Hai đường đi

Time Limit: 1.0s **Memory Limit:** 512M

Một mạng giao thông gồm N nút giao thông, và có M đường hai chiều nối một số cặp nút, thông tin về một đường gồm ba số nguyên dương u, v là tên hai nút đầu mút của đường, và l là độ dài đoạn đường đó. Biết rằng hai nút giao thông bất kì có không quá 1 đường hai chiều nhận chúng làm hai đầu mút.

Cho hai nút giao thông s và f, hãy tìm hai đường đi nối giữa s với f sao cho hai trên hai đường không có cạnh nào được đi qua hai lần và tổng độ dài 2 đường đi là nhỏ nhất.

Input

- Dòng đầu ghi N, M $(N \leq 100)$
- Dòng thứ 2 ghi hai số s, f.
- M dòng tiếp theo, mỗi dòng mô tả một đường gồm ba số nguyên dương u, v, l.

Output

- Dòng đầu ghi T là tổng độ dài nhỏ nhất tìm được hoặc -1 nếu không tìm được.
- Nếu tìm được, hai dòng sau, mỗi dòng mô tả một đường đi gồm: số đầu là số nút trên đường đi này, tiếp theo là dãy các nút trên đường đi bắt đầu từ s, kết thúc tại f.

Giới hạn

Phạm vi tính toán là số nguyên $32 - bit$ có dấu

Sample Input

```
5 8
1 5
1 2 1
1 4 8
2 3 5
2 4 1
3 5 1
4 3 8
4 5 1
1 3 1
```

Sample Output

```
5
3 1 3 5
4 1 2 4 5
```

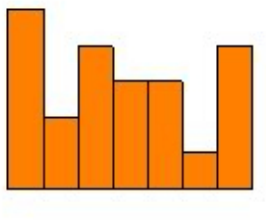
Bán dừa

Time Limit: 0.6s **Memory Limit:** 512M

Nếu các bạn biết câu chuyện thương tâm "ăn dưa leo trả vàng" của Pirate hẳn đã phải khóc hết nước mắt khi anh ấy, vì lòng thương chim, đã bán rẻ trái dưa leo siêu bự của mình.

Dưa leo cũng đã bị chim to lấy đi rồi, Pirate giờ chuyển sang nghề bán dừa để bù lỗ. Bất đắc dĩ thôi, vì trên đảo toàn là dừa...

Nhưng mà bán cái gì thì đầu tiên cũng phải có biển hiệu đã. Pirate quyết định lùng sục trên đảo các mảnh ván còn sót lại của những con tàu đắm để ghép lại thành tấm biển. Cuối cùng anh cũng tìm được N tấm ván hình chữ nhật, tấm thứ i có chiều rộng là 1 đơn vị và chiều dài là a_i đơn vị. Pirate dựng đứng chúng trên mặt đất và dán lại với nhau để được một mảnh ván to hơn (xem hình minh họa).



Việc cuối cùng chỉ là đem mảnh ván này đi cưa thành tấm biển thôi. Nhưng hóa ra đây lại là công việc khó khăn nhất. Pirate rất thích hình vuông và muốn tấm biển của mình càng to càng tốt, nhưng khổ nỗi trên đảo lại không có nhiều dụng cụ đo đạc. Không êke, không thước đo độ, nên Pirate chỉ còn cách dựa vào cạnh của N tấm ván ban đầu để cưa cho thẳng thôi. Pirate chỉ có thể cưa theo những đoạn thẳng chứa một cạnh nào đó (dọc hoặc ngang) của các tấm ván.

Hãy giúp anh ấy cưa được tấm biển lớn nhất có thể.

Input

- Dòng thứ nhất: ghi số nguyên N - số tấm ván.
- N dòng tiếp theo: mô tả độ cao của các tấm ván theo thứ tự trái sang phải sau khi đã dán lại.

Output

- Một số nguyên duy nhất là độ dài cạnh của tấm biển lớn nhất có thể cưa được.

Giới hạn

- Độ cao của các tấm ván là các số nguyên dương không vượt quá 10^9.
- $1 \leq N \leq 10^6$.
- 60% số test có $1 \leq N \leq 2000$.
- 80% số test có $1 \leq N \leq 10^5$.

