

Задача А. Как разрушить свой мозг

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Победителей не судят. Их приглашают в жюри

девиз Воронежской олимпиады

Пока Сквив, Гвидо и Нунцио регистрировались в качестве участников, к Аазу подошёл представитель организаторов и спросил:

- Вы приехали как тренер?
- Да, а разве Вам не пришла заявка? Команда Possiltum U M.Y.T.H. . . .
- Конечно пришла. Просто у нас тут возникла идея. Раз Вы, во-первых, тренер, а во-вторых, как я вижу, извращенец, то мы приглашаем Вас принять участие в работе жюри.
- Значит так. Во-первых, я не извращенец, а изверг. . . — сурово уточнил Ааз.
- Тем более, у нас в жюри как раз не хватает сторонников строгого подхода. Так что давайте, жюри Вас ждёт.

Ааз решил, что лишняя информация о ходе соревнований ему не повредит, и принял приглашение, предварительно осведомившись о вознаграждении, положенном членам жюри за работу.

Вчитываясь в условие очередной задачи, как раз и посвящённой вычитыванию условий задач, Ааз почувствовал, что этот текст начинает разрушать его мозг. Он вспомнил, что изверг может разрушить свой мозг чтением строки S , если хотя бы половина из всех её k -буквенных подпоследовательностей совпадает.

Чтобы уберечь мозг ценного сотрудника корпорации «МИФ», напишите программу, которая по заданной строке S выясняет, может ли Ааз разрушить свой мозг, читая эту строку.

Формат входных данных

В первой строке дана непустая строка S , длиной до 1 000 маленьких букв латинского алфавита. Во второй — натуральное число $1 \leq k \leq |S|$.

Формат выходных данных

Выведите «YES», если строка может разрушить мозг изверга, и «NO» иначе.

Примеры

стандартный ввод	стандартный вывод
aaaaaaa 3	YES
abc 2	NO

Задача В. SO-SAT

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Найдите решение 3-SAT. Гарантируется, что оно существует.

Формулировка 3-SAT: нужно подобрать значения n булевых переменных так, чтобы все m утверждений вида $x_{i_1} = e_1 \vee x_{i_2} = e_2 \vee x_{i_3} = e_3$ обратились в истину.

Формат входных данных

На первой строке число переменных n и число утверждений m ($1 \leq n \leq 90, 1 \leq m \leq \min(n^2, 1000)$).

Каждая из следующих m строк содержит числа $i_1, e_1, i_2, e_2, i_3, e_3$ и задает утверждение $x_{i_1} = e_1 \vee x_{i_2} = e_2 \vee x_{i_3} = e_3$.

Все тесты случайны, тем не менее гарантируется, что решение существует.

Формат выходных данных

Выведите строку из n нулей и единиц — значения переменных.

Если у данной задачи 3-SAT есть несколько решений, выведите любое.

Пример

стандартный ввод	стандартный вывод
2 3 1 0 1 0 1 0 2 0 2 1 1 1 1 1 2 1 1 1	01

Задача С. Трисочетание

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 0.5 секунд
Ограничение по памяти: 256 мегабайт

Дан трёхдольный полный взвешенный граф. Все три доли имеют одинаковый размер n . Трёхдольный означает, что нет ребер внутри доли. Полный означает, что между долями проведены все возможные ребра, каждое ребро ровно один раз. Нужно найти трисочетание минимального веса. Трисочетание — n троек индексов (a_i, b_i, c_i) такие, что все a_i различны, все b_i различны, все c_i различны. Вес трисочетания равен $\sum_{i=1..n} (w[0, a_i, b_i] + w[1, b_i, c_i] + w[2, c_i, a_i])$, где $w[i, a, b]$ — вес ребра из a -й вершины i -й доли в b -ю вершину $(i + 1) \bmod 3$ доли.

Формат входных данных

На первой строке число $n \geq 1$.

Следующие n строк содержат матрицу $w[0]$.

Следующие n строк содержат матрицу $w[1]$.

Следующие n строк содержат матрицу $w[2]$.

Гарантируется, что все веса — случайные числа от k до $2k$ для некоторого k . Все веса — целые числа от 0 до 10^5 .

