 Адаптивный поиск

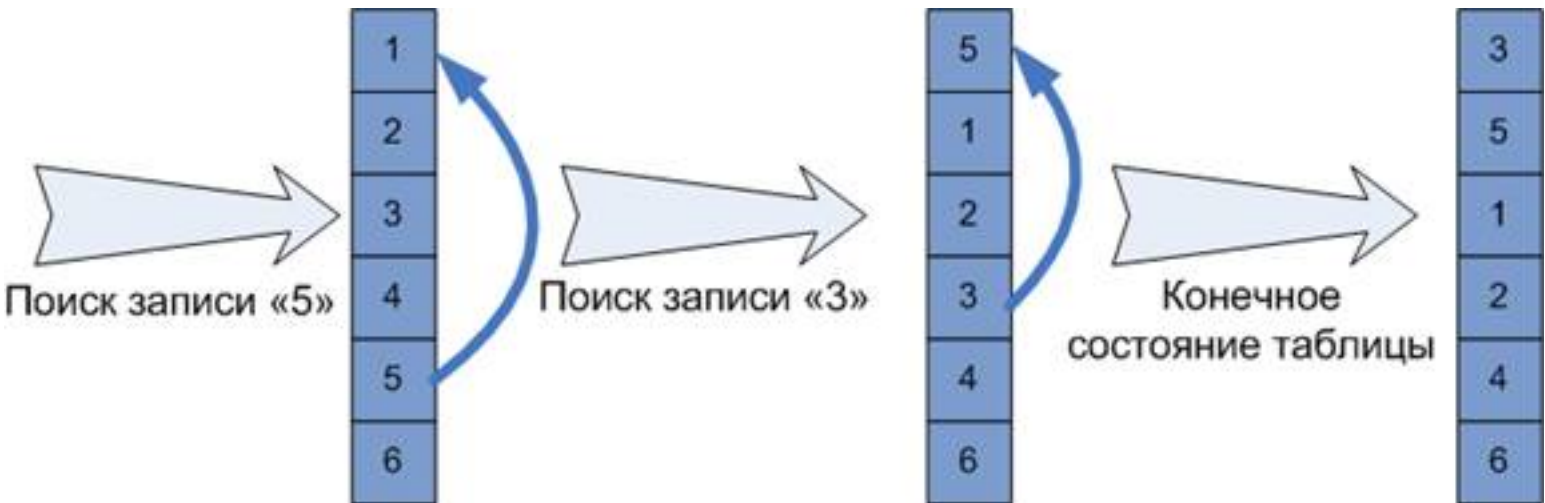
(Время: 2 сек. Память: 16 Мб Сложность: 80%)

Знаменитая компания "Gold&Silver Soft" решила занять ведущее место в области разработки реляционных баз данных. Руководство компании понимает, что для этого необходимо удивить потребителей быстрodeйствием своего программного продукта.

Ни для кого не секрет, что в основе реляционной базы данных лежит таблица, которую можно рассматривать как одномерный массив записей. Известно, что при поиске все записи таблицы просматриваются последовательно, начиная с самой первой и заканчивая найденной.

Технический отдел компании установил, что часто бывает так, что поиск одной и той же записи в таблице производится несколько раз. Основываясь на этом, программисты решили после каждого нового поискового запроса менять порядок следования записей в таблице. Другими словами, после поиска найденная запись перемещается на первое место в таблице. Очевидно, что чем чаще осуществляется поиск определенной записи, тем ближе она будет к началу таблицы и тем быстрее будет поиск этой записи.

Вашей задачей является написать программу, которая для каждого из М последовательно заданных поисковых запросов будет определять количество просмотренных записей при поиске заданной. Для простоты обозначения будем считать, что имеется таблица с N записями, где запись – это число от 1 до N. В начале все записи в таблице расположены в порядке возрастания, то есть на i-м месте в таблице находится число i. Для приведенного ниже примера при M = 2, N = 6 и запросах на поиск чисел «5» и «3» потребуется 5 и 4 просмотра записей соответственно.



Входные данные

Первая строка входного файла INPUT.TXT содержит два целых числа N и M (1 ≤ N, M ≤ 65535) — количество записей в таблице и количество запросов на поиск соответственно. Числа разделены одиночным пробелом.

Вторая строка содержит M натуральных чисел A_i ($1 \leq A_i \leq N$), разделенных одиночными пробелами, где A_i — запрос на поиск числа A_i в таблице. Запросы на поиск выполняются последовательно в порядке их ввода.

Выходные данные

Единственная строка выходного файла OUTPUT.TXT должна содержать M натуральных чисел, разделенных одиночными пробелами, i -е число — это количество просмотренных записей при поиске числа A_i .

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	6 2 5 3	5 4
2	10 10 10 9 8 7 6 5 4 3 2 1	10 10 10 10 10 10 10 10 10 10
3	3 14 3 2 3 3 1 1 2 2 1 1 1 1 2 1	3 3 2 1 3 1 3 1 2 1 1 1 2 2

Задача №648 Азартный Шрэк

(Время: 1 сек. Память: 16 Мб Сложность: 25%)

Как-то раз Шрэк решил посетить казино. Не будучи заядлым любителем азартных игр, Шрэк обнаружил, что он не знает правил ни одной из игр, доступных в казино. Недолго думая, Шрэк решил все-таки поиграть. Его взор привлекла игра с довольно незамысловатыми правилами.

На игровом столе лежат N карточек. На каждой карточке написано целое положительное число. Игра проходит между игроком и крупье. Карточки лежат на столе числами вниз. Игра заключается в том, что игрок открывает ровно $N/2$ карточек. Сумма всех чисел, написанных на карточках открытых игроком, называется “суммой игрока”. Следующим ходом крупье открывает оставшиеся $N/2$ карточек. Сумма всех чисел, написанных на карточках открытых крупье, называется “суммой крупье”. Выигрыш игрока определяется разностью чисел между “суммой игрока” и “суммой крупье”. Очевидно, что полученная разность может быть отрицательным числом. Это свидетельствует о том, что игрок проиграл и должен казино соответствующую сумму.

Все бы ничего, но Шрэк обладает способностью видеть надписи сквозь бумагу любой плотности. Ваша задача определить максимальную сумму выигрыша, которую может получить Шрэк с учетом того, что он видит все числа, написанные на карточках.

Входные данные

Первая строка входного файла INPUT.TXT содержит одно четное натуральное число N ($2 \leq N \leq 100$). Вторая строка входного файла содержит ровно N чисел A_i ($1 \leq A_i \leq 10^6$) — числа, написанные на игровых карточках. Все числа в строке разделяются одиночными пробелами, A_i — число, написанное на i -й карточке. Карточки нумеруются последовательно, начиная с единицы.

Выходные данные

Единственная строка выходного файла OUTPUT.TXT должна содержать ровно одно целое число — максимальный выигрыш, который может получить Шрэк с учетом своей уникальной

способности видеть числа, написанные на карточках.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	2 1 3	2
2	4 3 1 8 100	104

Задача №649 Защищенный пароль

(Время: 2 сек. Память: 16 Мб Сложность: 50%)

Очень надежная и совершенно бесплатная операционная система «Vokna» известна своей безопасностью, так как при проектировании разработчики уделили большое внимание проблемам генерации паролей. Ядро операционной системы содержит в себе строку S длиной N символов. Генерация пароля происходит с использованием символов строки S . Паролем будем называть подстроку $S_{i,j}$ строки S длиной не менее одного и не более N символов. Подстрокой $S_{i,j}$ строки S называется строка, последовательно составленная из символов $S[i], S[i+1], S[i+2], \dots, S[j-1], S[j]$. Символы в строке нумеруются последовательно начиная с единицы. Пароль $S_{i,j}$ считается защищенным, если в нем встречается не более K одинаковых символов. Вашей задачей является по заданной строке S и числу K определить количество различных вариантов выбора защищенного пароля. Два варианта выбора пароля $S^1_{i,j}$ и $S^2_{i',j'}$ называются различными, если $i \neq i'$ или $j \neq j'$.

Входные данные

Первая строка входного файла INPUT.TXT содержит два натуральных числа N ($1 \leq N \leq 10^6$) и K ($1 \leq K \leq N$), разделенных одиночным пробелом, где N – количество символов в строке S ; K – максимальное количество одинаковых символов в пароле. Вторая строка входного файла содержит ровно N символов. Каждый символ является либо маленькой латинской буквой, либо цифрой. Каждая строка входного файла заканчивается символом перевода строки.

Выходные данные

Единственная строка выходного файла OUTPUT.TXT должна содержать одно целое число – количество вариантов выбора защищенного пароля.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	6 2 7aaarr	15
2	4 1 ayay	7

Задача №650 Странная сеть

(Время: 1 сек. Память: 16 Мб Сложность: 65%)

В одной из самых крупных IT-компаний уже много лет действует локальная сеть, в которую

входят N компьютеров, последовательно пронумерованных целыми числами от 1 до N . Для соединения компьютеров используется провод специального типа. С помощью одного провода можно напрямую соединить ровно два компьютера, причем соединение позволяет передавать данные в двух направлениях. Каждый компьютер имеет два порта, поэтому он может быть соединен напрямую не более чем с двумя другими компьютерами. Сеть организована таким образом, что если два компьютера не соединены напрямую проводом, то передача данных может осуществляться через промежуточные компьютеры. *Путем передачи данных* между компьютерами с номерами V_1 и V_K называется такая последовательность компьютеров $V_1, V_2, \dots, V_{K-1}, V_K$, что для любого $2 \leq i \leq K$ существует прямое соединение между компьютерами V_{i-1} и V_i . *Длиной пути передачи данных* считается количество используемых промежуточных компьютеров. *Минимальным путем передачи данных* между компьютерами V_i и V_j является такой путь передачи данных, длина которого минимальна из возможных.

Для приведенного ниже примера длина минимального пути передачи данных между компьютерами 1 и 8 равна 2, а между компьютерами 2 и 7 равна 1, между компьютерами 2 и 4 равна 0.



Если не существует ни одного пути передачи данных между какими-либо двумя компьютерами, то в данном случае длина минимального пути передачи данных между этими компьютерами считается равной нулю. Для приведенного выше примера между компьютерами 1 и 4 не существует пути передачи данных, поэтому длина минимального пути передачи данных равна 0.

Для выяснения эффективности локальной сети необходимо определить *коэффициент связности*, значение которого равно сумме длин минимальных путей передачи данных между всеми парами компьютеров. Пары, отличающиеся порядком следования элементов считаются различными, то есть пара (1, 2) и (2, 1) считаются различными.

Вашей задачей является определить значение коэффициента связности для заданной локальной сети.

Входные данные

Первая строка входного файла INPUT.TXT содержит два целых числа N и M – число компьютеров и число прямых соединений соответственно ($2 \leq N \leq 10^5$, $1 \leq M \leq 10^6$). Числа разделены одиночным пробелом.

Каждая из следующих M строк содержит описание одного прямого соединения. Каждое прямое соединение описывается при помощи двух целых чисел X_i и Y_i ($1 \leq X_i, Y_i \leq N$; $X_i \neq Y_i$), разделенных пробелом – номера соответствующих компьютеров, соединенных напрямую. Известно, что $X_i \neq X_j$ или $Y_i \neq Y_j$, если $i \neq j$. Строки в файле нумеруются последовательно, начиная с единицы.

Выходные данные

Единственная строка выходного файла OUTPUT.TXT должна содержать одно целое число – коэффициент связности локальной сети.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	2 1 1 2	0
2	4 4 1 2 2 3 4 3 1 4	4
3	8 7 1 3 3 6 4 2 7 4 2 5 5 7 6 8	12

Задача №651 Преобразование моноклеточных

(Время: 1 сек. Память: 16 Мб Сложность: 20%)

Как великолепна страна Байтландия! В ней есть цветущие леса, прозрачные реки, кисельные берега... Но речь пойдет не о них. Уже много лет в Байтландии функционирует НИИ “Цитологии и генетики”. В нем выводятся новые формы жизни. Недавно ученым этого НИИ удалось разработать принципиально новый вид организмов. Особенностью этих организмов является то, что они состоят из большого количества однотипных клеток, то есть являются моноклеточными.

