

2020 autumn training episode 34, round F10

Statement is not available on
English language

A. Sáp nhập địa phương

time limit per test: 0.25 seconds

memory limit per test: 16 megabytes

input: merge.inp

output: merge.out

Thực hiện chủ trương "tinh giản bộ máy, cắt giảm biên chế" của chính phủ, thời gian tới nhiều tỉnh thành sẽ sáp nhập với nhau, nhiều cơ quan ban, bộ, ngành cũng được ghép làm một. Điều này làm sẽ làm giảm số lượng ghế chủ tịch, phó chủ tịch, giám đốc, phó giám đốc; qua đó giảm chi phí tiền lương để nuôi bộ máy nhà nước và tiết kiệm ngân sách cho người dân.

Tuy nhiên, đặt tên địa phương, ban bộ thế nào sau khi sáp nhập là cả một vấn đề lớn. Những tên địa danh, tỉnh thành vốn mang tính lịch sử lâu đời, ăn sâu vào tiềm thức của người dân và gắn bó như một phần máu mủ ruột thịt. Bởi thế, sau khi sáp nhập, chính phủ muốn dùng những tiếng có trong tên cũ của các địa phương để ghép lại thành tên mới. Điều này vừa giữ lại phần nào bản sắc đặc trưng của địa phương, vừa thể hiện sự gắn kết bền chặt. Ví dụ, thành phố Hòa Bình và tỉnh Hà Tĩnh sẽ sáp nhập thành tỉnh "Bình Tĩnh", huyện đảo Côn Đảo và huyện Đồ Sơn sáp nhập lại thành huyện "Côn Đồ". Ba tỉnh Đắk Lắk, Kon Tum, Play Ku sẽ sáp nhập thành tỉnh "Lắk Kon Ku". Tương tự, bộ Nội Vụ và bộ Y Tế có thể sẽ sáp nhập thành bộ "Nội Y".

Cho biết tên của n địa danh hoặc cơ quan chuẩn bị được sáp nhập với nhau, và số thứ tự của tiếng được lấy ra từ mỗi tên để tạo thành tên sau khi sáp nhập, hãy cho biết tên sau khi sáp nhập n địa danh hoặc cơ quan này.

Input

- Dòng đầu tiên chứa số nguyên n ($1 \leq n \leq 100$) — số địa phương hoặc cơ quan được tiến hành sáp nhập.
- n dòng tiếp theo, mỗi dòng ghi tên của một địa phương kèm theo một số nguyên dương là chỉ số của tiếng trong tên sẽ được dùng vào tên sáp nhập. Số nguyên dương này được

đảm bảo không lớn hơn số tiếng của tên gọi. Mỗi tên gọi gồm không quá 100 tiếng, mỗi tiếng gồm không quá 7 ký tự in thường.

Output

Gồm một dòng duy nhất chứa tên sau khi sáp nhập n cơ quan, tổ chức hay địa phương nói trên.

Examples

Input

Copy

```
2
noi vu 1
y te 1
```

Output

Copy

```
noi y
```

Input

Copy

```
3
dak lak 2
kon tum 1
play ku 2
```

Output

Copy

```
lak kon ku
```

Statement is not available on
English language

B. Viêm phổi Vũ Hán 5

time limit per test:0.75 seconds

memory limit per test:128 megabytes

input:social-distancing.inp

output:social-distancing.out

Viêm phổi Vũ Hán hay *Cúm Trung Quốc* là bệnh viêm đường hô hấp cấp do chủng mới của Virus Corona (tổ chức y tế thế giới gọi là SARS-CoV-2) gây ra. Bắt nguồn từ chợ hải sản Hoa Nam — một khu chợ chuyên buôn bán động vật hoang dã tại thành phố Vũ Hán, tỉnh Hồ Bắc, Trung Quốc; dịch đã lan ra hơn 200 quốc gia và vùng lãnh thổ trên thế giới. Tính đến ngày 16 / 08 / 2020, đã có hơn 21.8 triệu bệnh nhân nhiễm bệnh và 770 ngàn người tử vong. Hơn 14.5 triệu người đã khỏi bệnh hoặc hồi phục.

Để ngăn chặn COVID-19 lây lan trong cộng đồng, *giãn cách xã hội* là biện pháp then chốt. Tư tưởng là những người dân hạn chế đi lại, cắt giảm sự tiếp xúc với những người xung quanh, do đó cắt đứt đường lây nhiễm của virus. Trước nguy cơ dịch bệnh quay trở lại Hà Nội, chủ tịch NĐC lên phương án quy hoạch lại thành phố nhằm khoanh vùng dập dịch nhanh chóng và hiệu quả.