Формат выходных данных

На первой строке выведите суммарный вес найденного трисочетания. На каждой из следующих n строк выведите $a_i b_i c_i$ (вершины нумеруются от 0 до $n - 1$). Тройки можно выводить в любом порядке. Если трисочетаний с минимальным суммарным весом несколько, подойдет любое.

Система оценки

Подзадача 1 (25 баллов) $n \leq 5$.

Подзадача 2 (25 баллов) $n \leq 9$.

Подзадача 3 (25 баллов) $n \leq 11$.

Подзадача 4 (25 баллов) $n \leq 13$.

Пример

стандартный ввод	стандартный вывод
3	330
64 100 96	0 2 0
90 43 50	1 1 1
12 94 23	2 0 2
75 97 45	
84 19 45	
11 60 28	
9 16 79	
49 21 54	
85 2 74	

Задача D. Дорога, дорога, осталось немного...

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 0.5 секунд
Ограничение по памяти: 256 мегабайт

Дан случайный неориентированный граф G из n вершин и m ребер. Ваша задача — найти гамильтонов путь. Гарантируется, что гамильтонов путь в графе есть.

Формат входных данных

На первой строке число вершин $n \geq 2$ и число ребер $m \geq 1$.

Следующие m строк содержат пары чисел от 1 до n — ребра графа.

В графе нет ни петель, ни кратных ребер.

Поскольку почти полный граф — совсем не интересный тест, $m \leq 500$.

Формат выходных данных

На первой строке выведите n различных чисел от 1 до n — вершины гамильтонового пути в порядке прохода по ним. Начинать и заканчивать можно в любой вершине. Если гамильтоновых путей несколько, выведите любой.

Система оценки

Подзадача 1 (20 баллов) $n \leq 26$.

Подзадача 2 (20 баллов) $n \leq 35$.

Подзадача 3 (20 баллов) $n \leq 50$.

Подзадача 4 (20 баллов) $n \leq 70$.

Подзадача 5 (20 баллов) $n \leq 100$.

Пример

стандартный ввод	стандартный вывод
5 8 3 1 2 5 5 4 3 4 1 4 3 5 3 2 1 2	1 4 3 5 2

Задача Е. Видеонаблюдение

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	0.3 секунд
Ограничение по памяти:	256 мегабайт

Город-Таз известен нереально высоким уровнем криминала. Полиция не видит вариантов развития ситуации, кроме как усилить меры безопасности. Они хотят установить движущихся дронов-беспилотников на некоторых перекрёстках города, чтобы знать, кто проезжает перекрёстки на красный свет. Если машина проедет на красный свет, дрон погонится за ней и остановит машину, чтобы вручить водителю соответствующий талон. Дроны-беспилотники довольно глупы, поэтому останавливаются обязательно до следующего перекрёстка, иначе они рискуют заблудиться и потерять дорогу домой. Домом дрона считается перекрёсток, к которому тот прикреплен. Дроны не умеют определять присутствие других дронов (отличать их от машин-нарушителей), поэтому полицейский технический департамент принял решение, что никакие два дрона нельзя прикреплять к соседним перекрёсткам. Как и в большинстве городов, в ГОРОДЕ-ТАЗЕ нет перекрёстков с более чем четырьмя соседними перекрёстками.

Дроны выделяются государством (бесплатно!), поэтому полиция хочет получить настолько много дронов, насколько это возможно. Вас попросили определить, возможно ли в городе расположить заданное число дронов, не нарушая правило, что ни на каких двух соседних перекрёстках не должно одновременно быть дронов.

Формат входных данных

Первая строка содержит число k ($0 \leq k \leq 15$) – количество дронов, которых нужно разместить в городе. На второй строке n ($1 \leq n \leq 100\,000$) – число перекрёстков в ГОРОДЕ-ТАЗЕ. Следующие n строк описывают перекрёстки в i -й строке сперва дано число d ($0 \leq d \leq 4$), количество соседних перекрёстков с i -м, затем d номеров соседних перекрёстков. Все эти d чисел различны и отличны от i . Отношение “соседние перекрёстки” симметрично – если i соседний с j , то и j соседний с i . Номера перекрёстков – числа от 1 до n .

Формат выходных данных

На первой строке `possible` или `impossible`. Если ответ – `possible`, выведите на второй строке k целых чисел от 1 до n , номера вершин.