Правительство Байтландии заинтересовалось новой разработкой и сделало заказ на производство двух моноклеточных организмов, в каждом из которых должно быть по М клеток. За несколько дней до сдачи проекта было обнаружено, что в одном из организмов получается не М клеток, а N. На какой из стадий разработки была допущена ошибка неизвестно, но положение надо исправлять!

Сотрудниками НИИ было принято решение о преобразовании моноклеточного с N клетками в моноклеточное с М клетками. Для этого в экстренном режиме было разработано два типа вещества:

1. Вещество, которое делит клетки моноклеточного организма, т.е. каждая клетка делится на Р частей. В результате количество клеток умножается на Р, где Р – простое число.
2. Вещество, объединяющее клетки. Клетки организма объединяются в группы по Т штук, где Т также простое число. Далее каждая группа клеток объединяется в одну клетку. В результате общее количество клеток делится на Т. При этом Т выбирается таким, чтобы деление происходило без остатка.

Отметим, что натуральное число называется простым, если оно имеет только два натуральных делителя – это единица и само число.

Серьезным недостатком этих веществ является их высокая стоимость. В соответствии с этим

требуется преобразовать моноклеточное с М клетками в моноклеточное с N клетками за минимальное количество операций. За одну операцию к моноклеточному можно применить одно вещество из двух заданных типов. Помогите НИИ “Цитологии и генетики” разрешить эту непростую задачу!

Входные данные

В первой строке входного файла INPUT.TXT заданы два натуральных числа N и M ($1 \leq N, M \leq 10^9$) разделенные одиночным пробелом.

Выходные данные

Единственная строка выходного файла OUTPUT.TXT должна содержать целое число – минимальное количество операций, необходимое для преобразования моноклеточного организма с N клетками в моноклеточный организм с M клетками.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	2 36	3
2	32768 3	16
3	1434 1434	0

Задача №652 Квантовый имитатор

(Время: 2 сек. Память: 16 Мб Сложность: 70%)

Знаменитая компания по разработке аппаратного обеспечения «Zhaleza» после многолетних исследований разработала опытный образец квантового компьютера. Разработанная архитектура концептуально отличалась от используемой микропроцессорной архитектуры. Будущей областью применения своего изобретения компания видит обработку и хранение огромных объемов информации. Основным объектом манипулирования квантового компьютера является одномерный массив. На данный момент в процессоре квантового компьютера реализованы операции двух типов:

1. *Инвертирование части массива, начиная с индекса L и заканчивая индексом R.* Под инвертированием части массива понимается изменение прямого порядка следования элементов массива на обратный, начиная с индекса L и заканчивая индексом R, то есть элемент L меняется местами с элементом R, элемент L+1 с элементом R-1 и так далее. На псевдоязыке эту операцию можно записать следующим образом:

```
For i = 1 to [(R-L+1)/2] do Swap(A, L+i-1, R-i+1)
```

Обозначение $[X]$ – это наибольшее целое число, не превосходящее X, где X –действительное число. Процедура Swap(A, i, j) может быть записана следующим образом:

```
Procedure Swap(A, i, j)
Begin
  Tmp = A[i]
  A[i] = A[j]
  A[j] = Tmp
End
```

2. *Вывод на экран суммы элементов массива, начиная с индекса L и заканчивая индексом R.*

На псевдоязыке процедура может быть реализована следующим образом:

```

Procedure PrintSum(A, L, R)
Begin
    Sum = 0
    For i = L to R do Sum = Sum + A[i]
    Print(Sum)
End
```

Массив, с которыми работает квантовый компьютер, являются одномерным. Все элементы являются натуральными числами и нумеруются последовательно, начиная с единицы. Первоначально массив из N элементов заполнен числами от 1 до N, таким образом что $A[1] = 1$, $A[2] = 2, \dots, A[N] = N$.

Программой будем называть последовательность команд длины K, где каждая команда – это операция одного из двух типов. Ваша задача разработать имитатор работы квантового компьютера, позволяющий по заданной размерности массива N и заданной программе определить последовательность чисел выведенных квантовым компьютером на экран.

Входные данные

Первая строка входного файла INPUT.TXT содержит два целых числа N и K ($1 \leq N \leq 10^9, 1 \leq K \leq 5000$). Каждая последующая строка описывает ровно одну операцию. Первый символ строки входного файла характеризует тип операции: 'T'(ASCII 73) – первый тип запроса, 'S'(ASCII 83) – второй тип запроса. Далее в строке через пробел следуют два числа, характеризующие соответствующие индексы L и R ($1 \leq L \leq R \leq N$), разделенные одиночным пробелом.

Выходные данные

Выходной файл OUTPUT.TXT должен содержать последовательность чисел выведенных на экран квантовым компьютером. Каждое число выводится в отдельную строку. Гарантируется, что в программе присутствует операция второго типа.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	10 2 I 1 5 S 3 7	19
2	15 4 S 2 11 I 10 15 I 1 10 S 5 10	65 21
3	1000 10 S 17 149 I 199 428 I 17 417 I 212 987 S 300 420 I 400 700 I 633 759 S 11 238 S 477 872	11039 101519 86244 194331

Задача №653

Аттракцион

(Время: 1 сек. Память: 16 Мб Сложность: 50%)

На протяжении многих лет в Байтландии существует парк развлечений “Funny byte”, в котором представлено много различных аттракционов: колесо вычислений, веселые горки с трассами в форме интегралов, равнобедренная ромашка и многие другие.

В последние годы популярность “Funny byte” стала неуклонно падать. В целях привлечения посетителей руководство парка решило открыть новый аттракцион, который представляет собой сложный механизм.

Кресла аттракциона расположены в N рядов по M кресел в каждом. То есть каждое кресло характеризуется номером ряда и номером колонки, в котором оно находится. Ряды нумеруются последовательно сверху вниз начиная с единицы, колонки нумеруются слева направо начиная с единицы. Каждое кресло имеет свой уникальный номер. Кресло, находящееся в i -м ряду и в j -ой колонке, имеет номер $(i-1)*M + j$.

После посадки отдыхающих, кресла поднимают вверх над землей. И начинается веселье! Механизм аттракциона случайным образом производит некоторое количество операций. Под одной операцией понимается взаимная перестановка двух рядов либо двух колонок. При взаимной перестановке двух рядов или колонок каждое кресло в ряду или колонке заменится на соответствующее ему кресло в другом ряду или колонке.

И вот аттракцион завершил свою работу. Но есть одна трудность! Кресла надо вернуть в начальное положение при помощи таких же операций. Как количество, так и сами операции не обязательно должны быть идентичны тем, которые производил аттракцион во время сеанса. Руководство парка решило, что вернуть кресла в начальное положение необходимо не более чем за 1000 операций.

Такая задача оказалась не по силам разработчикам механизма. Помогите им! От вас требуется разработать программу, позволяющую вернуть кресла в начальное положение. Гарантируется, что решение существует.

Входные данные

В первой строке входного файла INPUT.TXT заданы два натуральных числа N и M ($1 \leq N, M \leq 250$). В последующих N строках задано по M натуральных чисел, где j -е число в $i+1$ -й строке соответствует номеру кресла после окончания сеанса аттракциона. Числа в строках разделяются одиночными пробелами.

Выходные данные

В первой строке выходного файла OUTPUT.TXT должно быть выведено количество операций перестановки K , которое не должно превышать 1000. Каждая из следующих K строк описывает одну операцию. Каждая операция описывается строкой вида $Q \ X \ Y$, где Q – символ 'R'(ASCII 82) либо символ 'C'(ASCII 67). Если Q равно 'R', то данная операция является перестановкой рядов, если Q равно 'C', то операция является перестановкой колонок. X и Y – два натуральных числа, соответствующие номерам рядов (колонок), которые будут переставлены в результате данной операции. Операции должны быть выведены в порядке осуществления, то есть последовательное применение которых позволит вернуть кресла в начальное положение.

Примеры

№	INPUT.TXT	OUTPUT.TXT
---	-----------	------------

1	2 2 4 3 2 1	2 C 1 2 R 1 2
2	3 5 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15	0
3	4 5 10 7 9 8 6 15 12 14 13 11 20 17 19 18 16 5 2 4 3 1	5 R 1 4 C 1 5 C 3 4 R 2 4 R 3 4

Задача №654

Концертный зал

(Время: 1 сек. Память: 16 Мб Сложность: 60%)

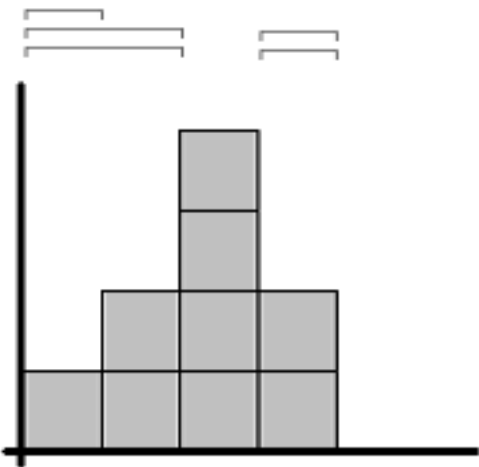
Большой концертный зал Байтланда — известное на весь мир место, где мечтают выступать величайшие оперные певцы и певицы. В желающих послушать их выступления, как правило, недостатка нет. Поэтому Министерством культуры Байтланда было принято решение увеличить размеры зала.

Однако это вызвало другую проблему: на задних рядах обновленного концертного зала посетители практически ничего не слышат. Поэтому Инженерный институт предложил проект акустической системы, которая будет состоять из микрофонов, записывающих происходящее на сцене, и динамиков, транслирующих усиленный звук в зал.

В идеальном случае динамики должны достоверно воспроизводить звук, записываемый с микрофонов. На практике этого добиться почти невозможно, так как при текущем уровне развития технологий практически все динамики воспроизводят различные частоты с различной громкостью. Мириться с этим инженеры, однако, не собираются.

В лаборатории удалось измерить АЧХ (амплитудно-частотную характеристику) динамиков и представить ее в следующей форме:

1. весь диапазон частот, воспроизводимых динамиками, разделен на N последовательных интервалов, нумеруемых от 1 до N;
2. в i-м интервале известно A_i - значение усредненной по интервалу громкости в децибелах (Прим.: децибел (дБ) — единица измерения громкости).



АЧХ из примера №2 и усилители, которые нужны, чтобы ее «выровнять» на уровне в 4 дБ (5 штук, усилители применяются в порядке от верхних к нижним)

АЧХ динамиком можно править с помощью специальных электронных устройств — усилителей. Усилитель может поднять силу сигнала на всех интервалах с А-го по В-й на 1 дБ, где А, В — произвольные натуральные числа, не превосходящие N, $A \leq B$. Из-за особенностей применяемых в устройстве радиодеталей, сила сигнала перед применением усилителя должна быть одинакова на всех интервалах с А-го по В-ой. Так как стоимость усилителя достаточно велика, то их количество должно быть минимально.

Считается, что динамики воспроизводят звук достоверно, если на всех интервалах значение громкости одинаково. Напишите программу, которая вычислит минимальное число усилителей, необходимое для достижения достоверного звучания.

Входные данные

Первая строка входного файла INPUT.TXT содержит одно число N ($1 \leq N \leq 200000$) — количество интервалов, на которых замерялась АЧХ. Вторая строка содержит N натуральных чисел A_i ($1 \leq i \leq N, 1 \leq A_i \leq 10^9$), разделенных одиночными пробелами — усредненная громкость на i-м интервале в децибелах.