Hiện nay, thành phố có n công trình mọc lên ở khắp nơi. Nếu mô tả thành phố trên mặt phẳng tọa độ Descartes, n công trình này được đặt tại n điểm có tọa độ lần lượt là $(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)$. Trong phương án tái quy hoạch thành phố, chủ tịch dự định phá một số công trình đã có để làm đường. Đầu tiên chủ tịch chọn ra số nguyên d gọi là kích thước của mỗi khu dân cư. Các con đường sẽ được xây dựng song song với một trong hai trục tọa độ. Sẽ có vô số con đường song song với trục Ox được mở sao cho khoảng cách giữa hai con đường liên tiếp nhau luôn là d . Tương tự, có vô số con đường song song với trục Oy được mở và khoảng cách giữa hai con đường liên tiếp cũng phải là d .

Những con đường được mở sẽ khoanh vùng các khu dân cư thành nhiều khu vực, và khi ghi nhận ca dương tính ở một khu vực nào đó, chỉ cần rà soát và cách ly nơi đó là xong. Kế hoạch có tương lai là vậy, nhưng gặp phải một trở ngại lớn: Tất cả các công trình nằm trên khu vực dự kiến mở đường phải bị phá hủy. Vì vậy, chủ tịch thành phố muốn chọn cách mở đường sao cho số công trình bị phá dỡ là nhỏ nhất.

Input

Dòng đầu tiên chứa hai số nguyên d và n ($1 \leq d \leq 1000, 0 \leq n \leq 10^5$) — khoảng cách giữa hai con đường song song liên tiếp và số công trình đang có trong thành phố.

n dòng tiếp theo, mỗi dòng chứa hai số nguyên x_i và y_i ($-10^9 \leq x_i, y_i \leq 10^9$) thể hiện vị trí của một công trình.

Output

In ra một số nguyên duy nhất là số công trình nhỏ nhất phải bị phá hủy.

Example

Input

Copy

```
3 10
2 4
4 5
5 2
5 4
5 7
6 2
7 2
7 7
8 2
8 4
```

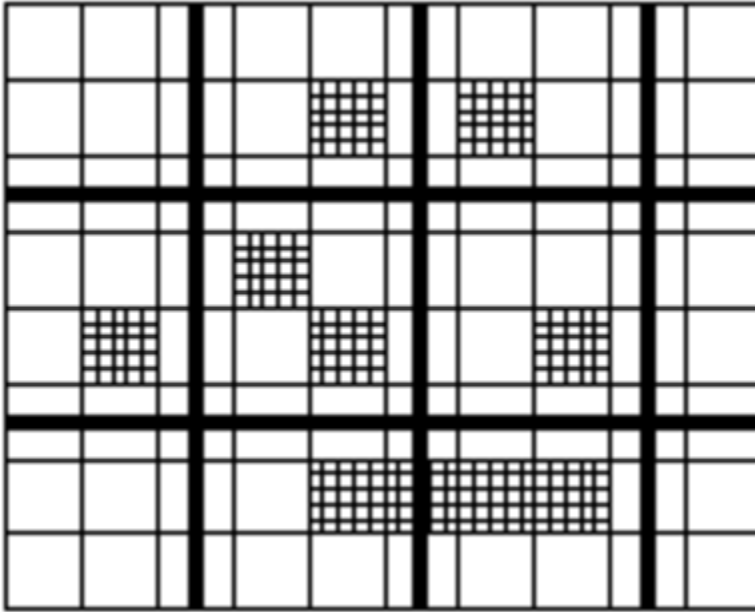
Output

Copy

1

Note

Hình vẽ dưới đây mô tả test ví dụ. Các ô vuông được đánh dấu miêu tả các công trình. Các đường in đậm thể hiện các tuyến đường được mở trong phương án cần phá hủy ít công trình nhất.



Statement is not available on
English language

C. Số cuối cùng

time limit per test: 0.5 seconds
memory limit per test: 128 megabytes

input: digits.inp

output: digits.out

Bạn có ba số nguyên không âm p , δ và σ . Bạn viết các số nguyên lần lượt theo thứ tự tăng dần, bắt đầu từ số p và số sau bằng số trước cộng thêm δ . Bạn dừng lại ngay lập tức sau khi đã viết ra đủ σ chữ số, kể cả là khi bạn đang viết dở một số nào đó. Hãy tìm ra số cuối cùng bạn viết.

Input

Gồm một dòng duy nhất chứa ba số nguyên p , δ và σ ($0 \leq p \leq 10^9$, $1 \leq \delta \leq 10$, $1 \leq \sigma \leq 10^{16}$).