Система оценки

Потестовая оценка.

Примеры

стандартный ввод	стандартный вывод
4 7 2 2 4 3 1 3 5 1 2 2 1 5 4 2 6 4 7 2 5 7 2 6 5	impossible
4 8 2 2 4 3 1 3 5 1 2 2 1 5 4 2 6 4 7 2 5 8 2 8 5 2 7 6	possible 5 3 1 8

Задача F. Множество множеств

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 0.5 секунд
Ограничение по памяти: 256 мегабайт

Даны несколько случайных подмножеств множества $A = \{1, 2, \dots, n\}$. Нужно выбрать минимальное количество подмножеств, которые в объединении дают A . Гарантируется, что объединение всех подмножеств равно A .

Формат входных данных

На первой строке число $n \geq 1$ и число множеств $m \geq 1$. Следующие m строк задают множества. Каждое множество задается числом элементов $k_i \geq 5$ и k_i различными числами от 1 до n . Гарантируется, что множества генерировались следующим образом: фиксируем c , далее m раз сперва выбираем случайный размер s от 1 до c , далее равновероятно из всех C_n^s вариантов выбираем множество размера s .

Формат выходных данных

На первой строке выведите x — минимальное количество множеств. На следующей строке выведите номера выбранных множеств, x различных чисел от 1 до m . Номера можно выводить в произвольном порядке. Если оптимальных ответов несколько, выведите любой.

Система оценки

- Подзадача 1 (25 баллов) $n \leq 20, m \leq 50$.
Подзадача 2 (25 баллов) $n \leq 30, m \leq 100$.
Подзадача 3 (25 баллов) $n \leq 40, m \leq 100$.
Подзадача 4 (25 баллов) $n \leq 50, m \leq 100$.

Пример

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

Задача G. Проще всего красить в три цвета

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Дан случайный неориентированный граф G из n вершин и m ребер. Ваша задача — покрасить его вершины в три цвета таким образом, чтобы смежные вершины были покрашены в разные цвета. Гарантируется, что покрасить граф в три цвета возможно.

Формат входных данных

На первой строке число вершин n ($1 \leq n \leq 250$) и число ребер $m \geq 1$.

Следующие m строк содержат пары чисел от 1 до n — ребра графа.

В графе нет ни петель, ни кратных ребер.

Формат выходных данных

На следующей строке n целых чисел от 1 до 3 — цвета вершин. Если требуемых раскрасок несколько, выведите любую.

Пример

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

Задача Н. Макс клика

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 0.25 секунд
Ограничение по памяти: 256 мегабайт

Макс, опытейший игрок в доту, постоянно кликал.

Дан случайный неориентированный граф G из n вершин и m ребер.
Подкликой называется такое подмножество вершин $A: \forall a, b \in A, a \neq b \quad \exists$ ребро (a, b) .
Ваша задача — найти подклику $A: |A|$ максимально.

Формат входных данных

На первой строке число вершин $n \geq 1$ и число ребер $m \geq 1$.
Следующие m строк содержат пары чисел от 1 до n — ребра графа.
В графе нет ни петель, ни кратных ребер.

Формат выходных данных

На первой строке выведите k — количество вершин в максимальной подклике. На следующей строке k целых чисел от 1 до n — номера вершин в подклике. Вершины можно выводить в любом порядке. Если максимальных подклик несколько, выведите любую.

Система оценки

Подзадача 1 (10 баллов) $n \leq 20$.
Подзадача 2 (10 баллов) $n \leq 25$.
Подзадача 3 (20 баллов) $n \leq 40$.
Подзадача 4 (20 баллов) $n \leq 60$.
Подзадача 5 (20 баллов) $n \leq 70$.
Подзадача 6 (20 баллов) $n \leq 80$.

Пример

стандартный ввод	стандартный вывод
5 8	4
5 4	3 5 1 4
3 5	
1 5	
1 3	
2 3	
1 4	
5 2	
3 4	

Задача I. Греко-латинские квадраты

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Греко-латинский квадрат, или эйлеров квадрат, — квадрат $n \times n$ в каждой клетке которого стоят 2 числа от 1 до n так, что выполняются следующие условия:

1. В каждой строке и столбце каждая цифра встречается один раз на первом месте в паре, и один раз на втором.
2. Каждая цифра стоит в паре с каждой другой цифрой и с самой собой по одному разу.

