

Problem A: Тролль и пароль

Prob Id: 1

Full score: 100

Score bonus:

Input file name: `input.txt` or standard input

Output file name: `output.txt` or standard output

Time limit: 1 s

Real time limit: 2 s

Memory limit: 64M

Тролль и пароль

Тролль Зумбагеш хочет придумать надежный пароль для своего аккаунта в сети "Троллеграм". Правила таковы, что для этого он должен иметь карточки с теми символами, которые входят в пароль. У Зумбагеша есть **a** карточек с большой буквой 'A', **b** карточек с большой буквой 'B' и **c** карточек с большой буквой 'C'. Из них он и может составить свой пароль.

С одной стороны он хочет, чтобы пароль был надежным, поэтому в его пароле не должны рядом встречаться два одинаковых символа. Например пароли "AAA", "CBVACA", "ABABCCAB" он считает простыми, так как в них присутствуют два рядом стоящих одинаковых символа.

С другой стороны, Зумбагеш боится забыть свой пароль, поэтому хочет составить пароль так, чтобы среди любых трёх подряд идущих символов пароля все три не были разными. Например, он забудет пароль "ABACBC", так как в нём есть три подряд идущих символа "ACB", каждый своего вида.

По заданным числам **a**, **b**, **c** определить длину самого большого пароля, который сможет при таких условиях придумать Зумбагеш.

Input format

На вход подаются три целых числа **a**, **b** и **c** через пробел. $0 \leq a, b, c \leq 10^{18}$. $a + b + c > 0$.

Output format

Вывести одно число --- длину самого большого пароля, который сможет придумать Зумбагеш при таких условиях.

Examples

Input

2 1 0

Output

3

Notes

В примере из условия, у тролля есть две буквы 'A', одна буква 'B' и ни одной буквы 'C'. Из них он может составить пароль "ABA", удовлетворяющий всем его условиям. И это будет самый длинный пароль, который он сможет составить из этих букв.

Подзадачи

Подзадача	Номера тестов	Баллы	Ограничения	Необходимые подзадачи
0 (тесты из условия)	1	0		
1	2-26	50	$0 \leq a, b, c \leq 10^9$	0
2	27-51	50	без дополнительных ограничений	0, 1

Проверка этой задачи проводится по группам тестов. Это значит, что решение должно выдать правильный ответ для каждого теста группы — только в этом случае все баллы за эту подзадачу будут засчитаны проверяемому решению. Если хотя бы на один тест из группы будет дан неверный ответ, баллы, начисляемые проверяемой программе за эту подзадачу, равны 0. Помимо этого, баллы за подзадачу начисляются только в том случае, когда пройдены тесты всех необходимых для нее подзадач.

Problem B: Поиск перестановки

Prob Id: 2

Full score: 100

Score bonus:

Input file name: `input.txt` or standard input

Output file name: `output.txt` or standard output

Time limit: 1 s

Real time limit: 1 s

Memory limit: 64M

Поиск перестановки

Две сестры Аня и Яна играют в следующую математическую игру. Аня загадывает последовательность **T** из **n** натуральных чисел **a**. После этого она пишет эту последовательность на доске, а Яна находит в ней **самую большую** перестановку **P**, которая является непрерывным отрезком предложенной последовательности.

Последовательность **P** являются отрезком последовательности **T**, если можно у последовательности **T** удалить несколько подряд идущих (возможно ни одного) элементов из начала и несколько подряд идущих (возможно ни одного) элементов из конца и получить после этого последовательность **P**.

Последовательность **P** является перестановкой длины **k**, если она содержит все числа от **1** до **k** по одному разу в произвольном порядке.

Требуется проверить результат, полученный Яной, то есть по исходной последовательности **T** найти длину самой большой перестановки, являющейся её непрерывным отрезком.

Input format

В первой строке записано число **n** --- количество чисел в исходной последовательности, $1 \leq n \leq 1000$. В следующей строке находятся **n** чисел **a**, через пробел --- элементы последовательности **T**, загаданной Аней. Все они в пределах от **1** до **n**.

