

## Содержание

<b>Новогодние баллы</b>	<b>2</b>
Задача А. King. Король [0.1, 256]	2
Задача В. Ёжики [0.1, 256]	3
Задача С. Ориентируй меня полностью! [2.0, 256]	4
Задача D. План эвакуации [0.1, 256]	5
Задача E. Телефонные номера [0.1, 256]	7
Задача F. Подпалиндромы [0.1, 256]	9
Задача G. Под-бор [0.1, 256]	10
Задача H. Взлом RSA [0.1, 256]	11
Задача I. Квадратное уравнение [0.1, 256]	12
Задача J. Массивы-палиндромы [0.1, 256]	13
Задача K. Длинное деление [0.3, 256]	14
Задача L. Mumatrix [1.5, 256]	15

---

Обратите внимание, входные данные лежат в **стандартном потоке ввода** (он же `stdin`), вывести ответ нужно в **стандартный поток вывода** (он же `stdout`).

В некоторых задачах большой ввод и вывод. Пользуйтесь **быстрым вводом-выводом**.

В некоторых задачах нужен STL, который активно использует динамическую память (set-ы, map-ы) **переопределение стандартного аллокатора** ускорит вашу программу.

Обратите внимание на GNU C++ компиляторы с суффиксом `inc`, они позволяют пользоваться **дополнительной библиотекой**. Под ними можно сдать **вот это**.

## Новогодние баллы

### Задача А. King. Король [0.1, 256]

И было у мужика  
три дочери...

В Тридесятом царстве, Тридевятиом государстве жил-был король. И было у короля  $n$  дочерей. В Тридесятом царстве жили  $n$  прекрасных юношей, и король знал, какие юноши нравятся каждой дочери (поскольку дочери были молодыми и безшабашными, то им могли нравиться несколько юношей одновременно).

Однажды король приказал своему советнику подобрать для каждой дочери прекрасного юношу, за которого та сможет выйти замуж. Советник выполнил приказ и подобрал для каждой дочери для замужества прекрасного юношу, который ей нравился. Разумеется, каждый юноша может жениться только на одной из дочерей.

Посмотрев на список женихов, король сказал: «Мне нравится этот список, но я хочу знать для каждой дочери список всех юношей, за которых она может выйти замуж. Разумеется, при этом все остальные дочери также должны сохранить возможность выйти замуж за юношей, которые им нравятся».

Эта задача оказалась для советника слишком сложной. Помогите ему избежать казни, решив ее.

#### Формат входных данных

Первая строка входного файла содержит число  $n$  — количество дочерей ( $1 \leq n \leq 2000$ ). Следующие  $n$  строк содержат списки прекрасных юношей, которые нравятся дочерям. В начале идет  $k_i$  — количество юношей, которые нравятся  $i$ -ой дочери. Затем идут  $k_i$  целых чисел от 1 до  $n$  — номера юношей. Сумма  $k_i$  не превышает 200 000.

Последняя строка входного файла содержит список, составленный советником —  $n$  различных чисел от 1 до  $n$ : для каждой дочери — номер прекрасного юноши, за которого она может выйти замуж. Гарантируется, что список корректен — то есть каждой девушке нравится выбранный для нее юноша.

#### Формат выходных данных

Выходной файл должен содержать  $n$  строк. Для каждой девушки выведите  $l_i$  — количество различных юношей, за которых она может выйти замуж. После этого выведите  $l_i$  чисел — номера этих юношей в произвольном порядке.

#### Пример

stdin	stdout
4	2 1 2
2 1 2	2 1 2
2 1 2	1 3
2 2 3	1 4
2 3 4	
1 2 3 4	

### Задача В. Ёжики [0.1, 256]

Дан неориентированный граф без петель и кратных ребер, требуется найти в нем:

1. Размер минимального контролирующего множества вершин.
2. Размер максимального независимого множества вершин.
3. Размер минимального глобального разреза.

**Def:** *контролирующее множество* — множество вершин, такое, что у каждого ребра хотя бы один из двух концов лежит в множестве.