Sample Input

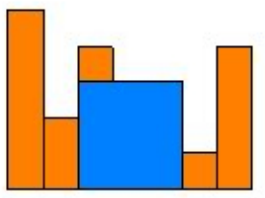
```
7
5
2
4
3
3
1
4
```

Sample Output

```
3
```

Note

Hình dưới đây minh họa phương án tối ưu.



VOI 06 Bài 5 - Mạng máy tính

Time Limit: 0.38s **Memory Limit:** 512M

Một hệ thống n máy tính (các máy tính được đánh số từ 1 đến n) được nối lại thành một mạng bởi m kênh nối, mỗi kênh nối hai máy nào đó và cho phép ta truyền tin một chiều từ máy này đến máy kia. Giả sử s và t là 2 máy tính trong mạng. Ta gọi đường truyền từ máy s đến máy t là một dãy các máy tính và các kênh nối chúng có dạng:

$s = u_1, e_1, u_2, ..., u_i, e_i, u_{i+1}, ..., u_{k-1}, e_{k-1}, u_k = t$

trong đó $u_1, u_2, ..., u_k$ là các máy tính trong mạng, e_i— kênh truyền tin từ máy u_i đến máy u_{i+1}. $(i = 1, 2, ..., k - 1)$.

Mạng máy tính được gọi là thông suốt nếu như đối với hai máy u, v bất kỳ ta luôn có đường truyền tin từ u đến v và đường truyền tin từ v đến u. Mạng máy tính được gọi là hầu như thông suốt nếu đối với hai máy u, v bất kỳ, hoặc là có đường truyền từ u đến v, hoặc là có đường truyền từ v đến u.

Biết rằng mạng máy tính đã cho là hầu như thông suốt nhưng không thông suốt.

Yêu cầu: hãy xác định xem có thể bổ sung đúng một kênh truyền tin để biến mạng đã cho trở thành thông suốt được không?

Input

- Dòng đầu tiên ghi 2 số nguyên n và m.
- Dòng thứ i trong số m dòng tiếp theo mô tả kênh nối thứ i bao gồm 2 số nguyên dương u_i và v_i cho biết kênh nối thứ i cho phép truyền tin từ máy u_i đến máy v_i, $i = 1, 2, ..., m$.

Các số trên cùng một dòng được ghi cách nhau bởi dấu cách.

Output

- Dòng đầu tiên ghi 'YES' nếu câu trả lời là khẳng định, ghi 'NO' nếu câu trả lời là phủ định.
- Nếu câu trả lời là khẳng định thì dòng thứ hai ghi hai số nguyên dương u, v cách nhau bởi dấu cách cho biết cần bổ sung kênh truyền tin từ máy u đến máy v để biến mạng thành thông suốt.

Giới hạn

Trong tất cả các test, $n \leq 2000$, $m \leq 30000$.

Sample Input

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

Sample Output

```
YES
3 1
```

Chia dãy

Time Limit: 1.0s **Memory Limit:** 512M

Dãy số M phần tử B được gọi là dãy con của dãy số A gồm N phần tử nếu tồn tại một mã chuyển C gồm M phần tử thoả mãn $B_i = A_{C_i}$ với mọi $i = 1..M$ và $1 \leq C_1 < C_2 < ... < C_m \leq N$.

Một cách chia dãy A thành các dãy con "được chấp nhận" nếu các dãy con này là các dãy không giảm và mỗi phần tử của dãy A thuộc đúng một dãy con.

Yêu cầu: Bạn hãy chia dãy con ban đầu thành ít dãy con nhất mà vẫn "được chấp nhận".

Input

Dòng đầu tiên ghi số N là số phần tử của dãy A ($N \leq 10^5$).

N dòng tiếp theo ghi N số tự nhiên là các phần tử của dãy A ($A_i \leq 10^9$).

Output

Ghi một duy nhất là số lượng dãy con ít nhất thỏa mãn.