Output format

Вывести одно число --- ответ на задачу. Если ни одной перестановки нет, вывести 0.

Examples

Input

```
15
3 4 2 5 1 3 6 7 2 4 3 1 8 2 1
```

Output

```
7
```

Notes

В предложенной последовательности встречаются перестановки 2 1, и 2 4 3 1, 4 2 5 1 3, 3 4 2 5 1, но самой большой является перестановка 5 1 3 6 7 2 4 длины 7.

Эта задача оценивается на 33 тестах. За каждый пройденный тест ваше решение получит 3 балла. Тест из условия не входит в набор оцениваемых тестов для этой задачи, но проверяется первым. В целях уменьшения нагрузки на тестирующую систему, для проведения полной проверки на тестах выше теста №17, требуется успешное прохождение этого теста. За прохождение теста №17 начисляется 4 балла.

Problem C: Ботаоси-2

Prob Id: 3

Full score: 100

Score bonus:

Input file name: input.txt or standard input

Output file name: output.txt or standard output

Time limit: 1 s

Real time limit: 1 s

Memory limit: 64M

Ботаоси-2

Продолжаем увлекательную историю школьного чемпионата по ботаоси. Напомним, что в этой игре есть два столба и, соответственно, две команды, которые защищают свой столб и пытаются организовать атаку на столб противника. Физрук Борис Васильевич уже нашел на школьном дворе два подходящих столба для игры. Расстояние d между этими столбами не обязательно целое, но оно таково, что **квадрат** этого расстояния равен целому неотрицательному числу k , то есть верно, что $d * d = k$.

Для начала двоечниками Петровым и Васечкиным была очищена от снега часть площадки между столбами. Так как Борис Васильевич велел очистить снег так, столбы были на границе или внутри очищенной площадки, но забыл сказать про её форму, Петров и Васечкин просто очистили круг такой, что столбы оказались ровно на концах его диаметра и сразу после этого убежали.

Теперь Борис Васильевич хочет разметить границы площадки для игры, которая по его мнению должна иметь форму четырехугольника, вписанного в уже очищенный круг. Два **противоположных** угла четырехугольника уже есть --- это исходные два столба, осталось выбрать еще два угла.

Борис Васильевич хочет, чтобы стороны этого четырехугольника были целыми числами. Так как для игры очень важным является правильный подбор площади игрового поля, тренер хочет рассмотреть все варианты площадей, которые могут иметь получающиеся при этих условиях четырехугольники и выбрать самый оптимальный на его взгляд. Вам нужно помочь ему, составив список чисел, которые

могут быть площадью вписанного в круг четырехугольника с двумя **противоположными** вершинами в концах его диаметра и целыми сторонами.

Input format

На вход подаётся одно целое неотрицательное число k --- **квадрат** расстояния между столбами. $1 \leq k \leq 10^{12}$.

Output format

В первой строке вывести число n --- количество различных вариантов ответа, найденных вашим решением. В следующих n строках по одному вывести различные числа S_i , такие, что S_i является площадью некоторого вписанного в круг четырехугольника с целыми сторонами и двумя противоположными углами в концах диаметра круга. Числа S_i нужно выводить в порядке возрастания. Можно показать, что все они должны быть целыми.