**Def:** *независимое множество* — множество вершин, такое, что никакие две вершины множества не соединены ребром.

**Def:** *глобальный разрез* — разбиение множества всех вершин  $V$  на непустые  $S$  и  $T$  ( $S \cup T = V, S \cap T = \emptyset$ ). Величиной разреза называется количество ребер между  $S$  и  $T$ .

#### Формат входных данных

В первой строке файла записаны два числа —  $n, m$  (количество вершин и количество ребер).

В следующих  $m$  строках записаны пары чисел. Пара чисел  $xy$  означает, что есть ребро между вершинами  $x$  и  $y$ .

Ограничения:  $2 \leq n \leq 15$ .

#### Формат выходных данных

Выведите 3 числа — ответы для трех задач.

#### Пример

stdin	stdout
4 4 1 2 2 3 3 1 3 4	2 2 1

### Задача С. Ориентировать меня полностью! [2.0, 256]

Вам дан неориентированный граф без петель и кратных рёбер. Ваша задача — ориентировать граф таким образом, чтобы максимальная исходящая степень была бы минимально возможной.

#### Формат входных данных

В первой строке заданы числа  $n$  и  $m$  — количество вершин и рёбер в графе ( $1 \leq n \leq 25\,000; 0 \leq m \leq 25\,000$ ). В следующих  $m$  строках даны пары чисел от 1 до  $n$  — рёбра графа.

#### Формат выходных данных

Выведите минимально возможную максимальную степень. Далее выведите  $m$  целых чисел от 0 до 1. Если  $i$ -е ребро было задано парой чисел  $a, b$ , то ноль означает, что оно после ориентации ведёт из  $a$  в  $b$ , а единица — что из  $b$  в  $a$ .

#### Примеры

stdin	stdout
4 4 1 2 1 3 4 2 4 3	1 0 1 1 0
5 5 1 2 2 3 3 1 1 4 1 5	1 0 0 0 1 1

#### Подсказка по решению

Если будете сдавать потоком, примените к обычному Форд-Фалкерсону такую же оптимизацию, как в Куне «вообще не обнулять пометки» и получите ОК.

### Задача D. План эвакуации [0.1, 256]

В городе есть муниципальные здания и бомбоубежища, которые были специально построены для эвакуации служащих в случае ядерной войны. Каждое бомбоубежище имеет ограниченную вместительность по количеству людей, которые могут в нем находиться. В идеале все работники из одного муниципального здания должны были бы бежать к ближайшему бомбоубежищу. Однако, в таком случае, некоторые бомбоубежища могли бы переполниться, в то время как остальные остались бы наполовину пустыми.

Чтобы разрешить эту проблему Городской Совет разработал специальный план эвакуации. Вместо того, чтобы каждому служащему индивидуально приписать, в какое бомбоубежище он должен бежать, для каждого муниципального здания определили, сколько служащих из него в какое бомбоубежище должны бежать. Задача индивидуального распределения была переложена на внутреннее управление муниципальных зданий.

План эвакуации учитывает количество служащих в каждом здании — каждый служащий должен быть учтен в плане и в каждое бомбоубежище может быть направлено количество служащих, не превосходящее вместимости бомбоубежища.

Городской Совет заявляет, что их план эвакуации оптимален в том смысле, что суммарное время эвакуации всех служащих города минимально.

Мэр города, находящийся в постоянной конфронтации с Городским Советом, не слишком то верит этому заявлению. Поэтому он нанял Вас в качестве независимого эксперта для проверки плана эвакуации. Ваша задача состоит в том, чтобы либо убедиться в оптимальности плана Городского Совета, либо доказать обратное, представив в качестве доказательства другой план эвакуации с меньшим суммарным временем для эвакуации всех служащих.

Карта города может быть представлена в виде квадратной сетки. Расположение муниципальных зданий и бомбоубежищ задается парой целых чисел, а время эвакуации из муниципального здания с координатами  $(X_i, Y_i)$  в бомбоубежище с координатами  $(P_j, Q_j)$  составляет  $D_{ij} = |X_i - P_j| + |Y_i - Q_j| + 1$  минут.