Output

In ra một số duy nhất là số cuối cùng bạn sẽ viết.

Scoring

- Subtask 1 (25.2 điểm): $\sigma \leq 106$.
- Subtask 2 (19.6 điểm): $\delta = 1$.
- Subtask 3 (19.6 điểm): $\delta = 2$.
- Subtask 4 (35.6 điểm): Không có ràng buộc gì thêm.

Examples

Input

Copy

47 10 7

Output

Copy

7

Input

Copy

98 1 10

Output

Copy

101

Note

Trong ví dụ đầu tiên, bạn định viết ra dãy số 47, 57, 67, 77 nhưng số 77 bạn chỉ viết một chữ số 7 đầu tiên do bạn đã viết đủ 7 chữ số. Do đó dãy số bạn viết ra là 47 57 67 7 với số cuối cùng là 7.

Trong ví dụ thứ hai, dãy số bạn viết là 98 99 100 101.

Statement is not available on
English language

D. Vòng trang sức

time limit per test: 0.5 seconds
memory limit per test: 256 megabytes
input: puzzles.inp
output: puzzles.out

Bé Hạnh có n dây chuyền, mỗi dây chuyền có một số hạt trang sức. Có tất cả 26 loại hạt trang sức khác nhau. Bé Hạnh muốn làm một chiếc vòng đeo tay từ số đồ trang sức này.

Để làm chiếc vòng mơ ước, đầu tiên bé Hạnh xếp n dây chuyền đã có ra trên một hàng. Sau đó bé Hạnh chọn ra một số (ít nhất một) dây chuyền để làm chiếc vòng của mình. Theo thứ tự từ trái qua phải, bé Hạnh lần lượt đặt những chiếc dây đã chọn lên một vòng tròn theo chiều kim đồng hồ nhưng không làm thay đổi thứ tự của các hạt trang sức trên mỗi dây. Cuối cùng, bé lấy chỉ buộc nối những cặp hạt trang sức cạnh nhau trên những dây chuyền khác nhau để tạo thành chiếc vòng đeo tay. Bé Hạnh muốn chiếc vòng của mình càng to càng tốt (nghĩa là tổng số hạt trang sức trên các dây được chọn là lớn nhất có thể). Ngoài ra, bé Hạnh muốn rằng hai hạt trang sức được buộc chỉ lại với nhau phải thuộc cùng một loại.

Nói cách khác, ta ký hiệu 26 loại hạt trang sức bằng các chữ cái in thường từ 'a' tới 'z' và biểu diễn mỗi sợi dây chuyền của bé Hạnh bằng một xâu ký tự, mỗi ký tự thể hiện loại của một hạt trang sức. Gọi s_1, s_2, \dots, s_n là n dây chuyền của bé Hạnh được xếp ra theo thứ tự từ trái qua phải và giả sử bé Hạnh chọn ra k dây chuyền thứ i_1, i_2, \dots, i_k ($1 \leq i_1 < i_2 < \dots < i_k \leq n$) để làm chiếc vòng đeo tay. Khi đó:

- Ký tự cuối cùng của xâu s_{i_1} phải trùng với ký tự đầu tiên của xâu s_{i_2} .
- Ký tự cuối cùng của xâu s_{i_2} phải trùng với ký tự đầu tiên của xâu s_{i_3} .
- ...
- Ký tự cuối cùng của xâu $s_{i_{k-1}}$ phải trùng với ký tự đầu tiên của xâu s_{i_k} .
- Ký tự cuối cùng của xâu s_{i_k} phải trùng với ký tự đầu tiên của xâu s_{i_1} .

Chú ý rằng, bé Hạnh không làm thay đổi các hạt trong mỗi sợi dây chuyền cũng như đảo ngược thứ tự của chúng. Bé Hạnh có thể chỉ chọn đúng một sợi dây chuyền, khi đó bé sẽ dùng chỉ buộc hạt đầu tiên và hạt cuối cùng của sợi dây này với nhau, và do đó chúng phải thuộc cùng một loại.

Hãy giúp bé Hạnh tạo được chiếc vòng từ nhiều hạt trang sức nhất có thể nhé.

Input

Dòng đầu tiên chứa số nguyên n ($1 \leq n \leq 700000$) — số sợi dây chuyền bé Hạnh có.

Dòng thứ hai chứa n xâu ký tự mô tả n dây chuyền của bé Hạnh theo thứ tự từ trái qua phải. Mỗi xâu ký tự có từ 1 tới 10 ký tự latin in thường.