Examples

Input

2

Output

1
1

Input

50

Output

3
7
16
25

Input

325

Output

6
18
60
84
102
126
150

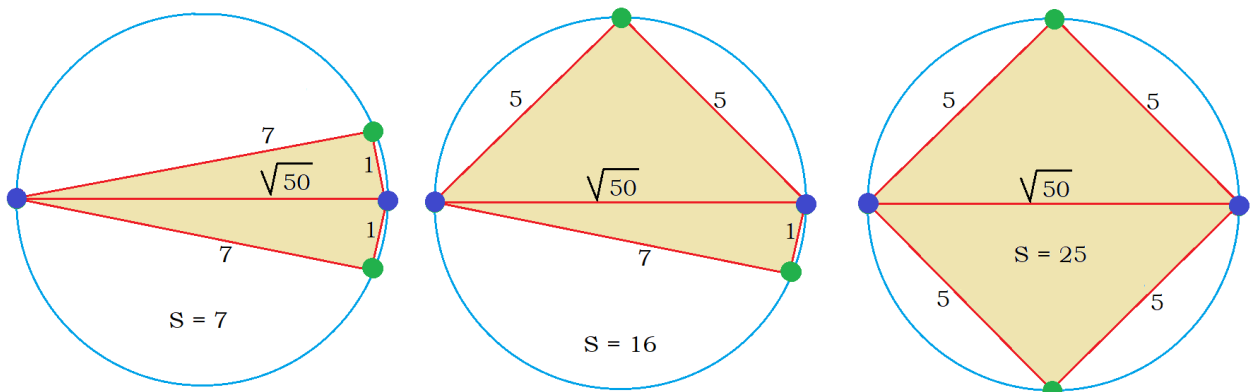
Input

1105

Output

10
132
210
252
288
330
342
372
420
462
552

Notes



На рисунке можно видеть иллюстрацию ко второму примеру из условия. Синими кружками обозначены исходные два столба на концах диаметра. Зелёными кружками обозначены две добавленные вершины. Видно что есть три разновидности четырехугольников, вписанных в круг с диаметром $\sqrt{50}$. Первая разновидность таких четырехугольников имеет площадь 7 (их может быть несколько различных с одной и той же площадью), вторая имеет площадь 16 и третья имеет площадь 25. Нас интересуют именно различные по площади варианты четырехугольников, все четырехугольники, имеющие одинаковую площадь, считаются неразличимыми.

Подзадачи

Подзадача	Номера тестов	Баллы	Ограничения	Необходимые подзадачи
0 (тесты из условия)	1-4	0		
1	5-25	33	$n \leq 10^6$	0
2	26-60	67	без дополнительных ограничений	0, 1

Проверка этой задачи проводится по группам тестов. Это значит, что решение должно выдать правильный ответ для каждого теста группы — только в этом случае все баллы за эту подзадачу будут засчитаны проверяемому решению. Если хотя бы на один тест из группы будет дан неверный ответ, баллы, начисляемые проверяемой программе за эту подзадачу, равны 0. Помимо этого, баллы за подзадачу начисляются только в том случае, когда пройдены тесты всех необходимых для нее подзадач.

Problem D: Коллективные поздравления

Prob Id: 4
Full score: 100
Score bonus:
Input file name: input.txt or standard input
Output file name: output.txt or standard output
Time limit: 1 s
Real time limit: 2 s
Memory limit: 64M

Коллективные поздравления

В одной организации работают n сотрудников. Само собой, у каждого из них есть свой день рождения. Профсоюз этой организации регулярно проводит коллективные дни рождения, которые организуются следующим образом: выбираются несколько работников, у которых **дни рождения идут в календаре подряд**, объявляется день, когда все остальные их будут поздравлять, дарить подарки и самое главное, для именинников проводится концерт по их заявкам.

Профсоюз может организовать выступление на коллективных днях рождения нескольких известных в городе исполнителей, на данный момент в этом списке есть 26 певцов и ансамблей, которых для краткости будем обозначать строчными латинскими буквами от 'a' до 'z'. Ограничений на число выступлений у исполнителей в течение года нет, но выступление каждого, само собой, требует оплату. Для упрощения, будем считать, что оплата у всех исполнителей одинаковая.

Каждый сотрудник в организации в самом начале текущего года заполнил анкету, в которой указал имена исполнителей из предложенных на выбор 26 певцов и ансамблей, которых бы он хотел бы услышать на своем дне рождения. Будем считать, что те певцы, которых он не выбрал, ему не нравятся и он не хочет их слышать.