Выходные данные

Единственная строка выходного файла OUTPUT.TXT должна содержать одно целое число — минимальное число усилителей, с помощью которых можно добиться достоверного звучания.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	3 1 3 2	3
2	4 1 2 4 2	5
3	5 3 1 4 1 1	6

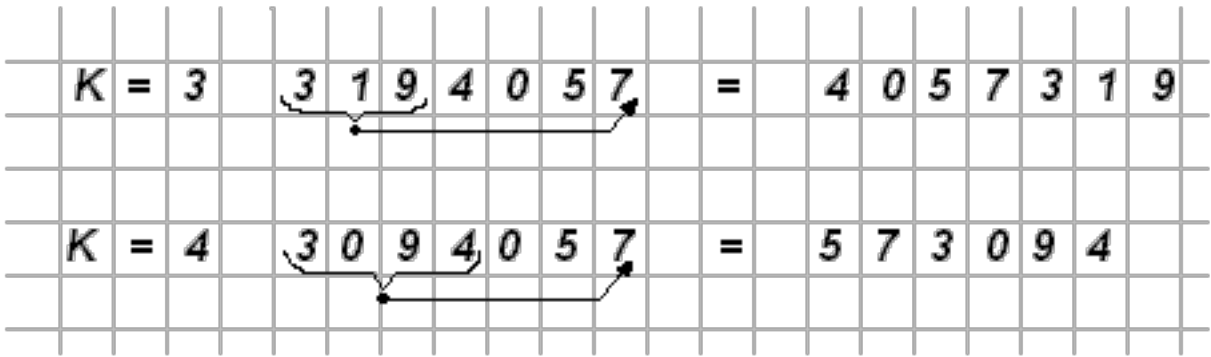
Задача №655 Кодирование данных

(Время: 1 сек. Память: 16 Мб Сложность: 50%)

В компании «Gold&Silver Soft» разработан новый алгоритм кодирования натуральных чисел. Этот алгоритм основан на операции циклического сдвига числа.

Циклическим сдвигом натурального числа N относительно числа K назовем операцию переноса K крайних слева цифр в десятичной записи числа N к цифре крайней справа. Обозначим через $cyclic(N, K)$ число, получаемое из N посредством циклического сдвига относительно числа K.

Введенное понятие можно проиллюстрировать на примерах:



В разработанном алгоритме предлагается для кодирования натурального числа N использовать число $code(N) = N + cyclic(N, K)$ (где K – известный параметр).

Вы работаете в «Gold&Silver Soft» и Вам была поручена разработка алгоритма декодирования, то есть восстановления числа N по числам $code(N)$ и K .

Входные данные

В первой строке входного файла INPUT.TXT находится целое число $code(N)$. Во второй строке входного файла находится целое число K . Ограничения: $10^{K+1} \leq code(N) \leq 10^{18}$, $1 \leq K \leq 17$.

Выходные данные

В первой и единственной строке выходного файла OUTPUT.TXT должно быть выведено целое число N . Если существует несколько вариантов восстановления числа N по числам $code(N)$ и K , то выведите любой из них.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	512 1	319

Задача №656 Дорожный аукцион

(Время: 1 сек. Память: 16 Мб Сложность: 75%)

Где-то далеко от нас, на краю земли, есть одна небольшая, но красивая страна WWW с богатейшим историческим прошлым. Люди, населяющие ее, известны всему миру своей добротой и гостеприимством. Вся территория страны условно поделена на районы. Каждый район состоит из определенного количества городов, один из которых является районным центром.

Некоторые пары городов данного государства соединены двусторонними дорогами, причем известно, что любой районный центр соединен дорогами со всеми остальными городами района, а также не более чем с двумя другими районными центрами. Никакая дорога не соединяет два города, не являющиеся районными центрами. Дорога может соединять два города из разных районов только в том случае, если оба они являются районными центрами. Между любой парой городов может быть не более одной дороги. Дороги построены таким образом, что по ним можно доехать из любого города в любой другой.

Исторически сложилось, что право на владение всеми дорогами до недавних пор принадлежали одной известной компании «АвтоДор». В связи с этим в конституционный суд был подан антимонопольный иск, который был удовлетворен – теперь компании предстоит продать часть своих владений. Экономисты компании определили для каждой дороги ее стоимость.

Одна маленькая, небогатая, но гордая фирма «КурсИнвест», в которой Вы работаете финансовым директором, захотела выкупить часть дорог, а именно, K из них. Причем необходимо, чтобы для любых двух городов, к которым примыкает хотя бы одна из выкупленных K дорог, существовало не менее одного соединяющего их пути, состоящего только из приобретенных дорог. Вам, как финансовому директору, было поручено найти экономически выгодное решение. Решение будем называть экономически выгодным, если денежная сумма, потраченная на приобретение дорог, является минимальной.

Входные данные

В первой строке входного файла INPUT.TXT находятся три целых числа N ($3 \leq N \leq 2000$), M и K ($1 \leq K < M \leq 10^5$), где N – общее количество городов, M – общее количество дорог, K – количество дорог, которое необходимо приобрести.

Далее следует M строк, в каждой из которых записаны три целых числа A_i , B_i и C_i , где A_i и B_i – номера городов, которые соединены дорогой ($1 \leq A_i, B_i \leq N$, $A_i \neq B_i$), а C_i – стоимость дороги ($1 \leq C_i \leq 10^6$). Все числа в строках разделены одиночными пробелами.

Выходные данные

Выходной файл OUTPUT.TXT должен состоят из K строк, каждая из которых должна содержать одно число – номер приобретенной дороги. Дороги нумеруются в порядке их ввода начиная с единицы. Если решений несколько, то выведите любое из них.

Пример

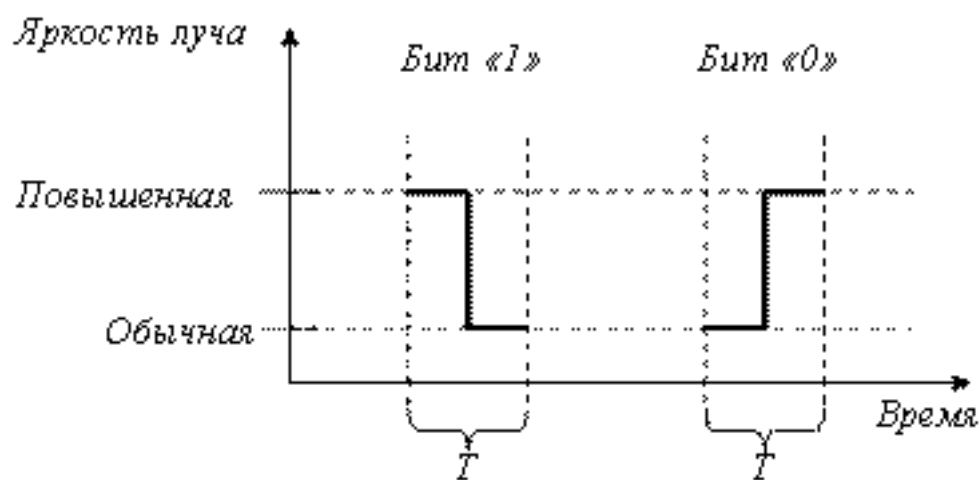
№	INPUT.TXT	OUTPUT.TXT
1	9 8 4	4
	8 9 2	1
	5 1 10	8
	3 8 11	3
	2 5 7	
	8 7 8	
	5 6 12	
	4 7 9	
	3 5 5	

Задача №657 Hello space

(Время: 1 сек. Память: 16 Мб Сложность: 60%)

Ученые из центра космических наблюдений зафиксировали полезный информационный сигнал, который поступает из далекой галактики. Оказалось, что информацию передают некоторые разумные существа, посылая в сторону Земли луч, яркость которого изменяется с обычной на повышенную, и наоборот. Если информация не передается, то яркость луча обычная и не меняется во времени (пустой сигнал).

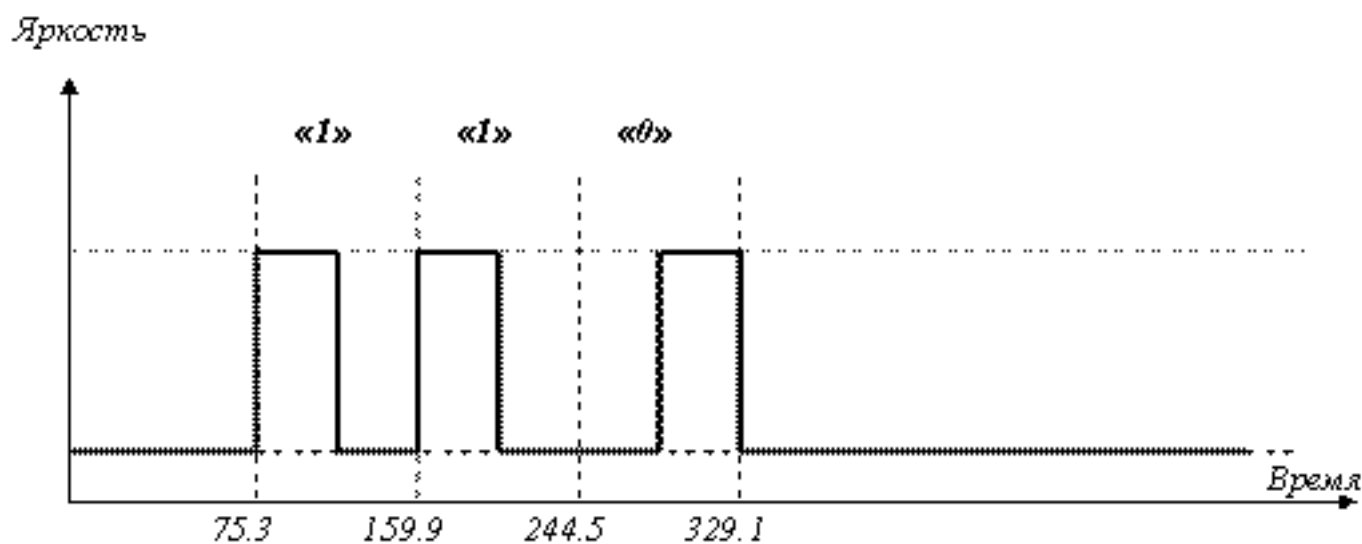
Ученым удалось распознать правило, по которому инопланетяне кодируют полезный сигнал. Он является двоичным кодом, причем в течение сеанса связи передача каждого бита сигнала занимает одно и то же вещественное время T . Для передачи бита “1” яркость луча в течение времени $T/2$ повышенная, а в течение второй половины времени $T/2$ – обычная. Передача бита “0” осуществляется наоборот: в течение первой половины времени $T/2$ яркость луча обычная, а в течение второй половины времени $T/2$ – повышенная. Ученые считают, что первый бит полезного сигнала всегда равен единице, а последний бит – нулю. Для наглядности пример передачи каждого из битов представлен на рисунке:



Ученым удалось сконструировать прибор, который позволяет представить информацию, поступающую из далекой галактики, в виде последовательности нечетной длины, состоящей из целых чисел, определяющих, сколько времени передающий луч находился в постоянном состоянии. Первое число этой последовательности показывает время, в течение которого луч находился в состоянии повышенной яркости, второе число – в состоянии обычной яркости, третье – снова повышенной яркости и т.д. Последнее число последовательности (как и первое) показывает время нахождения луча в состоянии повышенной яркости.

К сожалению, построенный прибор не является абсолютно точным – результаты всех измерений, которые он производит, имеют относительную погрешность, не превосходящую 10%. Это значит, что реальное время, в течение которого луч находился в постоянном состоянии (– вещественное число), и зафиксированное прибором время (– целое число) удовлетворяют неравенству $(L - L') / L \leq 0.1$.

В качестве примера рассмотрим передачу сигнала “110”, если $T = 84.6$.