#### Формат входных данных

Входной файл содержит описание карты города и плана эвакуации, предложенного Городским Советом. Первая строка входного файла содержит два целых числа  $N$  ( $1 \leq N \leq 100$ ) и  $M$  ( $1 \leq M \leq 100$ ), разделенных пробелом.  $N$  — число муниципальных зданий в городе (все они занумерованы числами от 1 до  $N$ ),  $M$  — число бомбоубежищ (все они занумерованы числами от 1 до  $M$ ).

Последующие  $N$  строк содержат описания муниципальных зданий. Каждая строка содержит целые числа  $X_i, Y_i$  и  $B_i$ , разделенные пробелами, где  $X_i, Y_i$  ( $-1000 \leq X_i, Y_i \leq 1000$ ) — координаты здания, а  $B_i$  ( $1 \leq B_i \leq 1000$ ) — число служащих в здании.

Описание бомбоубежищ содержится в последующих  $M$  строках. Каждая строка содержит целые числа  $P_j, Q_j$  и  $C_j$ , разделенные пробелами, где  $P_j, Q_j$  ( $-1000 \leq P_j, Q_j \leq 1000$ ) — координаты бомбоубежища, а  $C_j$  ( $1 \leq C_j \leq 1000$ ) — вместимость бомбоубежища.

В последующих  $N$  строках содержится описание плана эвакуации. Каждая строка представляет собой описание плана эвакуации для отдельного здания. План эвакуации из  $i$ -го здания состоит из  $M$  целых чисел  $E_{ij}$ , разделенных пробелами.  $E_{ij}$  ( $0 \leq E_{ij} \leq 10\,000$ ) — количество служащих, которые должны эвакуироваться из  $i$ -го здания в  $j$ -е бомбоубежище.

Гарантируется, что план, заданный во входном файле, корректен.

#### Формат выходных данных

Если план эвакуации Городского Совета оптимален, то выведите одно слово OPTIMAL. В противном случае выведите на первой строке слово SUBOPTIMAL, а в последующих  $N$  строках

выведите Ваш план эвакуации (более оптимальный) в том же формате, что и во входном файле. Ваш план не обязан быть оптимальным, но должен быть лучше плана Городского Совета.

**Пример**

stdin	stdout
3 4 -3 3 5 -2 2 6 2 2 5 -1 1 3 1 1 4 -2 -2 7 0 -1 3 3 1 1 0 0 0 6 0 0 3 0 2	SUBOPTIMAL 3 0 1 1 0 0 6 0 0 4 0 1
3 4 -3 3 5 -2 2 6 2 2 5 -1 1 3 1 1 4 -2 -2 7 0 -1 3 3 0 1 1 0 0 6 0 0 4 0 1	OPTIMAL

### Задача Е. Телефонные номера [0.1, 256]

В современном мире вы встречаетесь с огромным количеством телефонных номеров, которые со временем становятся всё длиннее и длиннее. И вам приходится запоминать эти номера. Одним из простых способов запоминания является сопоставление букв каждой цифре, как показано на следующем рисунке:

1 ij	2 abc	3 def
4 gh	5 kl	6 mn
7 prs	8 tuv	9 wxu
	0 oqz	

Таким образом, каждому слову или группе слов может быть сопоставлен уникальный номер, так что можно запоминать слова вместо телефонных номеров. Очевидно, есть особый шарм в том, чтобы найти простую взаимосвязь между словом, используемым для запоминания телефонного номера, и владельцем этого номера. Так, телефонный номер 941837296 вашего друга, играющего в шахматы, может быть прочитан как WHITEPAWN (белая пешка), а номер 2855304 Вашего любимого учителя может быть прочитан как BULLDOG (бульдог). Напишите программу, находящую самую короткую последовательность слов (имеющую наименьшее количество слов), которая соответствует заданному номеру телефона и заданному списку слов. Соответствие описано на рисунке выше.

#### Формат входных данных

Ввод состоит из набора тестов. Первая строка каждого теста содержит номер телефона, к которому нужно подобрать мнемонику. Номер состоит не более чем из 100 цифр. Вторая строка содержит общее количество слов в словаре (максимум 50 000). Каждая из оставшихся строк содержит одно слово, состоящее не более чем из 50 строчных латинских букв. Общий размер ввода не превосходит 300 килобайт. Последняя строка ввода содержит число  $-1$ .