Председатель профсоюза, Петр Петрович может сам выбрать на какие группы делить именинников. Само собой, внутри одной группы именинники должны иметь подряд идущие дни рождения. После того, как он их разобьет, можно приступать к организации поздравительных концертов. Внутри каждой группы выбираются и приглашаются для её поздравления все исполнители, которых хотят видеть одновременно все члены этой группы. Само, собой, разбиение должно быть таким, чтобы концерт состоялся, то есть пересечение списков именинников внутри группы, не должно быть пустым. С другой стороны, если все именинники в группе хотят услышать некоторого исполнителя, то Петр Петрович просто обязан его пригласить.

Следует заметить, что количество таких коллективных дней рождения Петр Петрович может выбирать сам. Например, он может всех сотрудников поздравить за один раз, но при условии, что концерт состоится, то есть все сотрудники хотят услышать как минимум одного и того же певца. А может, наоборот, каждого сотрудника поздравлять своим концертом, но тогда ему придется удовлетворять абсолютно все заявки.

Выступление звезд эстрады стоит очень дорого, поэтому Петр Петрович хочет минимизировать затраты на проведение всех дней рождения в году. Он хочет разбить список именинников так, чтобы достичь минимального суммарного количества выступлений. Требуется найти это минимальное количество.

Input format

В первой строке находится число n --- количество сотрудников в организации. $1 \leq n \leq 1000$ Далее в следующих n строках содержатся заявки каждого из этих сотрудников. Заявки упорядочены по дню рождения сотрудников. Каждая заявка представляет собой непустую строку из различных строчных букв латинского алфавита, перечисленных в алфавитном порядке. Длина каждой строки не превышает 26.

Output format

Вывести минимальное возможное суммарное число выступлений, которые профсоюз должен организовать для поздравления всех сотрудников организации.

Examples

Input

```
8
cdefhz
abcdf
cdfgkh
abegh
aceg
cefg
bcfh
bf
```

Output

5

Input

4
ab
cd
ab
cd

Output

8

Input

2
abc
abc

Output

3

Input

7
bc
ab
a
a
a
ab
bc

Output

3

Notes

Рассмотрим первый пример из условия. Петр Петрович не может поздравить всех сотрудников за один раз, так как нет ни одного исполнителя, любимого всеми сотрудниками. Но и поздравлять отдельно каждого не тоже не стоит.

Можно попробовать разбить именинников на группы так: сначала организуем именины первых трех (из их заявок **cdefhz**, **abcdf** и **cdfgkh** придется пригласить трёх исполнителей **cdf**), потом объединим еще троих с заявками **abegh**, **aceg** и **cefg**, для них придется организовать выступление двух певцов **eg** и, наконец объединим последних двух с заявками **bcfh** и **bf**, у которых будут выступать двое исполнителей **bf**. В итоге такого разбиения придется оплатить семь номеров.

Есть более хорошее решение: сначала поздравим первых двух **cdefhz**, **abcdf** с теми же тремя исполнителями **cdf**, потом поздравим сразу четырех подряд **cdfgkh**, **abegh**, **aceg** и **cefg**, их придет поздравлять один очень популярный рок-певец **g**, и наконец двое оставшихся **bcfh** и **bf** снова будут слушать двоих певцов **bf**. При таком разбиении придется пригласить уже шесть исполнителей.

Но самым лучшим будет следующее: поздравим первых двух **cdefhz**, **abcdf** с теми же тремя исполнителями **cdf**, потом поздравим только трёх подряд **cdfgkh**, **abegh**, **aceg** и снова для них выступит только **g**, и наконец трое оставшихся **cefg**, **bcfh** и **bf** будут слушать одного певца **f**. В итоге профсоюз сможет ограничиться пятью номерами. Меньшим количеством при этих заявках не обойтись.

Во втором примере никаких двух подряд идущих именинников нельзя объединить в одном поздравлении. Придется поздравлять их каждого по-отдельности и организовать восемь музыкальных номеров.