При точной работе прибор зафиксировал бы следующую последовательность:

42.3, 42.3, 42.3, 84.6, 42.3

Обратите внимание, что данная последовательность не зависит от момента времени, в который началась передача сигнала.

Однако из-за наличия погрешности результат работы прибора достаточно непредсказуем. К примеру, полученная последовательность может быть следующего вида:

42, 43, 39, 93, 44

Ученым при помощи построенного прибора удалось получить информацию о К инопланетных сигналах. Теперь задача состоит в том, чтобы по полученной прибором информации определить сами сигналы. Эта задача осложняется тем, что ученые не знают, какое значение T инопланетяне использовали для передачи сигналов. Более того, для передачи разных сигналов могли использоваться разные значения T . Вам как начальнику отдела инновационных технологий центра космических наблюдений была поручена разработка программы для решения этой задачи.

Входные данные

В первой строке входного файла INPUT.TXT записано целое число K – количество принятых инопланетных сигналов ($1 \leq K \leq 5$). Далее следует описание K сигналов. Описание каждого сигнала начинается со строки, содержащей целое нечетное число N – длина последовательности, которую зафиксировал прибор для данного сигнала ($3 \leq N \leq 32767$). Далее следует N строк, в каждой из которых записано одно целое число T_i – значение i -го элемента последовательности ($0 < T_i \leq 10^9$).

Выходные данные

В выходной файл OUTPUT.TXT для каждого из сигналов необходимо вывести одну строку из N символов '0'/'1' – двоичный код принятого сигнала. Если существует несколько способов восстановления сигнала, то выведите любой из них. Если не существует ни одного способа восстановления сигнала, то выведите строку "ERROR".

Пример

№	INPUT.TXT	OUTPUT.TXT
1	2	1 1 0
	5	ERROR
	4 2	
	4 3	
	3 9	
	9 3	
	4 4	
	5	
	1 0	
	1 0	
	2 2	
	1 0	
	3 0	

Задача №658 Фермерское счастье

(Время: 1 сек. Память: 16 Мб Сложность: 65%)

На протяжении многих лет кролик Роджер известен всей округе как выдающийся фермер, в искусстве ведения хозяйства ему нет равных. В проведенном международном конкурсе «Фермер года», с большим отрывом выиграл кролик Роджер.

Главный приз – возможность первым из всех фермеров отгородить себе участок на специально выделенном поле. Известно, что на поле растет N деревьев. Отгораживаемый участок должен иметь форму треугольника, вершинами которого являются некоторые из растущих на поле деревьев.

Кролик быстро смекнул, что на самом деле существует много вариантов выбора участка, но его интересуют только такие варианты, при которых участок не содержит внутри себя других деревьев. Ваша задача состоит в том, чтобы помочь Роджеру определить количество интересующих его вариантов выбора участка.

Входные данные

В первой строке входного файла INPUT.TXT находится одно целое число N ($3 \leq N \leq 255$). В каждой из последующих N строк записаны два целых числа X_i и Y_i – координаты очередного дерева

в прямоугольной декартовой системе координат ($-10000 \leq X_i, Y_i \leq 10000$). Деревья достаточно малы, поэтому их можно считать точками на плоскости. Никакие три дерева не лежат на одной прямой. Числа в строках разделены одним пробелом.

Выходные данные

Единственная строка выходного файла OUTPUT.TXT должна содержать одно целое число – количество вариантов выбора участка.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	6 1 4 0 0 4 5 3 1 -2 2 1 2	15

Задача №659 Перетягивание каната

(Время: 2 сек. Память: 16 Мб Сложность: 52%)

Для участия в соревнованиях по перетягиванию каната зарегистрировалось N человек. Некоторые из участников могут быть знакомы друг с другом. Причем, если двое из них имеют общего знакомого, то это не означает, что они обязательно знакомы друг с другом.

Организаторы соревнований заинтересованы в их качественном проведении. Они хотят разделить всех участников на две команды так, чтобы в первой команде было K человек, а во второй – $N-K$ человек. Из всех возможных вариантов формирования команд, организаторы хотят выбрать такой вариант, при котором сумма сплоченностей обеих команд максимальна. Сплоченностью команды называется количество пар участников этой команды, знакомых друг с другом. Ваша задача – помочь организаторам найти требуемое разделение участников на две команды.

Входные данные

В первой строке входного файла INPUT.TXT задаются три числа N, K, M , разделенные одиночными пробелами, где N – общее число зарегистрированных участников, K – требуемое количество человек в первой команде, M – количество пар участников, знакомых друг с другом.

Каждая из следующих M строк содержит два различных числа, разделенные пробелом – номера двух участников, знакомых друг с другом. Все участники нумеруются от 1 до N .

Ограничения: все числа целые, $0 < K < N < 25, 0 \leq M \leq N(N-1)/2$

Выходные данные

Выходной файл OUTPUT.TXT должен содержать одну строку, состоящую из K чисел, каждое из которых задает номер участника, попавшего в первую команду. Числа должны быть разделены пробелами. Если существует несколько решений данной задачи, то выведите любое из них.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	5 3 3 1 3 2 5 5 4	5 2 4

Задача №660

Контрольная работа

(Время: 2 сек. Память: 16 Мб Сложность: 50%)

Петя для выполнения контрольной работы по математике аккуратно вырезал из бумаги в клеточку лист прямоугольной формы размером N клеток по вертикали и M клеток по горизонтали.

Перед уроком, в ожидании учителя, чтобы как-нибудь развлечься, Петя раскрасил ручкой на листе бумаги K различных клеток. Получив замечание учителя, Петя решил вырезать из этого листа бумаги меньший прямоугольный листок, на котором не было бы раскрашенных клеток, и сказал учителю, что знает сколькими различными способами это можно сделать.

Петя считает, что два способа вырезания прямоугольных листков являются различными, если на листе бумаги найдется хотя бы одна клетка, которая принадлежит одному из вырезаемых листков и не принадлежит другому. На самом деле, Петя не знает точного числа всевозможных способов вырезания листка прямоугольной формы. Однако он слышал о больших возможностях современных вычислительных машин, поэтому решил попросить помощи у участников республиканской олимпиады по информатике.

Производить разрезы разрешается только по линиям, разделяющим клетки друг от друга.

Входные данные

В первой строке входного файла INPUT.TXT находятся три числа N, M и K, разделенные пробелами, где N и M – размеры листа бумаги, а K – количество закрасенных Петей клеток. Каждая из следующих K строк содержит два числа, разделенные пробелом и описывающие одну закрасненную клетку. Первое число определяет номер строки листа бумаги, в которой находится закрасненная клетка, а второе число – номер столбца. Строки листа бумаги нумеруются сверху вниз от 1 до N, а столбцы – слева направо от 1 до M.

Ограничения: все числа натуральные, $K \leq 100000$; $N, M \leq 5000$; $K < NM$.

Выходные данные

Выходной файл OUTPUT.TXT должен состоять из одной строки, содержащей одно целое число, равное количеству способов, которыми можно вырезать прямоугольный листок, не содержащий раскрашенных клеток, из испорченного Петей листа бумаги.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	4 3 3 1 2 4 3 3 1	22

Задача №661

«Стабильный» интернет

(Время: 1 сек. Память: 16 Мб Сложность: 55%)

Закончив университет, Петя решил заняться разработкой web-приложений. Получив первый заказ, Петя посчитал разумным на время выполнения заказа обеспечить себя доступом в Интернет. Провайдер, с которым Петя заключил контракт, обеспечивает доступ в Интернет только посредством сервис-карт. Когда Петя пришел в местное почтовое отделение за их покупкой, то оказалось, что в продаже имеется N сервис-карт.

Каждая сервис-карта характеризуется тремя числами B_i, E_i, S_i , где S_i – цена карты, а B_i и E_i – это начало и окончание промежутка времени её действия, т.е. сервис-карта позволяет получить доступ в Интернет в любой момент времени t такой, что $B_i \leq t \leq E_i$. Время отсчитывается от некоторого фиксированного момента в прошлом.

Петя решил, что будет работать над выполнением заказа, начиная с момента времени B и заканчивая моментом времени E . Основная проблема состоит в том, что Петя не богат, поэтому он хочет на время выполнения заказа обеспечить себя «стабильным» Интернетом и потратить на это как можно меньшую сумму денег. Будем говорить, что данное множество сервис-карт обеспечивает Пете «стабильный» Интернет, если для любого момента времени t , такого, что $B \leq t \leq E$, найдется хотя бы одна сервис-карта из данного множества, которая позволяет получить доступ в Интернет в этот момент времени.

Ваша задача состоит в том, чтобы определить минимальную сумму денег, необходимую Пете для приобретения множества карт, которое обеспечит ему «стабильный» Интернет на время выполнения заказа.

Входные данные

В первой строке входного файла INPUT.TXT находится одно число N . Во второй строке два числа B и E , разделенные пробелом, где B – момент начала промежутка времени, в который Петя будет работать над выполнением заказа, а E – момент окончания.

Следующие N строк описывают сервис-карты, i -ая из этих строк описывает i -ю сервис-карту и содержит три числа B_i, E_i и S_i , разделенные одиночными пробелами.

Ограничения: все числа натуральные, $1 \leq N \leq 10^5$, $1 \leq B_i < E_i \leq 10^9$, $1 \leq B < E \leq 10^9$, $1 \leq S_i \leq 20\,000$. Существует хотя бы одно множество карт, которое обеспечивает «стабильный» Интернет на время выполнения заказа

Выходные данные

Выходной файл OUTPUT.TXT должен содержать одно число, равное минимальной сумме денег, необходимой Пете для приобретения множества сервис-карт, которое обеспечит ему «стабильный» Интернет на время выполнения заказа.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	7 10 30 9 14 10 13 19 18 14 18 16 18 24 14 24 30 9	49

Задача №662

Green Darts

(Время: 1 сек. Память: 16 Мб Сложность: 75%)

В день рождения Вовы родители решили пригласить всех его друзей. После того, как ребята выпили сок и съели яблочный пирог, они решили поиграть в увлекательную игру Green Darts, правила которой достаточно просты:

- В игре принимают участие N человек.
- Каждый игрок 3 раза подряд бросает дротик в мишень. Каждый бросок в зависимости от точности может принести игроку целое количество очков от 0 до 100.
- После того, как все броски сделаны, количество очков, набранных каждым игроком по окончании игры, вычисляется по следующей формуле:
$$S_i = a_{i1} * p^2 + a_{i2} * p + a_{i3},$$
где S_i – итоговое количество очков, которое набрал i -ый игрок; a_{ij} – количество очков, которое принес i -ому игроку сделанный им j -ый бросок; p – одинаковое для всех игроков действительное число ($0 < p \leq 1$).
- Затем для каждого игрока определяется место, которое он занял по окончании игры. Для этого используется следующая формула:
$$M_i = K_i + 1,$$
где M_i – место, которое занял i -ый игрок по окончании игры; K_i – количество игроков, которые набрали большее итоговое количество очков, чем i -ый игрок. Нетрудно видеть, что несколько игроков могли разделить одно и то же место по окончании игры.
- После этого формируется итог игры. Итог игры – это набор из N целых чисел; i -е число в этом наборе равно месту, которое занял i -ый игрок по окончании игры. Два набора из N чисел считаются различными, если существует такое i , что i -е число в первом наборе не равно i -му числу во втором наборе.

Выбор значения числа p гости решили предоставить виновнику торжества. Вова сразу задумался над тем, как его выбор может повлиять на итог игры и многое ли зависит от выбора числа p . Вова задумался над ответом на следующий вопрос: сколько существует различных итогов игры, в зависимости от выбора значения числа p ? Помогите Вове разобраться с этим нелегким вопросом.

Входные данные

Первая строка входного файла INPUT.TXT содержит целое число N – количество участников игры ($1 \leq N \leq 100$). Каждая из следующих N строк содержит 3 целых числа a_{i1} , a_{i2} , a_{i3} , разделенные одиночными пробелами, определяющие количество очков, которое принесли i -ому игроку первый, второй и третий броски соответственно ($0 \leq a_{ij} \leq 100$). Все числа целые.

