

Problem A. Message

Input file: **standard input**
Output file: **standard output**
Time limit: **2 seconds**
Memory limit: **64 Mebibytes**

Шифрованное сообщение, отправленное Юстасом Алексу, в качестве «контрольной суммы» содержит некоторую перестановку последовательности $(0, 1, 2, \dots, N)$, из которой убрано одно число. По отсутствующему числу Центр может узнать, является ли шифровка дезинформацией.

Каждое число передаётся двумя цифрами. Однако из-за помех при приёме радиogramмы некоторые цифры кода распознать не удалось. По записи принятых данных требуется определить наименьшее и наибольшее число, которое могло отсутствовать в этой последовательности.

Input

В первой строке входного файла задано целое положительное число N ($1 \leq N \leq 99$). В следующей строке через пробел записаны N чисел, причём нераспознанные цифры в них представлены знаками вопроса.

Output

В выходной файл выведите два числа через пробел — минимальное и максимальное число, которое могло отсутствовать в принятой радиogramме.

Example

standard input	standard output
11 05 ?2 0? 04 ?1 06 07 09 08 1? ?1	00 03

Problem B. Numbers

Input file: standard input
Output file: standard output
Time limit: 2 seconds
Memory limit: 64 Mebibytes

Цифровой замок в кабинете у Мюллера выглядит следующим образом: заданы n натуральных чисел. Замок открывается, если введённое число не будет делиться ни на одно из заданных чисел.

Ваша задача — найти хотя бы одно подобное число.

Input

Первая строка входного файла содержит число $1 \leq n \leq 10$. В следующей строке через пробелы записано n натуральных чисел. Каждое число не превосходит 100.

Output

В единственной строке выходного файла выведите любое целое положительное число, не превосходящее 10^{20} и не делящееся ни на одно из n заданных чисел. Если такого числа не существует, то выведите число -1 .

Example

standard input	standard output
3 2 3 5	31
4 2 2 3 10	121
3 1 2 3	-1

Problem C. Numbers

Input file: standard input
Output file: standard output
Time limit: 2 seconds
Memory limit: 64 Mebibytes

Для облегчения запоминания телефонных номеров на некоторых аппаратах на кнопках с цифрами также написаны буквы. Будем считать номер хорошим, если при его наборе можно получить какое-то осмысленное слово (или несколько слов подряд без пробелов), взяв любую букву с первой нажатой кнопки, затем любую со второй и т.д.

Телефонные номера в Берлине имеют 6 цифр. В этой задаче будем считать, что номер может быть последовательностью любых 6 цифр (в том числе с ведущими нулями). Дан словарь, в котором все слова имеют длину от 1 до 6 букв. Также известно размещение букв на кнопках телефонного аппарата. Требуется определить, сколько телефонных номеров может быть представлено словами или цепочками слов из этого словаря (в цепочке слова могут повторяться).

Input

Первые десять строк входного файла mnemonic.in описывают размещение букв на кнопках. В первой строке находится цифра 0 и через пробел строка из назначенных ей букв (буквы записаны без пробелов), во второй строке стоит цифра 1, пробел и её буквы и так далее до цифры 9 включительно. Возможно, что каким-то кнопкам ни одной буквы не назначено — в строке, соответствующей этой кнопке, стоит цифра, за которой следует один пробел. Гарантируется, что каждая буква встретится на какой-нибудь цифре и что ни одна буква не встретится более двух раз. В следующей строке записано число N от 1 до 10^4 — количество слов в словаре. Далее следует N строк, каждая строка содержит одно слово, записанное строчными латинскими буквами.

Output

Выведите в первой строке выходного файла одно число — найденное количество номеров.

Example

standard input	standard output
0 wabq	4
1 ceyr	
2 td	
3 hfg	
4 jki	
5 mln	
6 po	
7 us	
8	
9 zvx	
3	
abc	
cde	
qwerty	

Problem D. Space war

Input file: standard input
Output file: standard output
Time limit: 2 seconds
Memory limit: 64 Mebibytes

Штирлиц бегло ознакомился с принесёнными его агентом документами. Да, это обещало стать сенсацией. Перед ним был проект договора, подписанного между Третьим Рейхом и Галактической Империей. Публикация этого договора могла привести к далеко идущим политическим последствиям, и уж как минимум — к смене поставщика травы в бункер Гитлера.

Согласно договору, космический корабль Рейха «Фау-3» со специально обученной дивизией СС «Darth Vader» и небольшим количеством топлива на борту должен был прибыть в некоторую точку A планетной системы. На каждой планете системы существуют «заправочные станции», контролируемые Империей. Цена топлива на каждой планете при этом различна. Расход топлива на перелёт из одной точки в другую равен округлённому вверх отношению $(S/100)$, где S — расстояние между точками. Конечной точкой путешествия «Фау-3» является планета B .