Формат входных данных

В первой строке содержится одно целое число n ($n \in \{4, 7, 8, 9\}$) — размер квадрата

Формат выходных данных

Выведите n^2 пар чисел — греко-латинский квадрат размера n .

Пример

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

Задача J. Китайские кроссворды

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Решите китайский кроссворд отсюда: <https://1.aismagilov.ru/chinese>

Формат входных данных

В первой строке содержится одно целое число t ($0 \leq t \leq 4$) — номер теста.

Формат выходных данных

Выведите решение, состоящее из символов “X” и “.”.

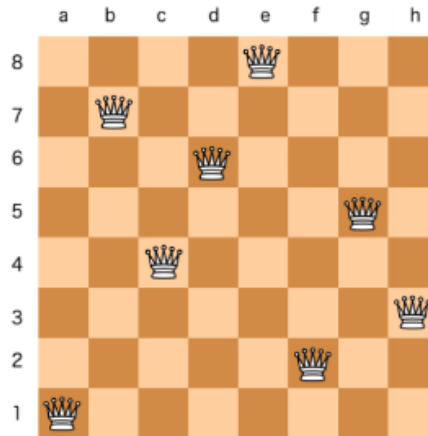
Пример

стандартный ввод	стандартный вывод
0	.XX.. .XX.X ..X.X .XXX. X.X.. X.X.. ..XX. .X.X. .X.XX XX...

Задача К. Королевская битва

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Расставьте на доске размера $n \times n$ ровно n не бьющих друга ферзей. (Два ферзя бьют друг друга, если они стоят на одной вертикали, горизонтали или диагонали.)



Эту задачу надо решить только для $n = 8, 20, 80, 90, 100$.

Формат входных данных

В первой строке содержится одно целое число n , $n \in \{8, 20, 80, 90, 100\}$ — размер доски.

Формат выходных данных

Выведите n целых чисел a_1, a_2, \dots, a_n : для каждой строки i — столбец в котором находится ферзь ($1 \leq a_i \leq n$).

Пример

стандартный ввод	стандартный вывод
8	5 2 4 7 3 8 6 1

Задача L. Судуку

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Решите судуку. В этой задаче несколько тестов:

0.

4			2			
5			7 3			1
	3 9	1 5 6				
7 1 3					8	
	2	6 8 7 3				
	6			9 2 7		
		9 4 1 8 5				
6		7 2				9
		6				2

1.

4			2 9	8 3		
				1		
5			6 8			
1	4	7				
	2				8	
			3 9	1		
		8 7			3	
	6					
7 9	1 3					2

2.

			6			
	2			9 5 3		
8 1		2 4				
			5	2 9		
	4				5	
7	9		6			
			3 1	9 8		
1	8 4			3		
			7			

3.

4 2				9 3		
	3		8			
	5		2		4	
9		5 4				
1						9
		6 1			8	
5		2		6		
		1		3		
8	6				9 2	

Формат входных данных

В первой строке содержится одно целое число t ($0 \leq t \leq 3$) — номер теста.

Дальше идёт 9 строк по 9 чисел — ячейки sudoku. Если ячейка пустая, то там содержится ноль.

Формат выходных данных

Выведите 9 строк по 9 целых чисел — заполненное sudoku.

Пример

стандартный ввод	стандартный вывод
0	4 7 1 8 2 9 5 6 3
4 0 0 0 2 0 0 0 0	5 8 6 7 3 4 2 9 1
5 0 0 7 3 0 0 0 1	2 3 9 1 5 6 4 7 8
0 3 9 1 5 6 0 0 0	7 1 3 2 9 5 6 8 4
7 1 3 0 0 0 0 8 0	9 4 2 6 8 7 3 1 5
0 0 2 6 8 7 3 0 0	8 6 5 4 1 3 9 2 7
0 6 0 0 0 0 9 2 7	3 2 7 9 4 1 8 5 6
0 0 0 9 4 1 8 5 0	6 5 8 3 7 2 1 4 9
6 0 0 0 7 2 0 0 9	1 9 4 5 6 8 7 3 2
0 0 0 0 6 0 0 0 2	