Выходные данные

Выходной файл OUTPUT.TXT должен содержать целое число, равное количеству различных итогов игры, в зависимости от выбора значения числа p .

Пример

№	INPUT.TXT	OUTPUT.TXT
1	3 1 99 1 100 100 100 51 49 13	3

Задача №663
Предприимчивый Петя

(Время: 1 сек. Память: 16 Мб Сложность: 65%)

Петя решил организовать у себя в классе чемпионат по решению математических головоломок. В нем изъявили желание участвовать N человек. Каждый человек при регистрации указывает свое имя, состоящее из маленьких латинских букв. Петя предлагает участникам решить M задач, каждая из которых имеет название, состоящее из больших латинских букв.

Проверку решений задач осуществляет Петя. За каждое решение он ставит некоторое количество баллов и заносит эту информацию в некоторую базу данных на своем карманном компьютере как строку вида “A-B-C”, где A – имя участника, который сдал решение на проверку, B – название задачи, решение которой сдал этот участник, C – количество баллов, полученное решением. На каждую из предложенных задач участник мог сдать не более одного решения, кроме того, участник мог не сдавать решения некоторых задач. Количество баллов, которое набрал участник в итоге соревнования, равно сумме баллов, полученных каждым из решений, которые этот участник сдавал на проверку.

После окончания проверки решений Петя обнаружил, что информация в базе данных была повреждена, так как карманный компьютер, на котором она хранилась, был заражен компьютерным вирусом. А именно, некоторые большие и/или маленькие латинские буквы в строках, которые хранятся в базе данных, могли быть заменены на вопросительные знаки. В этот момент Петя очень пожалел, что не может заново проверить все работы, так как они у него не сохранились. Но все же Петя хочет использовать возникшую ситуацию определенным образом. Дело в том, что некоторые из участников – его хорошие друзья, и он им симпатизирует. Поэтому Петя хочет объявить результаты соревнования таким образом, чтобы все участники, которым он симпатизирует, набрали в сумме максимальное количество баллов. Однако, объявленные Петей результаты не должны противоречить информации, которая хранится в базе данных.

Сформулируем условие задачи более точно. Известны имена N человек $Name_1, \dots, Name_N$ и названия M задач $Problem_1, \dots, Problem_M$. Назовем статусной строкой строку вида “A-B-C”, где $A = Name_i$ для некоторого i , $B = Problem_j$ для некоторого j , C – целое неотрицательное число и ‘-’ – символ с ASCII-кодом 45. Величины A, B и C будем называть атрибутами статусной строки. Набор статусных строк назовем статусом, если никакие две статусные строки в этом наборе не имеют одновременно одинаковые атрибуты A и B. Назовем маской статусной строки произвольную строку, символами которой могут быть большие и маленькие латинские буквы, цифры и символы ‘-’ и ‘?’ (ASCII-код 63). Будем говорить, что некоторая статусная строка S соответствует маске Q, если в маске Q можно заменить каждый символ ‘?’ на большую или маленькую латинскую букву так, что в результате получится строка S. Произвольный набор масок статусных строк назовем маской статуса. Будем говорить, что маска статуса MS соответствует статусу ST, если количество масок в MS равно количеству статусных строк в ST и маски статусных строк из MS и статусные строки из ST можно расположить в некотором порядке так, что i -я статусная строка в ST будет соответствовать i -й маске в MS. Результатом i -го участника, вычисленным по статусу ST, назовем целое число, равное сумме значений атрибутов C по всем статусным строкам из ST, для которых значение атрибута A равно $Name_i$. Множеством симпатий назовем произвольное множество, элементами которого являются числа от 1 до N. Значением симпатичности статуса ST относительно множества симпатий L будем называть число, равное сумме результатов i -х участников, вычисленных по статусу ST, по всем i из множества L.

Ваша задача заключается в следующем. Вашей программе на вход подается маска статуса MS и множество симпатий L. Пусть X – это множество всех статусов, соответствующих маске MS. Необходимо найти максимальное из значений симпатичности по всем статусам из множества X относительно множества L.

Входные данные

Первая строка входного файла INPUT.TXT содержит целые числа N и M – количества участников и задач соответственно. Следующие N строк описывают имена участников. В i-й из этих строк записано значение Name_i – имя i-го участника. Следующие M строк описывают названия задач. В j-й из этих строк записано значение Problem_j – название j-й задачи.

В следующей строке записано целое число K – количество строк в исходной маске статуса. В каждой из следующих K строк записана одна маска статусной строки.

В следующей строке находится целое число P – количество чисел в исходном множестве симпатий. В следующей строке находятся P различных целых чисел от 1 до N, разделенных одиночными пробелами – элементы исходного множества симпатий.

Ограничения: $2 \leq N \leq 10$, $2 \leq M \leq 10$, Name_i – непустая строка из не более 100 маленьких латинских букв, Problem_j – непустая строка из не более 100 больших латинских букв. Каждая маска статусной строки содержит не более 5 цифр. Существует хотя бы один статус, соответствующий исходной маске статуса

Выходные данные

В первой строке выходного файла OUTPUT.TXT выведите одно целое число, равное максимальному из значений симпатичности по всем статусам, соответствующим исходной маске статуса, относительно исходного множества симпатий.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	3 2 petr lexa sergey PROBA PROBB 4 lexa-PROBA-100 ???g??-?????-0 ?e??-PROBA-70 ?e??-PROB?-60 2 3 2	160

Задача №664 Котлеты

(Время: 1 сек. Память: 16 Мб Сложность: 30%)

На сковородке одновременно можно положить k котлет. Каждую котлету нужно с каждой стороны обжаривать m минут непрерывно. За какое наименьшее время удастся поджарить с обеих сторон n котлет?

Входные данные

В единственной строке входного файла INPUT.TXT записано три натуральных числа: k, m, n, не превышающих 30000.

Выходные данные

В единственную строку выходного файла OUTPUT.TXT нужно вывести единственное число – минимальное время, за которое можно поджарить все котлеты.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	1 1 1	2
2	2 2 1	4

Задача №665 Палиндромное время

(Время: 1 сек. Память: 16 Мб Сложность: 24%)

Пекарь считает, что для получения рождественского пирога идеальной симметричной формы его нужно вынимать из духовки в тот момент, когда часы показывают «палиндромное» время, которое читается одинаково слева-направо и справа-налево.

Напишите программу, которая определяет по времени установки пирога в духовку время, когда будет подходящее время для его извлечения.

Входные данные

В единственной строке входного файла INPUT.TXT записано время установки пирога в духовку в формате HH:MM ($00 \leq HH \leq 23, 00 \leq MM \leq 59$).

Выходные данные

В единственную строку выходного файла OUTPUT.TXT нужно вывести ближайшее «палиндромное» время в формате HH:MM.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	00:00	01:10
2	12:34	13:31
3	23:59	00:00

Задача №666 Абракадабра

(Время: 1 сек. Память: 16 Мб Сложность: 31%)

Последовательность из латинских букв строится следующим образом. Вначале она пуста. На каждом последующем шаге последовательность удваивается, после чего к ней слева дописывается очередная буква латинского алфавита (a, b, c, ...). Ниже приведены первые шаги построения последовательности:

Шаг 1. a

Шаг 2. baa

Шаг 3. cbaabaa

Шаг 4. dcbaabaacbaabaa

.....

Требуется написать программу, которая по заданному числу N находит символ, который стоит на N-ом месте в последовательности, получившейся после 26-го шага.

Входные данные

В единственной строке входного файла INPUT.TXT записано число N ($1 \leq N < 2^{26}$).

Выходные данные

В единственную строку выходного файла OUTPUT.TXT нужно вывести символ, стоящий в N-й позиции получившейся последовательности.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	4	w

Задача №667
Автобусы - 2

(Время: 1 сек. Память: 16 Мб Сложность: 23%)

Для заезда в оздоровительный лагерь организаторы решили заказать автобусы. Известно, что в лагерь собираются поехать N детей и M взрослых. Каждый автобус вмещает K человек. В каждом автобусе, в котором поедут дети, должно быть не менее двух взрослых.

Определите, удастся ли отправить в лагерь всех детей и взрослых, и если да, то какое минимальное количество автобусов требуется для этого заказать.

Входные данные

В единственной строке входного файла INPUT.TXT записано через пробел 3 натуральных числа - N, M и K, каждое из них не превосходит 10 000.

Выходные данные

В единственную строку выходного файла OUTPUT.TXT нужно вывести количество автобусов, которые нужно заказать. Если же отправить всех в лагерь невозможно, выведите 0 (ноль).

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	10 4 7	2
2	10 4 5	0

Задача №668

Змей Горыныч

(Время: 1 сек. Память: 16 Мб Сложность: 34%)

В некотором царстве жил Змей Горыныч. У него было N голов и M хвостов. Иван-царевич решил уничтожить губителя человеческих душ, для чего ему его кума Баба Яга подарила волшебный меч, так как только им можно убить Змея Горыныча. Если отрубить одну голову, то на её месте вырастает новая, если отрубить хвост, то вместо него вырастет 2 хвоста. Если отрубить два хвоста, то вырастает 1 голова, и только когда отрубить 2 головы, то не вырастет ничего. Змей Горыныч гибнет только в том случае, когда ему отрубить все головы и все хвосты. Определить минимальное количество ударов мечом, нужное для уничтожения Змея Горыныча.

Входные данные

В единственной строке входного файла INPUT.TXT записаны через пробел два числа N, M ($0 \leq N, M \leq 1000$).

Выходные данные

В единственную строку выходного файла OUTPUT.TXT нужно вывести одно число – минимальное количество ударов мечом, или -1, если уничтожить Змея Горыныча невозможно.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	3 3	9

Задача №669

Произведение цифр - 3

(Время: 1 сек. Память: 16 Мб Сложность: 39%)

Найти наименьшее и наибольшее натуральные числа, произведение цифр в которых равно заданному натуральному числу M или сообщить, что таких чисел не существует. Для записи искомых чисел нельзя использовать цифры 0 и 1.

Входные данные

В единственной строке входного файла INPUT.TXT записано натуральное число M ($2 \leq M \leq 1000$).

Выходные данные

В единственную строку выходного файла OUTPUT.TXT нужно вывести два натуральных числа в неубывающем порядке. Если таких чисел не существует, то вывести -1 -1.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	12	2 6 3 2 2
2	11	-1 -1

Задача №670

Числа без одинаковых цифр

(Время: 1 сек. Память: 16 Мб Сложность: 25%)

Антон записал ряд натуральных чисел в порядке возрастания: 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23 и т.д. Затем вычеркнул из него все числа, в которых имеется хотя бы две одинаковых цифры, и получил последовательность: 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 23 и т.д.

Вам необходимо по заданному N найти N-ое по счету число в получившейся последовательности.

Входные данные

В единственной строке входного файла INPUT.TXT записано натуральное число N ($1 \leq N \leq 10000$).

Выходные данные

В единственную строку выходного файла OUTPUT.TXT нужно вывести N-ое по счету число без одинаковых цифр.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	100	123

Задача №671 Счастливые числа

(Время: 1 сек. Память: 16 Мб Сложность: 42%)

Все знают, что счастливые числа – это те числа, которые содержат только счастливые цифры 7 и 4. Вам нужно найти количество счастливых чисел не больших N.

Входные данные

В единственной строке входного файла INPUT.TXT записано натуральное число N, не превышающее 10^{32} .

Выходные данные

В единственную строку выходного файла OUTPUT.TXT нужно вывести одно целое число — ответ на задачу.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	56	4

Задача №672 N-значные числа

(Время: 1 сек. Память: 16 Мб Сложность: 32%)

Найти количество N-значных чисел, у которых сумма цифр равна их произведению. Вывести

наименьшее среди таких чисел для заданного N.

