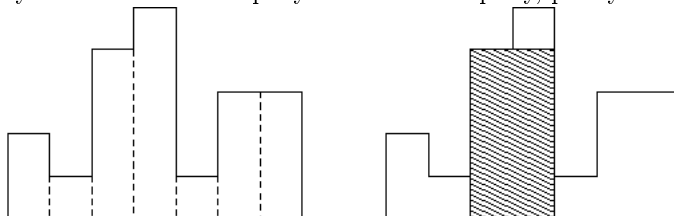


## Задача А. Гистограмма

Ограничение по времени: 2 секунды  
Ограничение по памяти: 64 Мб

Гистограмма является многоугольником, сформированным из последовательности прямоугольников, выровненных на общей базовой линии. Прямоугольники имеют равную ширину, но могут иметь различные высоты. Например, фигура слева показывает гистограмму, которая состоит из прямоугольников с высотами 2, 1, 4, 5, 1, 3, 3. Все прямоугольники на этом рисунке имеют ширину, равную 1.



Обычно гистограммы используются для представления дискретных распределений, например, частоты символов в текстах. Отметьте, что порядок прямоугольников очень важен. Вычислите область самого большого прямоугольника в гистограмме, который также находится на общей базовой линии. На рисунке справа заштрихованная фигура является самым большим выровненным прямоугольником на изображенной гистограмме.

### Формат входных данных

В первой строке входного файла записано число  $N$  ( $0 \leq N \leq 10^6$ ) — количество прямоугольников гистограммы. Затем следует  $N$  целых чисел  $h_1, \dots, h_n$ , где  $0 \leq h_i \leq 10^9$ . Эти числа обозначают высоты прямоугольников гистограммы слева направо. Ширина каждого прямоугольника равна 1

### Формат выходных данных

Выведите площадь самого большого прямоугольника в гистограмме. Помните, что этот прямоугольник должен быть на общей базовой линии.

### Примеры

ВВОД	ВЫВОД
7 2 1 4 5 1 3 3	8

## Задача В. Битва титанов

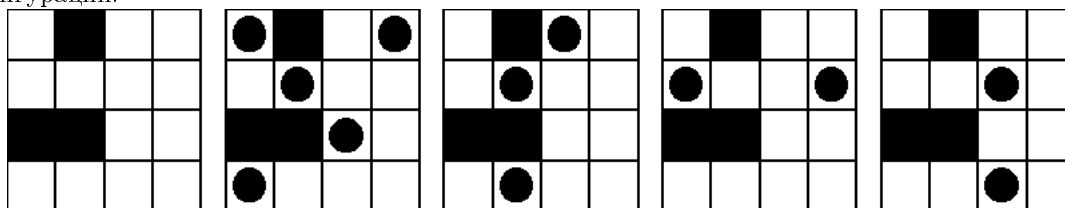
Ограничение по времени: 2 секунды

Ограничение по памяти: 64 Мб

Саша увлекается программированием компьютерных игр. Вот уже три дня он пишет новую игру для сотового телефона под названием "Битва титанов". Героями игрушки являются оловянные солдатики. В качестве прототипа для описания действий оловянного солдата Саша взял шахматную ладью.

Шахматная ладья — это фигура, которая может перемещаться на любое количество клеток по вертикали или горизонтали. Ладьи не могут перемещаться за препятствия. Задача - вычислить максимальное количество ладей, которые можно поставить на доске так, чтобы никакие две не били друг друга. Это означает, что конфигурация правильна при условии, что никакие две ладьи не находятся на одной горизонтали или вертикали в пределах видимости друг друга.

Следующий пример показывает пять изображений. Первое изображение является пустым, второе и третье изображения показывают правильные конфигурации, а четвертый и пятый рисунок — примеры неправильных конфигураций.



Помогите Саше поскорее закончить программу и вычислите максимальное количество ладей на заданной конфигурации доски.

### Формат входных данных

Во входном файле в первой строке содержится натуральное число  $N$  — размер доски, не превышающий 4. Следующие  $N$  строк содержат по  $N$  символов — описание шахматной доски, причем символ '.' указывает пустую клетку, а символ верхнего регистра 'X' указывает препятствие. Во входном файле нет пробелов.

### Формат выходных данных

Выведите максимальное количество ладей на правильной конфигурации доски.

### Примеры

ВВОД	ВЫВОД
4 .X.. .... XX.. ....	5

## Задача С. Обходчик лабиринтов

Ограничение по времени: 2 секунды  
Ограничение по памяти: 64 Мб

В скором времени на телеэкраны одной страны выйдет новое шоу «Двое в лабиринте». Его сюжет будет состоять в том, что два участника будут помещены в лабиринт. Их целью является найти из него выход. Первый, кто найдет выход, получит крупный денежный приз.

Однако, прежде чем шоу выйдет на экраны, лабиринты должны быть сертифицированы Государственным Бюро по Сертификации Лабиринтов. В своей работе бюро использует специальные машины, называемые *обходчиками лабиринтов*.

Поскольку в силу специфики работы этих машин для каждого лабиринта приходится строить нового обходчика, Вам поручено провести компьютерное моделирование обхода лабиринта обходчиком.