Output

Gồm một số nguyên duy nhất là số hạt trang sức tối đa trên chiếc vòng đeo tay của bé Hạnh. Nếu bé không thể làm được một chiếc vòng như ý, in ra 0.

Scoring

- Subtask 1 (25 điểm): $n \leq 20$.
- Subtask 2 (25 điểm): $n \leq 300$.
- Subtask 3 (25 điểm): $n \leq 50000$.

- Subtask 4 (25 điểm): $n \leq 700000$.

Examples

Input

Copy

3
abc ca cba

Output

Copy

6

Input

Copy

4
chuyen khtn dhkhtn dhqghn

Output

Copy

0

Input

Copy

3
gs p vh

Output

Copy

1

Note

Trong ví dụ đầu tiên, bé Hạnh chọn hai sợi dây chuyền thứ nhất ('abc') và sợi dây chuyền thứ hai ('cba') với tổng cộng 6 hạt trang sức.

Trong ví dụ thứ hai, bé Hạnh không thể tạo được chiếc vòng đeo tay như ý.

Trong ví dụ thứ ba, bé Hạnh chỉ có thể chọn sợi dây chuyền thứ hai ('p') với chỉ 1 hạt trang sức.

Statement is not available on
English language

E. Kẹo

time limit per test: 0.75 seconds

memory limit per test:256 megabytes

input:candy.inp

output:candy.out

Tiểu Ly là cô bé dễ thương, tinh nghịch và rất thích ăn kẹo. Từ ngày cậu bạn Thư rời đi, Tiểu Ly không còn được ai cho kẹo nữa. Sau khi làm quen với GS.PVH, Tiểu Ly biết được ở một vùng đất xa xôi, có một chiếc xe khổng lồ với vô số loại kẹo hấp dẫn (như ảnh bên dưới). Không để hàm răng sún của mình thất nghiệp, Tiểu Ly quyết lên đường tìm kiếm chiếc xe. Để làm được điều đó, Tiểu Ly cần vượt qua mạng lưới giao thông nơi đây.



Mạng lưới giao thông của vùng đất gồm N giao lộ (đánh số từ 1 tới N) và M con đường hai chiều nối giữa chúng, với những độ dài bất kỳ. Mạng lưới này có tính thông suốt, nghĩa là việc di chuyển từ một giao lộ tới mọi giao lộ khác là luôn khả thi. Với mỗi cặp giao lộ (khác nhau), có một tuyến xe bus 2 tầng phục vụ việc di chuyển giữa hai giao lộ này. Các xe bus luôn chọn lộ trình sao cho thời gian di chuyển giữa hai điểm đầu và cuối là nhỏ nhất. Tiểu Ly sẽ đi bằng xe bus để tới được nơi chứa kẹo. Nhưng thật không may, hôm đó dự kiến sẽ có cuộc diễu hành lớn, và có thể khiến cho xe bus không thể đi qua một số con đường. Với tính tò mò, Tiểu Ly tự hỏi: Nếu một con đường nào đó ngăn cản xe bus đi qua, sẽ có bao nhiêu tuyến bị ảnh hưởng. Hay nói cách khác, nếu một con đường bị phá hủy, sẽ có bao nhiêu cặp giao lộ mà khoảng cách nhỏ nhất giữa chúng tăng lên.

Lưu ý: Nếu một con đường khi bị phá đi làm việc di chuyển giữa hai giao lộ trở nên bất khả thi, con đường này cũng được coi là gây ảnh hưởng đến cặp giao lộ đó.

Input

- Dòng đầu tiên chứa hai số nguyên N và M ($1 \leq N \leq 500$, $1 \leq M \leq 10000$) — số giao lộ và số con đường.
- M dòng tiếp theo, mỗi dòng chứa ba số nguyên u , v , c ($1 \leq u, v \leq N$, $1 \leq c \leq 220797$) thể hiện một con đường hai chiều độ dài c giữa hai giao lộ u và v .

Chú ý:

- Trong 20% số test, $N \leq 100$, $M \leq 300$.
- Trong 40% số test, $N \leq 100$, $M \leq 2000$.

Output

Ghi ra M số nguyên (có giá trị từ 0 tới $\frac{N(N-1)}{2}$), số nguyên thứ i thể hiện số cặp giao lộ bị ảnh hưởng bởi con đường thứ i . Các con đường được đánh số từ 1 tới M theo thứ tự xuất hiện trong input.