Подзадачи

Подзадача	Номера тестов	Баллы	Ограничения	Необходимые подзадачи
0 (тесты из условия)	1-4	0		
1	5-15	25	$n \leq 50$	0
2	16-21	25	$n \leq 300$	0, 1
3	22-34	50	без дополнительных ограничений	0, 1, 2

Проверка этой задачи проводится по группам тестов. Это значит, что решение должно выдать правильный ответ для каждого теста группы — только в этом случае все баллы за эту подзадачу будут засчитаны проверяемому решению. Если хотя бы на один тест из группы будет дан неверный ответ, баллы, начисляемые проверяемой программе за эту подзадачу, равны 0. Помимо этого, баллы за подзадачу начисляются только в том случае, когда пройдены тесты всех необходимых для нее подзадач.

Problem E: Орбитальная станция

Prob Id: 5

Full score: 100

Score bonus:

Input file name: `input.txt` or standard input

Output file name: `output.txt` or standard output

Time limit: 2 s

Real time limit: 2 s

Memory limit: 64M

Орбитальная станция

Орбитальная станция состоит из n модулей, пристыкованных друг к другу в некотором порядке. Любые два соседних её модуля соединяются при помощи переходного шлюза. Так как при её построении использовался принцип минимизации шлюзов, то очередной пристыковываемый модуль соединялся шлюзом ровно с одним из тех, что уже находились в составе станции. Первый модуль, само собой, ни к чему не пристыковывался. Таким образом, структура орбитальной станции такова, что между любыми двумя модулями имеется ровно один путь.

Шло время, и наконец настала пора диагностики и ремонта всей станции. Для этого к ней был отправлен большой грузовой челнок, на котором находится ровно n комплектов для диагностики и ремонта --- по одному для каждого модуля. Челнок можно пристыковать к любому модулю. После этого космонавты сначала выгрузят все комплекты в стартовый модуль, к которому был пристыкован челнок. Далее они должны будут транспортировать из этого модуля ремонтные комплекты в каждый из $n - 1$ других модулей. Так как через любой шлюз за один раз можно переместить ровно один комплект, придется каждый раз транспортировать ровно один комплект в соответствующий модуль через все шлюзы на пути до нужного модуля, а после этого возвращаться обратно за следующим комплектом. При этом наибольшую сложность при транспортировке комплекта как раз представляет прохождение любого шлюза. Таким образом, требуется найти такой модуль, к которому нужно пристыковать грузовой челнок так, чтобы суммарное количество всех проходимых с грузом при транспортировке шлюзов было минимально возможным.

Input format

В первой строке находится число модулей n . $1 \leq n \leq 10^5$.

В следующих $n - 1$ строках содержится информация о шлюзах, соединяющих модули станции. В каждой строке содержится через пробел два числа --- номера двух различных модулей, соединённых шлюзом. Эти номера находятся в пределах от 1 до n .

Гарантируется, что между любыми двумя модулями заданной в тестах орбитальной станции существует ровно один путь.

Output format

Вывести два числа через пробел: номер модуля, суммарное количество проходимых с грузом шлюзов из которого во все остальные модули будет минимальным и само это суммарное число проходимых шлюзов. Если таких модулей несколько, нужно вывести тот, у которого минимальный номер.

Examples

Input

```
12
3 1
1 9
9 7
8 5
7 8
10 3
11 2
2 4
2 10
6 10
12 6
```

Output

```
3 29
```

Notes

В тесте из примера имеются два модуля с минимальным суммарным числом проходимых из них шлюзов, равным 29. Это модули номер 3 и номер 10. Так как минимальный среди них имеет номер 3, он и будет ответом.