Лабиринт состоит из  $n$  комнат, соединенных  $m$  коридорами. На концах коридора имеются две двери, одна из которых открывается только из коридора, в вторая — только из комнаты, из которой коридор выходит, — таким образом, движение по коридору разрешено только в одну сторону. Кроме этого, каждый из коридоров покрашен в один из  $k$  цветов (это сделано для того, чтобы немного облегчить участникам нахождение выхода из лабиринта). Цвет коридора указан на соответствующей ему двери в комнате, из которой он выходит. При этом из комнаты могут выходить несколько коридоров одного цвета.

Обходчик лабиринтов работает по программе, которая состоит из  $l$  инструкций. Каждая инструкция — это номер цвета (число от 1 до  $k$ ). Обход лабиринта начинается в комнате номер  $s$  и совершается следующим образом: обходчик поочередно считывает инструкции и на каждом шаге выбирает один из коридоров, покрашенных в цвет, указанный в этой инструкции. Если такого коридора не находится, то обходчик «зависает».

Так как на каждом шаге у обходчика может быть не один вариант выбора коридора, то комната, в которой он окажется после выполнения программы может определяться неоднозначно.

Ваша задача состоит в том, чтобы по описанию лабиринта и программе для обходчика определить, в каких комнатах обходчик может оказаться после выполнения соответствующей программы.

### Формат входных данных

Первая строка входного файла содержит три целых числа:  $n, m, k$  ( $1 \leq n, k \leq 1000, 0 \leq m \leq 10000$ ).

Далее идут  $m$  строк, описывающих коридоры. Описание каждого коридора состоит из трех целых чисел:  $u, v, c$  ( $1 \leq u, v \leq n, 1 \leq c \leq k$ ). Их значения таковы:  $u$  — номер комнаты, из которой выходит коридор,  $v$  — номер комнаты, в которую ведет коридор,  $c$  — цвет этого коридора. Коридор может вести из комнаты в саму себя, между двумя комнатами может существовать несколько коридоров (более того, несколько коридоров одного цвета).

$(m + 2)$ -ая строка входного файла содержит целое число  $l$  ( $1 \leq l \leq 1000$ ).  $(m + 3)$ -ая строка содержит  $l$  целых чисел от 1 до  $k$  — программы для обходчика лабиринта.

Последняя строка входного файла содержит целое число  $s$  ( $1 \leq s \leq n$ ).

### Формат выходных данных

В случае, если обходчик «зависает» независимо от того, какие коридоры он выбирает в случае существования нескольких коридоров одного цвета, выведите в выходной файл слово `Hangs`.

Иначе, выведите на первой строке выходного файла слово `OK`, во второй — количество  $r$  комнат, в которых обходчик может оказаться после выполнения программы. Во третьей строке выходного файла в этом случае выведите номера этих комнат в возрастающем порядке.

## Примеры

ввод	вывод
4 6 2 1 2 1 1 2 2 1 3 1 1 3 2 3 4 2 3 3 2 2 1 2 1	OK 2 3 4
4 6 2 1 2 1 1 2 2 1 3 1 1 3 2 3 4 1 3 3 1 2 1 2 1	Hangs

В первом примере обходчик после выполнения первой инструкции может оказаться в комнатах 2 и 3. Из комнаты 2 не выходят коридоры, имеющие цвет 2, поэтому в ней обходчик «зависает». Из комнаты 3 выходят два коридора, имеющие цвет 2, — они ведут в комнаты 3 и 4. Поэтому после выполнения всей программы обходчик может оказаться в комнатах 3 и 4.

Во втором примере обходчик после выполнения первой инструкции может оказаться в комнатах 2 и 3. Однако, все коридоры, выходящие из этих комнат, имеют цвет 1. Поэтому обходчик «зависает» в любой из комнат.

## Задача D. Игра в 24

Ограничение по времени: 2 секунды  
Ограничение по памяти: 64 Мб

«Игра в 24» — это математическая игра, в которой используются специальные карточки. На каждой карточке записаны четыре числа. Задача игроков состоит в том, чтобы получить число 24, используя эти числа и арифметические операции (сложение, вычитание, умножение и деление, скобки при этом можно расставить произвольным образом). «Игра в 24» используется в некоторых школах при изучении математики в начальных классах.

В этой задаче рассматривается упрощенный вариант этой игры, в котором в число разрешенных операций входят только сложение, вычитание и умножение.

Назовем карточку для упрощенной «Игры в 24» *правильной*, если из указанных на ней чисел с помощью сложения, вычитания, умножения и расстановки скобок произвольным образом можно получить число 24.

Фирма *American Card Manufacturer (ACM)* занимается выпуском наборов карточек для этой игры. Однако, выпуск таких карточек сопряжен с некоторыми трудностями. Одна из них состоит в том, что не любой набор из четырех чисел задает *правильную* карточку.

По этой причине задача проверки «правильности» данной карточки является весьма актуальной. Ваша задача состоит в написании программы, которая будет осуществлять указанную проверку.

### Формат входных данных

Входной файл содержит четыре натуральных числа, не превосходящих 30, которые написаны на исследуемой карточке.

### Формат выходных данных

В выходной файл выведите слово YES, если карточка является правильной, и слово NO — иначе.

### Примеры

ВВОД	ВЫВОД
1 2 3 4	YES
1 1 1 1	NO