#### Формат выходных данных

Каждая строка вывода должна содержать кратчайшую последовательность слов, найденную вашей программой. Слова должны быть разделены одиночными пробелами. Если для входных данных нет решения, соответствующая строка вывода должна содержать текст "No solution.". Если существует несколько решений, имеющих одинаковое количество слов, можете выбрать любое из них.

**Примеры**

stdin	stdout
7325189087	reality our
5	No solution.
it	
your	
reality	
real	
our	
4294967296	
5	
it	
your	
reality	
real	
our	
-1	

**Подсказка по решению**

Бор.

### Задача F. Подпалиндромы [0.1, 256]

Дано слово и запросы двух типов:

- заменить  $i$ -ю букву в слове на букву  $c$ ;
- проверить, является ли подстрока  $s_j \dots s_k$  палиндромом.

#### Формат входных данных

В первой строке записано слово из  $n$  строчных латинских букв. Во второй строке записано целое число  $m$  — количество запросов ( $5 \leq n, m \leq 10^5$ ). Следующие  $m$  строк содержат запросы. Каждый запрос имеет вид «change  $i$   $a$ » или «palindrome?  $j$   $k$ », где  $i, j, k$  — целые числа ( $1 \leq i \leq n; 1 \leq j \leq k \leq n$ ), а символ  $c$  — строчная латинская буква.

#### Формат выходных данных

На все запросы второго типа выведите «Yes», если подслово  $s_j \dots s_k$  является палиндромом, и «No» в противном случае.

#### Примеры

stdin	stdout
abcda	No
5	Yes
palindrome? 1 5	Yes
palindrome? 1 1	Yes
change 4 b	
palindrome? 1 5	
palindrome? 2 4	

### Задача G. Под-бор [0.1, 256]

*Бором* называется подвешенное дерево, на каждом из рёбер которого написано по символу, причём символы, написанные на рёбрах, выходящих из общей вершины-родителя, различны. Будем называть направление от родителя к детям “вниз”. Назовем *вхождением строки  $s$  в бор* такую вершину бора, от которой можно пройти несколько шагов вниз таким образом, что встретившиеся символы образуют строку  $s$ .

Даны бор и несколько строк, найдите сумму количеств вхождений этих строк в этот бор.

#### Формат входных данных

В первой строке входного файла записано единственное число  $n$ ,  $1 \leq n \leq 100\,000$  — количество вершин бора. В следующих  $n$  строках описаны вершины бора. В  $(i + 1)$ -й строке описаны дети  $i$ -й вершины: число  $k_i$  ее детей, затем  $k_i$  пар из номера вершины-ребёнка и символа, написанного на соответствующем ребре. Номер родителя всегда меньше номера ребёнка; корнем бора является вершина номер 1.

В  $(n + 2)$ -й строке записано количество  $m$  ( $1 \leq m \leq 100\,000$ ) строк для поиска. В следующих  $m$  строках перечислены сами строки. Входные строки непусты, а их суммарная длина не превышает 100 000 символов.

Все символы, написанные на рёбрах, а также все символы, составляющие строки — маленькие латинские буквы.

#### Формат выходных данных

Выведите одно число — сумму количеств вхождений.

#### Пример

stdin	stdout
7	9
2 2 a 4 b	
2 3 a 6 b	
0	
1 5 b	
1 7 b	
0	
0	
4	
b	
bb	
bbb	
bb	

### Задача Н. Взлом RSA [0.1, 256]

Проблема взлома RSA состоит в следующем. Дано положительное целое число  $n$ , которое является произведением двух различных простых нечетных чисел  $p$  и  $q$ , положительное целое  $e$  такое, что  $\text{НОД}(e, (p-1)(q-1)) = 1$ , а также целое  $c$ .

Необходимо найти такое положительное целое  $m$  что  $m^e \equiv c \pmod n$ .

Пара  $(n, e)$  – открытый ключ,  $m$  – исходное сообщение,  $c = m^e$  – зашифрованная версия.