Входные данные

В единственной строке входного файла INPUT.TXT записано одно натуральное число N, которое не превышает 10.

Выходные данные

В единственную строку выходного файла OUTPUT.TXT нужно вывести через пробел 2 числа: количество искомых чисел и наименьшее среди них.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	1	10 0
2	2	1 22

Задача №673 N-значные числа - 2

(Время: 1 сек. Память: 16 Мб Сложность: 52%)

Найти количество N-значных чисел, у которых сумма цифр равна их произведению. Вывести наименьшее среди таких чисел для заданного N.

Входные данные

В единственной строке входного файла INPUT.TXT записано одно натуральное число N, которое не превышает 20.

Выходные данные

В единственную строку выходного файла OUTPUT.TXT нужно вывести через пробел 2 числа: количество искомых чисел и наименьшее среди них.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	1	10 0
2	2	1 22

Задача №674 Выбор приборов

(Время: 1 сек. Память: 16 Мб Сложность: 37%)

Для проведения эксперимента надо выбрать из N имеющихся приборов только три. Для этого выполняют следующую операцию - если в группе приборов больше трех, то их нумеруют и выбирают одну из групп: с четными или нечетными номерами. Операцию повторяют до тех пор, пока в группе не останется три или менее приборов. Если их остается ровно три, то они и берутся для эксперимента.

Требуется написать программу, которая подсчитает количество способов такого выбора

приборов.

Входные данные

В единственной строке входного файла INPUT.TXT записано число N ($1 \leq N \leq 2147483647$).

Выходные данные

В единственную строку выходного файла OUTPUT.TXT нужно вывести одно число - найденное количество способов выбора приборов.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	3	1
2	6	2

Задача №675 Детали

(Время: 1 сек. Память: 16 Мб Сложность: 20%)

На клеточном поле $N \times M$ расположены две жёсткие детали. Деталь А накрывает в каждой строке несколько (не ноль) первых клеток, деталь В — несколько (не ноль) последних; каждая клетка либо полностью накрыта одной из деталей, либо нет.

A	A	.	B	B	B
A	B
A	A	A	.	.	B
A	.	.	B	B	B

Деталь В начинают двигать влево, не поворачивая, пока она не упрётся в А хотя бы одной клеткой. Определите, на сколько клеток будет сдвинута деталь В.

Входные данные

В первой строке входного файла INPUT.TXT записано два числа N и M — число строк и столбцов соответственно ($1 \leq N, M \leq 100$). Далее следуют N строк, задающих расположение деталей. В каждой находится ровно M символов "А" (клетка, накрытая деталью А), "В" (накрытая деталью В) или "." (свободная клет-ка).

Выходные данные

В единственную строку выходного файла OUTPUT.TXT нужно вывести одно число — ответ на задачу.

Пример

№	INPUT.TXT	OUTPUT.TXT
	4 6 AA.BBB	1

1	A...B AAA..B A..BBB	
---	---------------------------	--

Задача №676

Игра со спичками

(Время: 1 сек. Память: 16 Мб Сложность: 28%)

Двое играют в следующую игру. Из кучки спичек за один ход игрок вытягивает либо 1, либо 2, либо 1000 спичек. Выигрывает тот, кто забирает последнюю спичку. Кто выигрывает при правильной игре?

Входные данные

В единственной строке входного файла INPUT.TXT записано одно натуральное число — N ($1 \leq N \leq 10000$) начальное количество спичек в кучке.

Выходные данные

В единственную строку выходного файла OUTPUT.TXT нужно вывести 1, если выигрывает первый игрок (тот, кто ходит первым), или 2, если выигрывает второй игрок.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	2	1
2	3	2

Задача №677

Количество участников олимпиады

(Время: 1 сек. Память: 16 Мб Сложность: 17%)

Как известно, на вопрос о том, сколько у него учеников, древнегреческий учёный Пифагор отвечал так: "Половина моих учеников изучает математику, четвертая часть изучает природу, седьмая часть проводит время в молчаливом размышлении, остальную часть составляют 3 девы".

Секретарь олимпиады на вопрос: "Сколько участников на олимпиаде по информатике?", отвечал подобно Пифагору: "К-тая часть участников начала решать первую задачу, М-тая часть – вторую, а N-тая – третью. В то же время D участников решают проблему: "С чего начать?". Ваша задача определить количество участников олимпиады или вывести -1, если секретарь ошибся.

Входные данные

В единственной строке входного файла INPUT.TXT записаны числа K, N, M, D ($1 \leq K, N, M, D \leq 100$).

Выходные данные

В единственную строку выходного файла OUTPUT.TXT нужно вывести одно число - количество участников олимпиады или вывести -1, если секретарь ошибся.

Примеры

--	--	--

№	INPUT.TXT	OUTPUT.TXT
1	2 4 7 3	28
2	2 2 2 3	-1

Задача №678

Напёрстки

(Время: 1 сек. Память: 16 Мб Сложность: 15%)

Шулер показывает следующий трюк. Он имеет три одинаковых наперстка. Под первый (левый) он кладет маленький шарик. Затем он очень быстро выполняет ряд перемещений наперстков, каждое из которых – это одно из трех перемещений - А, В, С:

- А - обменять местами левый и центральный наперстки,
- В - обменять местами правый и центральный наперстки,
- С - обменять местами левый и правый наперстки.

Необходимо определить, под каким из наперстков окажется шарик после всех перемещений.

Входные данные

В единственной строке входного файла INPUT.TXT записана строка длиной не более 50 символов из множества {А, В, С} – последовательность перемещений.

Выходные данные

В единственную строку выходного файла OUTPUT.TXT нужно вывести номер наперстка, под которым окажется шарик после перемещений.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	СВАВСАССС	1

Задача №679

Просто простые числа

(Время: 1 сек. Память: 16 Мб Сложность: 40%)

Дано натуральное число N. Представить его в виде суммы простых натуральных чисел так, чтобы произведение этих слагаемых было максимально.

Входные данные

В единственной строке входного файла INPUT.TXT записано одно натуральное число N ($1 < N < 2\,000\,000\,000$).

Выходные данные

В единственную строку выходного файла OUTPUT.TXT нужно вывести простые числа по возрастанию с указанием их количества при разложении, т.е.: <число> <количество>.

Примеры

--	--	--

№	INPUT.TXT	OUTPUT.TXT
1	5	2 1 3 1
2	30	3 10

Задача №680
Садовник-художник

(Время: 1 сек. Память: 16 Мб Сложность: 28%)

Садовник посадил N деревьев в один ряд. После посадки деревьев садовнику нужно их покрасить. В его распоряжении есть краска трех цветов: белая, синяя и оранжевая. Сколько способов покраски деревьев есть у него, если никакие два соседних дерева нельзя красить в одинаковый цвет?

Входные данные

В единственной строке входного файла INPUT.TXT записано одно натуральное число - количество деревьев N ($1 \leq N \leq 50$).

Выходные данные

В единственную строку выходного файла OUTPUT.TXT нужно вывести одно число - количество способов покраски.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	3	12

Задача №681
Спички

(Время: 1 сек. Память: 16 Мб Сложность: 43%)

Какое минимальное количество спичек необходимо для того, чтобы выложить на плоскости N квадратов со стороной в одну спичку? Спички нельзя ломать и класть друг на друга. Вершинами квадратов должны быть точки, где сходятся концы спичек, а сторонами – сами спички.

Напишите программу, которая по количеству квадратов N, которые необходимо составить, находит минимальное необходимое для этого количество спичек.

Входные данные

В единственной строке входного файла INPUT.TXT записано одно целое число N ($1 \leq N \leq 10^9$).

Выходные данные

В единственную строку выходного файла OUTPUT.TXT нужно вывести одно целое число – минимальное количество спичек, требуемых для составления заданного количества квадратов.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	4	12

Задача №682

Сумма n-значных чисел

(Время: 1 сек. Память: 16 Мб Сложность: 30%)

По заданному n найти сумму всех n-значных чисел.

Входные данные

В единственной строке входного файла INPUT.TXT записано одно натуральное число n ($1 \leq n \leq 100$).

Выходные данные

В единственную строку выходного файла OUTPUT.TXT нужно вывести все цифры суммы всех n-значных чисел.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	1	45
2	2	4905

Задача №683

Числа - 3

(Время: 1 сек. Память: 16 Мб Сложность: 58%)

Дана последовательность чисел a_1, a_2, \dots, a_N . За одну операцию разрешается удалить любое (кроме крайних) число, заплатив за это штраф, равный произведению этого числа на сумму соседних. Требуется удалить все числа, кроме крайних, с минимальным суммарным штрафом.

Например:

- Начальная последовательность: **1 50 51 50 1**.
- Удаляем четвёртое число, штраф $50(51+1)=2600$, получаем **1 50 51 1**.
- Удаляем третье число, штраф $51(50+1)=2601$, получаем **1 50 1**.
- Удаляем второе число, штраф $50(1+1)=100$.
- Итого штраф **5301**.

Входные данные

В первой строке входного файла INPUT.TXT записано одно число N ($1 \leq N \leq 100$) - количество чисел в последовательности.

Во второй строке находятся N целых чисел a_1, a_2, \dots, a_N ; никакое из чисел не превосходит по модулю 100.

Выходные данные

В единственную строку выходного файла OUTPUT.TXT нужно вывести одно число - минимальный суммарный штраф.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	5 1 50 51 50 1	5301

Задача №684
Шашки - 2

(Время: 1 сек. Память: 16 Мб Сложность: 27%)

На доске стоит белая шашка. Требуется определить, может ли она попасть в заданную клетку, делая ходы по правилам (не превращаясь в дамку).

Входные данные

В единственной строке входного файла INPUT.TXT записаны клетка, где стоит шашка, в шахматной нотации, а затем, через пробел, клетка, куда шашка должна попасть. Начальная и конечная клетки не совпадают.

Выходные данные

В единственную строку выходного файла OUTPUT.TXT нужно вывести слово YES (заглавными буквами), если шашка может попасть из начальной клетки в конечную, и NO в противном случае.

Примеры

№	INPUT.TXT	OUTPUT.TXT	Комментарии
1	a1 b2	YES	Для выполнения указанного перемещения шашка должна сделать один ход вперед и вправо
2	b2 a1	NO	Назад шашка ходить не может
3	a1 h7	NO	a1 и h7 – клетки разного цвета
4	a1 h8	YES	7 ходов вправо вверх

Пояснение

Доска имеет размер 8x8, вертикали нумеруются маленькими латинскими буквами от a до h, горизонтали - числами от 1 до 8. Белая шашка ходит по чёрным полям по диагонали вверх.

Задача №685
Золотой песок

(Время: 1 сек. Память: 16 Мб Сложность: 10%)

Сотрудники завода по производству золотого песка из воздуха решили поправить свое финансовое положение. Они пробрались на склад завода, где хранился золотой песок трех видов. Один килограмм золотого песка первого вида они смогли бы продать за A_1 рублей, второго вида – за A_2 рублей, а третьего вида – за A_3 рублей. Так получилось, что у сотрудников оказалось с собой только три емкости: первая была рассчитана на V_1 килограмм груза, вторая на V_2 килограмм, а третья на V_3 килограмм. Им надо было заполнить полностью все емкости таким образом, чтобы получить как можно больше денег за весь песок. При заполнении емкостей нельзя смешивать песок разных видов, то есть, в одну емкость помещать более одного вида песка, и заполнять емкости песком так, чтобы один вид песка находился более чем в одной емкости.

Требуется написать программу, которая определяет, за какую сумму предприимчивые сотрудники смогут продать весь песок в случае наилучшего для себя заполнения емкостей песком.

Входные данные

В единственной строке входного файла INPUT.TXT записано 6 натуральных чисел $A_1, A_2, A_3, B_1, B_2, B_3$, записанных в одной строке через пробел. Все числа не превосходят 100.