Подзадачи

Подзадача	Номера тестов	Баллы	Ограничения	Необходимые подзадачи
0 (тесты из условия)	1-1	0		
1	2-17	15	$n \leq 100$	0
2	18-24	20	$n \leq 1000$	0, 1
3	25-47	65	без дополнительных ограничений	0, 1, 2

Проверка этой задачи проводится по группам тестов. Это значит, что решение должно выдать правильный ответ для каждого теста группы — только в этом случае все баллы за эту подзадачу будут засчитаны проверяемому решению. Если хотя бы на один тест из группы будет дан неверный ответ, баллы, начисляемые проверяемой программе за эту подзадачу, равны 0. Помимо этого, баллы за подзадачу начисляются только в том случае, когда пройдены тесты всех необходимых для нее подзадач.

Problem F: Куличики-2

Prob Id: 6
Full score: 100
Score bonus:
Input file name: input.txt or standard input
Output file name: output.txt or standard output
Time limit: 1 s
Real time limit: 1 s
Memory limit: 64M

Куличики-2

Двое друзей Петя и Вася снова играют в песочнице. Петя расчистил n мест, расположенных в один ряд, на которых собирается строить куличики. Время от времени он выбирает некоторую позицию номер i из этого ряда и стрит на ней новый куличик некоторой высоты $newh$. Если на этой позиции до этого был куличик, то Петя сначала его ломает, а потом строит новый.

Иногда в процесс куличикостроения вмешивается и Вася. Он делает горизонтальный взмах лопаткой для песка и некоторые куличики на некотором непрерывном отрезке уменьшают свою высоту. Более точно, если он производит взмах от куличика номер l до куличика номер r на высоте h , то все куличики на отрезке от l до r , у которых высота была больше h , принимают высоту равную h .

Таким образом, процесс их игры можно описать несколькими операциями двух видов:

Операция вида **1 i newh** означает, что Петя строит на позиции i новый куличик с высотой $newh$. Старый куличик при этом уничтожается. Можно считать, что изначально на всех позициях были куличики высотой 0.

Операция вида **2 l r h** означает, что Вася производит взмах лопаткой от куличика номер l до куличика номер r на высоте h . Все куличики на этом отрезке, имеющие высоту большую чем h , уменьшаются до этой высоты.

После долгой игры приятели пошли домой, а куличики остались в песочнице. Требуется для каждой позиции определить куличик какой высоты оказался на ней.

Input format

В первой строке содержатся через пробел два целых числа n и q --- число позиций, на которых строились куличики и число операций, которые друзья сделали при этом. $1 \leq n, q \leq 3 \cdot 10^5$.

В следующих q строках описаны операции над куличиками, сделанные ребятами в том порядке, в котором они производились. Эти операции имеют вид либо

1 i newh, где $1 \leq i \leq n, 0 \leq newh \leq 10^9$,

либо

2 l r h, где $1 \leq l \leq r \leq n, 0 \leq h \leq 10^9$.

Output format

Вывести n чисел через пробел. Число на позиции i должно быть равно высоте куличика, который оказался на этой позиции по окончании игры Пети и Васи. Если в процессе игры на некоторой позиции куличики не строились, нужно для неё вывести 0.

Examples

Input

```
5 14
1 3 4
1 2 7
2 2 5 6
1 5 2
1 4 8
1 1 7
2 1 4 5
1 5 12
2 3 5 10
1 2 0
2 1 3 6
2 4 4 0
1 2 9
2 1 5 8
```

Output

```
5 8 4 0 8
```

Notes

Подзадачи

Подзадача	Номера тестов	Баллы	Ограничения	Необходимые подзадачи
0 (тесты из условия)	1-1	0		
1	2-15	15	$n, q \leq 3000$	0
2	16-27	85	без дополнительных ограничений	0, 1

Проверка этой задачи проводится по группам тестов. Это значит, что решение должно выдать правильный ответ для каждого теста группы — только в этом случае все баллы за эту подзадачу будут засчитаны проверяемому решению. Если хотя бы на один тест из группы будет дан неверный ответ, баллы, начисляемые проверяемой программе за эту подзадачу, равны 0. Помимо этого, баллы за подзадачу начисляются только в том случае, когда пройдены тесты всех необходимых для нее подзадач.