К сожалению, приложение к документу, касающееся финансовой части договора, до Штирлица не дошло. Так что Центр хочет по заданному количеству топлива на борту, точкам A и B , а также координатам планет системы и ценам на топливо на каждой из них вычислить минимальный бюджет, требуемый для перелёта из точки A к планете B . вычислить,

Input

В первой строке входного файла заданы 6 чисел: X_A, Y_A, Z_A — координаты точки A ($11000 \leq X_A, Y_A, Z_A \leq 1000$), N — количество планет в системе, B — номер планеты, на которую направляется звездолёт ($1 \leq B \leq N \leq 50$) и K — запас топлива на корабле ($1 \leq K \leq 1000$).

Далее идут N строк, каждая из которых содержит по 4 числа: X_i, Y_i, Z_i — координаты i -й планеты ($11000 \leq X_i, Y_i, Z_i \leq 1000$) и C_i — цена единицы топлива на этой планете ($1 \leq C_i \leq 1000$). Все числа во входных файлах целые.

Output

Выведите в выходной файл одно число — минимальное количество денежных единиц, необходимое для достижения планеты B , или 11, если от точки A невозможно долететь до планеты B .

Example

standard input	standard output
0 0 0 4 3 3	6
100 100 0 2	
300 0 0 10	
400 0 0 1	
200 -200 0 3	

Problem E. Numbers

Input file: standard input
Output file: standard output
Time limit: 2 seconds
Memory limit: 64 Mebibytes

Из Центра пришло разрешение на установку «жучков» — миниатюрных подслушивающих устройств — в офисах компании «Рейхспейс». Офис разделён на ячейки и представляет собой квадрат размером 1001×1001 ячейку. Изначально подслушивающее устройство было размещено в центре офиса — в ячейке с координатами $(0,0)$. Каждый день люди Штирлица устанавливают дополнительные подслушивающие устройства. Однако в целях режима экономии за день устройства устанавливаются только в ячейках, имеющих ровно одну общую сторону с некоторой ячейкой, в которую «жучок» был поставлен в один из предыдущих дней..

Ниже приведена схема расстановки «жучков» после первого, второго и третьего дня работ:

```

           0001000
    00100  0011100
010  00100  0101010
111  11111  1111111
010  00100  0101010
           00100  0011100
           0001000
```

По заданному N вычислите количество «жучков», оказавшихся в офисе после N дней работ.

Input

Во входном файле задано одно целое положительное число N — номер дня ($1 \leq N \leq 100$).

Output

В выходной файл выведите количество жучков, которые будут установлены в офисе после N дней работ.

Example

standard input	standard output
1	5
2	9
3	21

Problem F. STAGE-4

Input file: standard input
Output file: standard output
Time limit: 3 seconds
Memory limit: 64 Mebibytes

Для ускорения перевода полученных секретных документов Центр заказал разработку экспериментальной системы машинного перевода STAGE-4. В этой системе языковые конструкции представлены в виде деревьев разбора. Исходная фраза разбивается на несколько семантических блоков порядка 1, каждый из которых, в свою очередь, может состоять из нескольких семантических блоков порядка 2 и так далее. Каждому семантическому блоку сопоставляется его тип — одна из строчных латинских букв *a*, *b*, *c*, *d*, *e*.

Для улучшения работы системы требуется реализовать следующую подзадачу: задано дерево разбора *M* и несколько запросов. Один запрос состоит в том, чтобы для некоторого дерева разбора *A_i* выяснить, насколько максимальный порядок семантического блока в этом дереве отличается от максимального порядка семантического блока в дереве *M*.

Input

В первой строке входного файла задано дерево разбора *M*, состоящее из не менее, чем одной, и не более, чем из 10^4 вершин.

Дерево задаётся следующим способом: для каждого семантического блока, начиная с основной фразы, задан его тип — одна из строчных латинских букв **a**, **b**, **c**, **d**, **e**, затем, если блок разбивается на несколько семантических блоков — задан список таких блоков. Список заключён в круглые скобки, а описания блоков в списке разделены запятой. Так, описание **e(e,d(b,b),a)** обозначает, что основная фраза имеет тип **e** и разбивается на три блока — первый из них имеет тип **e** и не содержит других семантических блоков, второй имеет тип **d** и разбивается на два семантических блока, каждый из которых имеет тип **b** и не содержит других семантических блоков, третий имеет тип **a** и не содержит других семантических блоков.

Во второй строке входного файла указано количество запросов *n* ($1 \leq n \leq 100$).

В каждой из последующих *n* строк входного файла задан один запрос — дерево разбора *A_i*, содержащее не менее одной и не более 16 вершин.

Output

Для *i*-го запроса в отдельной строке выведите разность между максимальным порядком семантического блока в дереве *M* и дереве *A_i*.

Example

standard input	standard output
e(e,d(b,b),a)	1
2	-1
e(b,a)	
e(d,c(a(e,e),b))	