Выходные данные

В единственную строку выходного файла OUTPUT.TXT нужно вывести единственное целое число – сумму в рублях, которую смогут сотрудники заработать в случае наилучшего для себя заполнения емкостей песком.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	1 2 3 3 2 1	14

Задача №686 Колокол

(Время: 1 сек. Память: 16 Мб Сложность: 33%)

Требуется написать программу, которая в массиве из n целых чисел наименьший элемент поместит на первое место, наименьший из оставшихся – на последнее, следующий по величине – на второе место, следующий – на предпоследнее и так далее – до середины массива.

Входные данные

Во входном файле INPUT.TXT записано в первой строке число n ($1 \leq n \leq 30000$). Во второй строке записаны через пробел элементы массива, числа по абсолютной величине не большие 32767.

Выходные данные

В единственную строку выходного файла OUTPUT.TXT нужно вывести элементы полученного массива, разделенные одним пробелом.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	5 1 2 3 4 5	1 3 5 4 2

Задача №687 Однонаправленная задача коммивояжёра

(Время: 1 сек. Память: 16 Мб Сложность: 41%)

Задана матрица размером $m \times n$ из целых чисел. Путь начинается в любой строке первого столбца и состоит из последовательности шагов, обрывающихся в столбце n . Каждый шаг состоит в переходе из столбца i в столбец $i+1$ в соседнюю (по горизонтали или диагонали) ячейку. Весом пути называется сумма целых чисел, записанных в каждой из n посещенных ячеек.

Требуется написать программу, которая вычисляет путь с минимальным весом с левого края матрицы до правого.

Входные данные

Входной текстовый файл INPUT.TXT содержит в первой строке количество строк и столбцов матрицы, которые обозначаются m и n соответственно. Далее следует m строк по n чисел в каждой. Числа отделяются друг от друга пробелами. Число строк не превышает 10, столбцов – 100. Вес любого пути не будет превышать целого числа, для хранения которого потребуется больше 30 бит.

Выходные данные

Выходной текстовый файл OUTPUT.TXT должен содержать две строки. Первая строка задает путь минимальной стоимости, а вторая – соответственно стоимость этого пути. Путь состоит из последовательности n целых чисел (разделенных одним или более пробелами), задающих номера строк, из которых состоит путь минимальной стоимости. Если путей минимальной стоимости больше одного, то должен быть выведен лексикографически минимальный путь.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	5 6 3 4 1 2 8 6 6 1 8 2 7 4 5 9 3 9 9 5 8 4 1 3 2 6 3 7 2 8 6 4	1 2 3 4 4 5 16

Задача №688 Суслик и собака

(Время: 1 сек. Память: 16 Мб Сложность: 19%)

На большом поле находятся суслик и собака. Собака хочет суслика съесть, а суслик хочет оказаться в безопасности, добежав до одной из норок, выкопанных в поле. Ни собака, ни суслик в математике не сильны, но, с другой стороны, они и не беспросветно глупы. Суслик выбирает определенную норку и бежит к ней по прямой с определенной скоростью. Собака, которая очень хорошо понимает язык телодвижений, угадывает, к какой норке бежит суслик, и устремляется к ней со скоростью вдвое большей скорости суслика. Если собака добегает до норки первой, то она съедает суслика; иначе суслик спасается.

Требуется написать программу, которая поможет суслику выбрать норку, в которой он может спастись, если таковая существует.

Входные данные

Во входном файле INPUT.TXT записано в первой строке два числа – координаты суслика. Во второй строке записаны два числа – координаты собаки. В третьей строке записано число n – число норок на поле. В следующих n строках записаны координаты норок. Все координаты являются целыми числами, по модулю не превышающими 10000, и записываются через пробел. Количество норок не превышает 1000.

Выходные данные

В единственную строку выходного файла OUTPUT.TXT нужно вывести число – номер норки,

если у суслика есть возможность в ней спастись. Если у суслика есть возможность спрятаться в нескольких норках, то выведите ту, которая первая шла во входных данных. Если суслик не может спастись, то выведите в выходной файл «NO» (без кавычек).

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	10 10 20 20 1 15 15	NO
2	20 20 10 10 2 15 15 25 25	2

Задача №689 Упрощение номеров

(Время: 1 сек. Память: 16 Мб Сложность: 26%)

Как, Вы не можете запомнить 6 или 7-значный номер телефона, появившийся на секунду на экране телевизора?! С помощью специальной методики, описываемой далее, Вы превратитесь в ходячий телефонный справочник!

Очевидно, что число 402 запомнить легче, чем число 110010010, а число 337377 запомнить легче, чем число 957472. Значит, нужно чтобы запоминаемое число, с одной стороны, содержало как можно меньше цифр, а с другой стороны, желательно, чтобы в числе было как можно больше повторяющихся цифр. В качестве критерия сложности запоминания примем сумму количества цифр в числе и количества различных цифр в числе. Запоминаемое число можно записать в другой системе счисления, возможно, тогда его окажется легче запомнить. Например, число 65535 в шестнадцатеричной системе исчисления выглядит как FFFF. Таким образом, нужно написать программу подбора основания системы счисления для минимизации критерия сложности. Основание системы счисления нужно выбирать в диапазоне от 2 до 36, тогда для представления числа можно использовать цифры 0-9 и латинские буквы A-Z.

Входные данные

Входной файл INPUT.TXT содержит в первой строке целое число n ($1 \leq n \leq 100$). Далее следует n строк, каждая строка содержит целое число от 1 до 999999999.

Выходные данные

Выходной файл OUTPUT.TXT должен содержать n строк. Для каждого из n заданных чисел строка содержит: основание системы счисления (от 2 до 36), минимизирующее критерий сложности запоминания, и число в выбранной системе исчисления, разделенные одним пробелом. Если несколько оснований дают одинаковое значение критерия, то выбрать наименьшее среди них.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	2 2	3 2 16 FFFF

Задача №690

Конная прогулка

(Время: 1 сек. Память: 16 Мб Сложность: 71%)

Требуется выполнить обход прямоугольного поля, перемещаясь в нем по правилам шахматного коня. В лабиринте имеются клетки, перемещение в которые невозможно. Начальная позиция коня определена. Необходимо посетить все клетки (в которые переход возможен) без повторных заходов. Гарантируется, что такой обход существует.

Входные данные

В первой строке входного файла INPUT.TXT содержится два натуральных числа N и M – размеры поля ($N, M \leq 100$). Далее, следует карта поля: N строк по M символов в каждой строке. Символом «.» (точка) обозначается пустое пространство. Символ «X» указывает на то, что перемещение в данную клетку поля запрещено. Начальная позиция коня задается единственным в поле символом «K».

Выходные данные

В выходной файл OUTPUT.TXT выведите матрицу обхода поля, в каждой ячейке которого должен быть вписан номер шага ее посещения (начиная с единицы). Клетки, помеченные как «X» во входных данных следует при выводе помечать нулевым значением. Числа следует разделять пробелами, допускается использовать лишние пробелы. В случае неоднозначного решения следует вывести любое.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	5 5 K.....	1 14 9 20 3 24 19 2 15 10 13 8 25 4 21 18 23 6 11 16 7 12 17 22 5
2	6 8 X.....X ..K..X..X.. ..XXXX.. X.....X	0 11 32 23 2 21 30 0 33 24 1 10 31 0 3 20 12 9 38 25 22 0 16 29 37 34 0 0 0 0 19 4 8 13 26 35 6 15 28 17 0 6 7 14 27 18 5 0
3	9 9K....XXXXX.. ..X...X.. ..X.X.X.. ..X...X.. ..XXXXX..	15 18 45 32 1 20 47 34 3 44 31 16 19 46 33 2 21 48 17 14 0 0 0 0 0 4 35 30 43 0 57 60 63 0 49 22 13 56 0 62 0 58 0 36 5 42 29 0 59 64 61 0 23 50 55 12 0 0 0 0 0 6 37 28 41 10 53 26 39 8 51 24 11 54 27 40 9 52 25 38 7

Задача №691

Номера автобусов

(Время: 1 сек. Память: 16 Мб Сложность: 16%)

Однажды Вася очень долго просидел на остановке, прежде чем дождался своего автобуса. Чтобы как-то занять время, он решил записывать на листочке государственные регистрационные номера проходящих мимо автобусов, следующих по другому маршруту, нежели нужен был Васе. При этом производилась запись лишь основного номера, без учета региональной принадлежности. В итоге Васе удалось записать N таких номеров.

Основная часть государственного регистрационного номера состоит из 6 символов: трех букв и трех цифр. Сначала идет буква, затем 3 цифры и еще 2 буквы заканчивают запись. В качестве цифр могут использоваться любые цифры от 0 до 9, а в качестве букв только прописные буквы, обозначения которых присутствуют как в английском, так и в русском алфавите, т.е. только следующие символы: A, B, C, E, H, K, M, O, P, T, X, Y. Например, «P204BT» - правильный номер, а «X182YZ» и «ABC216» - нет.

Ваша задача заключается в проверке правильности проделанной Васей работы. А именно, нужно определить, какие из номеров соответствуют принятому стандарту, а какие нет.

Входные данные

Первая строка входного файла INPUT.TXT содержит единственное натуральное число N – количество записанных Васей номеров ($N \leq 50$). Далее следует N строк с записями номеров автобусов. Длины строк не превышают 300 и содержат только символы с кодами ASCII от 33 до 127 (не содержат пробелов, специальных и русских символов).

Выходные данные

В выходной файл OUTPUT.TXT выведите N строк, в i-й строке должно содержаться «Yes», если соответствующая i-я запись номера верна и «No» в противном случае.

Пример

№	INPUT.TXT	OUTPUT.TXT
1	5	Yes
	P204BT	No
	X182YZ	No
	a216bc	Yes
	A216BC	No
	ABC216	

Задача №692

Бинарные числа

(Время: 1 сек. Память: 16 Мб Сложность: 8%)

Говорят, что плохой программист – это тот, кто считает, что в одном килобайте 1000 байт, а хороший программист – это тот, кто полагает, что в одном километре 1024 метра.

Многим эта шутка понятна, так как все знают, что в процессах, связанных с информатикой и компьютерной техникой, фигурирует множество значений, выражаемых степенью двойки, то есть чисел вида 2^K , где K – некоторое неотрицательное целое число. Назовем такие числа *бинарными*.

Это такие числа как 2, 4, 8, 16, 32 и т.д. Действительно, когда речь идет о размере памяти или о разрешении экрана монитора, то мы часто наталкиваемся на бинарные числа. Все это связано с принципом хранения информации в памяти ЭВМ.

Задано целое число N. Требуется определить, является ли оно бинарным.

Входные данные

Входной файл INPUT.TXT содержит единственное целое число N, не превосходящее 10000 по абсолютной величине.

Выходные данные

В выходной файл OUTPUT.TXT выведите YES, если заданное число является бинарным, и NO в противном случае.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	1024	YES
2	23	NO

Задача №693 Перестановка

(Время: 1 сек. Память: 16 Мб Сложность: 24%)

Если Вы читали Гарри Поттера, то знаете, что повелитель зла, Лорд Волдеморт создал свое имя путем перестановки букв в своем настоящем имени. Так из имени «Tom Marvolo Riddle» он получил «I am Lord Voldemort».

Напишите программу, которая проверяет, можно ли получить из одного имени другое путем перестановки его буквы. При этом регистром букв нужно пренебречь.

Входные данные

В первой строке входного файла INPUT.TXT содержатся два слова, разделенные единственным в строке пробелом. Слова содержать только символы латинского алфавита. Длина слов больше 0 и не превышает 50 символов.