Examples

Input

Copy

```
4 4
1 2 1
2 4 10
1 3 5
3 4 7
```

Output

Copy

```
3
2
2
1
```

Input

Copy

```
4 3
1 2 22
1 3 7
1 4 97
```

Output

Copy

```
3
3
3
```

Note

- Trong ví dụ thứ nhất:
 - Cạnh (1, 2) bị xoá sẽ làm độ dài đường đi ngắn nhất giữa 1 và 2 tăng từ 1 lên 22; đường đi ngắn nhất giữa 1 và 4 tăng từ 11 lên 12; đường đi ngắn nhất giữa 2 và 3 tăng từ 6 lên 17.
 - Cạnh (2, 4) bị xoá sẽ làm tăng độ dài đường đi ngắn nhất giữa 1 và 4; giữa 2 và 4.
 - Cạnh (1, 3) bị xoá sẽ làm tăng độ dài đường đi ngắn nhất giữa 1 và 3; giữa 2 và 3.
 - Cạnh (3, 4) bị xoá sẽ làm tăng độ dài đường đi ngắn nhất giữa 3 và 4.
- Trong ví dụ thứ hai: Cạnh (1, 2) bị xoá khiến cho từ 2 không thể di chuyển tới 1, 3 và 4. Tương tự với hai cạnh còn lại.

Statement is not available on
English language

F. Đường đi ngắn nhất

time limit per test: 0.5 seconds
memory limit per test: 128 megabytes

input: shortest.inp

output: shortest.out

Để chuẩn bị cho buổi dạy sắp tới về chủ đề *đường đi ngắn nhất trên đồ thị*, GSPVH đã tạo ra một số bộ dữ liệu là các đồ thị vô hướng liên thông có trọng số. Sau đó, GS dùng thuật toán backtrack tính ra toàn bộ đường đi ngắn nhất giữa hai đỉnh bất kỳ trên các đồ thị này.

Tuy nhiên, do sơ suất, GS đã làm mất các đồ thị gốc ban đầu và chỉ còn lại kết quả về độ dài đường đi ngắn nhất. Dựa vào file dữ liệu về đường đi ngắn nhất, GS có thể suy luận ra số đỉnh của đồ thị, nhưng không có chút thông tin gì về các cạnh của đồ thị. Dựa vào thói quen của mình, GS chắc chắn rằng:

- Các đỉnh của đồ thị được đánh số từ 1 tới n .
- Đồ thị có thể là đa đồ thị (có khuyên và nhiều cạnh nối cùng một cặp đỉnh), nhưng không thể có quá 105 cạnh.
- Trọng số của mỗi cạnh là một số nguyên từ 1 tới 106.
- Đồ thị chắc chắn liên thông

GS nhờ bạn dựng lại đồ thị ban đầu dựa trên thông tin về độ dài đường đi ngắn nhất giữa các cặp đỉnh và các tính chất của đồ thị mà GS nhớ được. Để dữ liệu được gọn nhẹ, GS muốn số cạnh của đồ thị là nhỏ nhất có thể.

Các bạn hãy giúp GS nhé.

Input

Dòng đầu tiên chứa số nguyên n ($1 \leq n \leq 227$) — số đỉnh của đồ thị. Trong n dòng còn lại, dòng thứ i chứa n số nguyên $d_{i,1}, d_{i,2}, \dots, d_{i,n}$ ($0 \leq d_{i,j} \leq 109, d_{i,i} = 0, d_{i,j} = d_{j,i}$), thể hiện độ dài đường đi ngắn nhất từ đỉnh i tới n đỉnh của đồ thị.

Output

Dòng đầu tiên chứa số nguyên m ($0 \leq m \leq 105$) — số cạnh của đồ thị. m dòng tiếp theo, mỗi dòng chứa ba số nguyên u, v và c ($1 \leq u, v \leq n, 1 \leq c \leq 106$) thể hiện một cạnh nối giữa hai đỉnh u và v với trọng số c .

Dữ liệu vào đảm bảo tồn tại một đồ thị thỏa mãn các điều kiện đề ra. Nếu có nhiều đồ thị đều thỏa mãn, bạn được đưa ra một đồ thị bất kì.

Scoring

Với mỗi test, bạn được 50% số điểm nếu in ra được một đồ thị thỏa mãn các ràng buộc đề ra, nhưng không có ít cạnh nhất. Bộ test gồm hai subtask:

- Subtask 1 (40 điểm): $n \leq 6$.
- Subtask 2 (60 điểm): Không có ràng buộc gì thêm.

Examples

Input

Copy

```
3
0 1 1
1 0 1
1 1 0
```

Output

Copy

```
3
1 2 1
1 3 1
2 3 1
```

Input

Copy

```
3
0 1 2
1 0 1
2 1 0
```

Output

Copy

```
2
1 2 1
2 3 1
```

