

## Задача А. Гамильтонов цикл в полном графе

Имя входного файла: fullham.in  
Имя выходного файла: fullham.out  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 256 мегабайт

Дан граф из  $N$  вершин, в котором степень любой вершины не меньше  $\frac{N}{2}$ . Ваша задача — найти гамильтонов цикл.

### Формат входных данных

На первой строке входного файла записано целое число  $N$  ( $3 \leq N \leq 4000$ ) — количество вершин в графе. На следующих  $N$  строках записана матрица смежности. Т.к. матрица смежности симметрична, а на диагонали всегда стоят нули, на  $i$ -й строке записаны  $i - 1$  символ — нули и единицы. Если  $j$ -й символ  $i$ -й строки равен единице, значит есть ребро между вершинами  $i$  и  $j$ .

Гарантируется, что в графе есть гамильтонов цикл и, что степень каждой вершины не меньше  $\frac{N}{2}$ .

### Формат выходных данных

Выведите перестановку из  $N$  чисел — номера вершин в порядке гамильтонова цикла.

### Примеры

fullham.in	fullham.out
4	1 2 3 4
1	
11	
101	

## Задача В. Поиск гамильтонова цикла в условиях теоремы Хватала

Имя входного файла: `chvatal.in`  
Имя выходного файла: `chvatal.out`  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 256 мегабайт

Дан граф из  $N$  вершин, для которого выполняется условие теоремы Хватала, то есть, в отсортированной последовательности его степеней вершин  $d_k$  для любого  $k < n/2$  верно либо  $d_k > k$ , либо  $d_{n-k} \geq n - k$ . Ваша задача — найти гамильтонов цикл.

### Формат входных данных

На первой строке входного файла записано целое число  $N$  ( $3 \leq N \leq 100$ ) — количество вершин в графе. На следующих  $N$  строках записана матрица смежности. Т.к. матрица смежности симметрична, а на диагонали всегда стоят нули, на  $i$ -й строке записаны  $i - 1$  символ — нули и единицы. Если  $j$ -й символ  $i$ -й строки равен единице, значит есть ребро между вершинами  $i$  и  $j$ .

### Формат выходных данных

Выведите перестановку из  $N$  чисел — номера вершин в порядке гамильтонова цикла.

### Примеры

<code>chvatal.in</code>	<code>chvatal.out</code>
4	1 2 3 4
1	
11	
101	

## Задача С. Интерактивная восточная сказка

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

У злого волшебника Джафара много ламп, которые он холит и лелеет, и любит очень сильно, но из каждой пары ламп он всё же может выбрать одну, которую он любит даже сильнее, чем другую.

Он захотел расставить их в ряд так, чтобы когда он будет идти вдоль этого ряда каждая следующая лампа была им более любима, чем предыдущая.

Новому слуге Джафара поручено это сделать, но... он не знает предпочтений Джафара!

Про любую пару ламп можно спросить у волшебника, какую он любит больше, но нельзя излишне навязываться с вопросами (отрубание головы еще никто не отменял).

Помогите слуге расположить лампы или узнать, что это невозможно (и сброситься со скалы).

### Формат входных данных

Первое число, которое будет передано во входном потоке, —  $N$  ( $1 \leq N \leq 1000$ ), количество ламп.

Затем на каждый вопрос слуги, который ваша программа выведет в выходной поток, во входном потоке будет дан ответ — слово “YES”, если лампа  $Y_i$  более любима чем  $X_i$ , и слово “NO”, если  $X_i$  более любима чем  $Y_i$ .

Заметьте, что отношение «более любима чем» не обязано быть транзитивным.

### Формат выходных данных

В выходной файл вы можете выводить запросы. Каждый вопрос — одна строка с тремя числами  $1, X_i, Y_i$  ( $1 \leq X_i, Y_i \leq N; X_i \neq Y_i$ ). Вы можете задать не более 10 000 вопросов.

В последней строке выведите число 0, а затем  $N$  целых чисел от 1 до  $N$  — номера ламп в порядке, в котором их надо расставить. Если же расставить лампы невозможно, выведите  $(N + 1)$  ноль.

### Примеры

стандартный ввод	стандартный вывод
3	1 1 2
YES	1 1 3
NO	0 3 1 2

### Замечание

Обязательно сбрасывайте буфер при выводе запросы: в Pascal используйте `flush(output);`, в C используйте `fflush(stdout);`, в C++ используйте `cout.flush();`, в Java используйте `System.out.flush();`.

## Задача D. Цикл в турнире

Имя входного файла: `guyaury.in`  
Имя выходного файла: `guyaury.out`  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 256 мегабайт

Найдите гамильтонов цикл в полном ориентированном сильносвязном графе.

### Формат входных данных

В первой строке входного файла содержится число  $n$  ( $1 \leq n \leq 1000$ ) — число вершин в графе.

Далее следует  $n$  строк имеющих длину, соответственно,  $0, 1, 2, \dots, n - 1$ . В  $i$ -й из этих строк  $j$ -й символ задает равняется 1, если ребро ведет из вершины  $i$  в вершину  $j$ , и 0, если из вершины  $j$  в вершину  $i$ .

### Формат выходных данных

В выходной файл выведите номера вершин в порядке их следования в найденном гамильтоновом цикле.

### Примеры

<code>guyaury.in</code>	<code>guyaury.out</code>
3	1 3 2
1	
01	