Выходные данные

В выходной файл OUTPUT.TXT выведите «Yes», если возможно получить из одного имени другое, и «No» в противном случае.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	TomMarvoloRiddle IamLordVoldemort	Yes
2	stop pots	Yes
3	abbc bac	No

Задача №694

Лентяй

(Время: 1 сек. Память: 16 Мб Сложность: 21%)

Студент Валера являет собой классический пример лентяя. На занятия он практически не ходит, и только в конце семестра появляется в университете и сдает ”хвосты”. Его заветная мечта: найти такой день, когда можно будет сдать сразу все долги. У него есть расписание работы преподавателей, из которого точно известно, с какого и по какой день месяца каждый преподаватель ежедневно будет доступен.

Помогите Валере написать программу, которая по расписанию будет определять, сможет ли Валера сдать все долги за один день или нет.

Входные данные

В первой строке входного файла INPUT.TXT содержится натуральное число N – количество предметов, которые нужно сдать Валере ($N \leq 100$). Далее идет N строк, каждая из которых состоит из двух чисел A и B, задающих отрезок работы очередного преподавателя ($1 \leq A \leq B \leq 31$).

Выходные данные

В выходной файл OUTPUT.TXT выведите «YES», если возможно встретить всех преподавателей за один день, или «NO», если это сделать невозможно.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	1 1 2	YES
2	2 1 2 3 4	NO
3	3 1 8 3 5 4 9	YES

Задача №695 Мифические шахматы

(Время: 1 сек. Память: 16 Мб Сложность: 45%)

Ваш друг Вася занимается разработкой компьютерной игры «Мифические шахматы». Он не укладывается в установленные сроки сдачи проекта.

Вася обратился к друзьям за помощью. Ему необходим модуль, вычисляющий оптимальные пути перемещения фигур из одной клетки в другую. Так как друзей у Васи много, то каждому досталась маленькая подзадача. Вам требуется написать программу, определяющую минимальное количество ходов, необходимое кентавру, чтобы добраться из одной клетки в другую.

В мифические шахматы играют на шахматной доске размером 9x9, угловые клетки которой окрашены в черный цвет. Кентавр – фигура мифических шахмат, объединяющая в себе свойства коня и слона. Когда кентавр стоит на белой клетке, он может ходить только как конь, а когда на черной – только как слон. На рисунках приведены варианты ходов для двух кентавров (буквой «К» отмечено местоположение кентавра, а звездочками – клетки, куда кентавр может сделать ход).

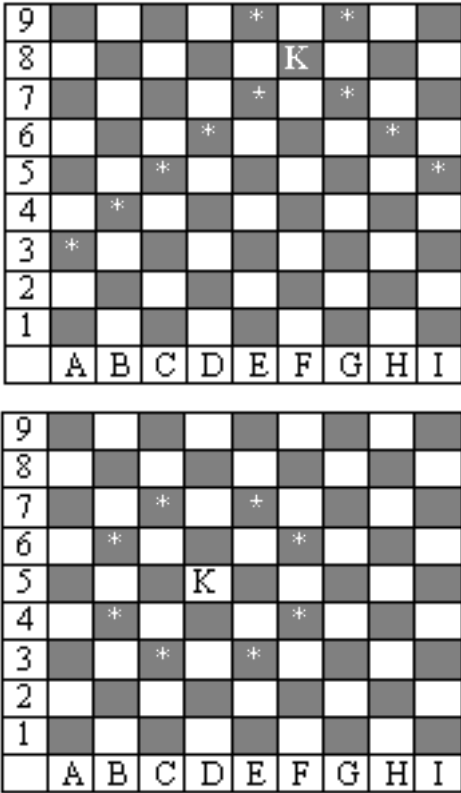
Входные данные

В первой строке входного файла INPUT.TXT содержатся координаты (большая латинская буква и цифра) двух клеток доски для мифических шахмат, разделенные пробелом.

Выходные данные

В выходной файл OUTPUT.TXT выведите минимальное число ходов, необходимое кентавру, чтобы добраться из первой клетки во вторую. Если добраться невозможно, то следует вывести число «-1» (без кавычек).

Примеры



№	INPUT.TXT	OUTPUT.TXT
1	H6 E5	2
2	A6 F6	3

Задача №696 Сон

(Время: 1 сек. Память: 16 Мб Сложность: 50%)

Любознательный школьник Петя очень любит программировать. Однажды он настолько увлекся этим делом, что уснул прямо за компьютером! Пете приснилось, что он попал в альтернативную реальность. Альтернативная реальность представляет собой прямоугольный лабиринт, который можно изобразить в виде таблицы размером R × C клеток. Чтобы не опоздать школу, Пете нужно найти выход из лабиринта. Начальная позиция Пети обозначается символом 'S', выход из альтернативной реальности – 'E'. За один ход Петя перемещается в одну из четырех смежных клеток (влево, вправо, вниз, вверх). Если клетка занята стеной (символ 'X'), то Петя пройти в нее не может. В некоторых клетках расположены двери с замками одного из четырех цветов ('R', 'G', 'B', 'Y'). Для прохода в эту клетку, необходимо обладать ключом определенного цвета. Так как ключи многоразовые, то одним ключом можно открыть сколь угодно много соответствующих ему замков.

Властелин альтернативной реальности предлагает Пете купить ключи, чтобы пройти к выходу. У нашего героя совсем немного денег с собой, поэтому ему хочется потратить как можно меньше денег и при этом пройти к выходу. Помогите ему определить минимальную сумму денег, которую нужно потратить на ключи.

Входные данные

Первая строка входного файла INPUT.TXT содержит натуральные числа R и C (R,C ≤ 50). Вторая строка теста содержит 4 целых числа P_i, стоимости покупки ключей 'R', 'G', 'B' и 'Y' соответственно (P_i ≤ 100). Далее следуют R строк, в каждой из которых C символов, описывающих лабиринт. Каждый лабиринт содержит только один символ 'S' и один символ 'E'.

Выходные данные

В выходной файл OUTPUT.TXT выведите минимальную сумму денег, необходимую для покупки набора ключей, позволяющего пройти к выходу. Если пути к выходу из альтернативной

реальности не существует, и Петя обречен проспать уроки, выведите «Sleep».

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	3 7 1 1 1 1 XXXXXXXX XS.X.EX XXXXXXXX	Sleep
2	6 6 1 5 3 1 XXXXXX XS.X.X X..R.X X.XXBX X.G.EX XXXXXX	4

Задача №697 Ремонт

(Время: 1 сек. Память: 16 Мб Сложность: 11%)

Ваш любимый дядя – директор фирмы, которая делает евроремонты в офисах. В связи с финансово-экономическим кризисом, дядюшка решил оптимизировать свое предприятие.

Давно ходят слухи, что бригадир в дядюшкиной фирме покупает лишнее количество стройматериалов, а остатки использует для отделки своей новой дачи. Ваш дядя заинтересовался, сколько в действительности банок краски необходимо для покраски стены в офисе длиной L метров, шириной – W и высотой – H , если одной банки хватает на 16м^2 , а размерами дверей и окон можно пренебречь? Заказов много, поэтому дядя попросил написать программу, которая будет все это считать.

Входные данные

Входной файл INPUT.TXT содержит три натуральных числа L , W , H – длину, ширину и высоту офиса в метрах соответственно, каждое из которых не превышает 1000.

Выходные данные

В выходной файл OUTPUT.TXT выведите одно целое число – минимальное количество банок краски, необходимых для покраски офиса.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	8 8 2	4
2	1 1 3	1

Задача №698 Игра в дурака

(Время: 1 сек. Память: 16 Мб Сложность: 36%)

Как Вам уже стало известно, Петя очень любит программировать. Недавно он решил реализовать популярную карточную игру «Дурак». Но у Пети пока маловато опыта, ему срочно нужна Ваша помощь.

Как известно, в «Дурака» играют колодой из 36 карт. В Петиной программе каждая карта представляется в виде строки из двух символов, где первый символ означает ранг ('6', '7', '8', '9', 'T', 'J', 'Q', 'K', 'A') карты, а второй символ означает масть ('S', 'C', 'D', 'H'). Ранги перечислены в порядке возрастания старшинства.

Пете необходимо решить следующую задачу: сможет ли игрок, обладая набором из N карт, отбить M карт, которыми под него сделан ход? Для того чтобы отбиться, игроку нужно покрыть каждую из карт, которыми под него сделан ход, картой из своей колоды. Карту можно покрыть либо старшей картой той же масти, либо картой козырной масти. Если кроющаяся карта сама является козырной, то её можно покрыть только старшим козырем. Одной картой можно покрыть только одну карту.

Входные данные

В первой строке входного файла INPUT.TXT находятся два натуральных числа N и M ($N \leq 35$, $M \leq 4$, $M \leq N$), а также символ R, означающий козырную масть. Во второй строке перечислены N карт, находящихся на руках у игрока. В третьей строке перечислены M карт, которые необходимо отбить. Все карты отделены друг от друга одним пробелом.

Выходные данные

В выходной файл OUTPUT.TXT выведите «YES» в случае, если отбиться можно, либо «NO» , если нельзя.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	6 2 C KD KC AD 7C AH 9C 6D 6C	YES
2	4 1 D 9S KC AH 7D 8D	NO

Задача №699 Грибной дождь

(Время: 1 сек. Память: 16 Мб Сложность: 30%)

На лесной опушке растет дружная семейка грибов. Местоположение каждого гриба задается координатами X, Y, а шляпка гриба имеет радиус R. Когда идет дождь, радиус шляпки каждого гриба непрерывно и равномерно увеличивается со скоростью 1 сантиметр в минуту. Когда дождь заканчивается (а он идет не более T минут), шляпки прекращают расти. Если во время дождя шляпки двух грибов соприкоснулись, то они немедленно перестают расти, чтобы не навредить друг другу. Грибы очень дружные, поэтому если перестают расти два гриба, то и все остальные тоже не растут.

Требуется посчитать, на сколько сантиметров увеличился радиус шляпки каждого гриба после завершения дождя.

Входные данные

Первая строка входного файла INPUT.TXT содержит два натуральных числа: количество грибов K ($K \leq 10$) и длительность дождя T ($T \leq 100$). Следующие K строк содержат описание грибов: целые координаты X и Y ($0 \leq X, Y \leq 100$) и радиус шляпки R ($1 \leq R \leq 10$). Координаты и радиус даны в сантиметрах.

Выходные данные

В выходной файл OUTPUT.TXT выведите величину в сантиметрах, на которую увеличится радиус всех грибов. Результат следует вывести с точностью, не меньшей, чем два знака после запятой.

Примеры

№	INPUT.TXT	OUTPUT.TXT
1	2 1 0 0 1 2 2 1	0.41
2	3 2 0 0 1 5 5 1 10 10 1	2.00

Задача №700 Мёд

(Время: 1 сек. Память: 16 Мб Сложность: 23%)

Все любят сладости и, в частности, мед. Винни-Пух тоже его любит. Каждый день он шел лакомиться медом, а по дороге домой заходил в гости к Кролику. Но приближалась зима, и Винни-Пух начал задумываться о запасах. Он решил в течении N дней не лакомиться медом, а собирать полный горшочек объемом V горстей и перекладывать его в бочку. В первый день своего собирательства он так и сделал. Терпения хватило на один день. А на следующий день он не смог устоять и по дороге домой съел горсть меда из горшочка. В каждый следующий день из полного горшочка он съедал на одну горсть больше.

Необходимо определить объем меда, собранного Винни-Пухом на зиму.

Входные данные

Входной файл INPUT.TXT содержит три натуральных числа N ($N \leq 300$), V ($V \leq 10^7$) и K ($K \leq 100$). K – объем, на который Винни-Пух с каждым днем съедал больше меда.

Выходные данные

В выходной файл OUTPUT.TXT выведите два значения через пробел. Сначала идет строка «NO», если случилось, что Винни-Пух пришел к Кролику с пустым горшочком и «YES» в противном случае. Второе значение – объем меда, заготовленного Винни-Пухом на зиму.

Примеры

№	INPUT.TXT	OUTPUT.TXT

1	15 100 10	NO 550
2	10 10 1	YES 55