#### Формат входных данных

В первой строке находится одно число  $K$  ( $K \leq 2000$ ) – количество тестов.

Каждая следующая строка представляет собой отдельный тест, который содержит три числа –  $e$ ,  $n$  и  $c$  ( $0 \leq e, n, c \leq 32000$ ;  $1 \leq e$ ;  $6 \leq n$ ,  $n = pq$ ;  $p, q$  – различные нечётные простые, кроме того  $\text{НОД}(e, (p-1)(q-1)) = 1$ ,  $e < (p-1)(q-1)$ ).

#### Формат выходных данных

Для каждого теста выведите  $m$ .

#### Примеры

stdin	stdout
3	7
9 187 129	23
11 221 56	17
7 391 204	

### Задача I. Квадратное уравнение [0.1, 256]

Детей в школе учат решать квадратные уравнения, т.е. уравнения вида

$$ax^2 + bx + c = 0,$$

где  $a$ ,  $b$  и  $c$  некоторые заданные действительные числа, а  $x$  — действительное число, которое необходимо найти.

В этой задаче вам потребуется решить квадратное уравнение для многочленов с коэффициентами из нулей или единиц, и все операции производятся по модулю 2.

Даны многочлены  $a(t)$ ,  $b(t)$  и  $c(t)$ , найдите такой полином  $x(t)$  что

$$a(t)x^2(t) + b(t)x(t) + c(t) = 0,$$

где равенство понимается как равенство многочленов. Напомним, что многочлены равны тогда и только тогда, когда равны их коэффициенты при соответствующих степенях  $t$ .

#### Формат входных данных

Входной файл содержит многочлены  $a(t)$ ,  $b(t)$  и  $c(t)$ , которые задаются их степенями, за которыми следуют коэффициенты, начиная со старшего. Нулевые многочлены в данной задаче имеют степень  $-1$ . Степени всех многочленов не превосходят 127. Между старшим коэффициентом и степенью находится **два** пробела. После многочлена степени  $-1$  также находится один пробел.

#### Формат выходных данных

Если есть хотя бы одно решение уравнения, выведите любое из них в таком же формате. Старший коэффициент найденного многочлена не должен быть нулевым. Степень полинома не должна превышать 512.

В противном случае напечатайте “no solution”.

#### Примеры

stdin	stdout
0 1	1 1 0
2 1 1 0	
3 1 0 0 0	

### Задача J. Массивы-палиндромы [0.1, 256]

Даны  $n$  целых чисел от 1 до 1 000. Числа могут повторяться. В сколько из  $2^n - 1$  непустых подмножеств этих  $n$  чисел произведение является точными квадратом?

#### Формат входных данных

На первой строке  $n$  ( $3 \leq n \leq 1\,000$ ).

На второй строке  $n$  целых чисел от 1 до 1 000.

#### Формат выходных данных

Выведите одно число – количество непустых подмножеств по модулю  $10^9 + 7$ .

#### Пример

stdin	stdout
6 1 2 3 4 5 6	7
4 2 2 2 2	7

**Задача К. Длинное деление [0.3, 256]**

Даны два целых положительных числа. Найти частное и остаток от деления.

**Формат входных данных**

Мультитест. Каждый тест задаётся двумя строками. Суммарная длина чисел до 50 000.

**Формат выходных данных**

Для каждого теста выведите две строки – частное и остаток

**Примеры**

stdin	stdout
10	3
3	1
17	0
100	17
100	5
17	15

**Подсказка по решению**

Решение за  $\mathcal{O}\left(\frac{n^2}{k^2}\right)$  зайдёт по времени. Система счисления равна  $10^k$ .

### Задача L. Mumatrix [1.5, 256]

Дана бинарная матрица. Возвести ее в квадрат по модулю 2.

#### Формат входных данных

Во входном файле число  $n$  ( $1 \leq n \leq 4000$ ). Далее  $n$  строк.

В каждой  $n$  символов. Каждый — или 0, или 1.

#### Формат выходных данных

В выходной одно число — количество единиц в матрице-квадрате-исходной.

#### Пример

stdin	stdout
3 011 100 101	5