Sample Input

```
4
1
5
4
6
```

Sample Output

```
2
```

Dãy số

Time Limit: 1.0s **Memory Limit:** 512M

Cho ba dãy số nguyên dương A = (A_1, ..., A_M) B = (B_1, ..., B_N) và C = (C_1, ..., C_P)

Hãy tìm một dãy con dài nhất gồm các phần tử liên tiếp của dãy C thỏa mãn hai điều kiện:

- Mọi phần tử của dãy A đều xuất hiện trong dãy con được chọn
- Không phần tử nào của dãy B xuất hiện trong dãy con được chọn

Input

- Dòng 1 chứa ba số nguyên dương M, N, P
- Dòng 2 chứa M số nguyên dương A_1,...,A_M
- Dòng 3 chứa N số nguyên dương B_1,...,B_N
- Dòng 4 chứa P số nguyên dương C_1,...,C_P
- Các số trong file dữ liệu đều là số nguyên dương không lớn hơn 10^5, các số trên cùng một dòng được ghi cách nhau bởi dấu cách. Dữ liệu vào đảm bảo tìm được dãy con khác rỗng gồm các phần tử liên tiếp củathỏa mãn yêu cầu đề bài.

Output

- Một số nguyên duy nhất là độ dài của dãy con tìm được

Giới hạn

- Trong 50% số test, $m,n,p \leq 1000$

Sample Input

```
3 2 11
1 2 3
5 9
1 2 9 2 2 1 4 5 3 1 2
```

Sample Output

```
3
```

Tính sai

Time Limit: 1.0s **Memory Limit:** 512M

Khi còn bé, các bạn học sinh học được cách trừ phân số bằng cách quy đồng mẫu số, rồi mới thực hiện phép trừ.

$$\frac{5}{4} - \frac{9}{12} = \frac{15}{12} - \frac{9}{12} = \frac{6}{12} = \frac{1}{2}$$

Nhưng một lần, An tính thử hiệu hai phân số bằng cách lấy hiệu hai tử số và hiệu hai mẫu số và thấy thật ngạc nhiên là kết quả vẫn đúng.

$$\frac{5}{4} - \frac{9}{12} = \frac{5-9}{4-12} = \frac{-4}{-8} = \frac{1}{2}$$

An thấy tính chất này thật kỳ diệu và An muốn biết, với phân số cho trước, có bao nhiêu cặp giá trị $a \geq 0$ và $m \geq 0$ sao cho:

$$\frac{a}{m} - \frac{b}{n} = \frac{a-b}{m-n}$$

Input

Một dòng chứa hai số nguyên dương b và n cách nhau ít nhất một dấu cách ($1 \leq b, n \leq 10^6$; trong 50% số test $b, n \leq 1000$).

Output

Một số nguyên duy nhất là số lượng cặp (a, m) tính được.

Sample Input

```
9 12
```

Sample Output

```
5
```

Bảo vệ

Time Limit: 1.0s **Memory Limit:** 512M

Một mạng lưới gồm N thành phố, và một số đường một chiều nối các cặp thành phố (giữa hai thành phố có thể có nhiều đường nối một chiều).

Quân địch đang tập trung ở thành phố N, định tiến công ta ở thành phố 1, và chúng sẽ tiến công trên tất cả các con đường chưa được bảo vệ để tiến vào thành phố 1. Bộ chỉ huy ta cần xác định số quân ít nhất trên các con đường để chặn địch tiến về thành phố 1.

Input

Dòng đầu ghi N $(N \leq 5000)$.

Các dòng tiếp theo cho đến hết file có không quá 10000 dòng, mỗi dòng một tả 1 đường gồm u, v, s cho biết có đoạn đường một chiều từ u đến v, và phải cần ít nhất s $(s \leq 65000)$ quân để chặn địch trên đường này.

Output

Số quân ít nhất cần điều động.

Sample Input

```
10
10 7 25050
6 1 12564
10 4 23916
5 1 61054
10 9 50950
9 1 35558
10 2 60941
3 1 22203
8 2 2853
5 7 31422
3 7 41491
8 7 27235
4 8 55965
8 6 41980
3 6 47707
2 3 45320
3 8 11237
7 6 38734
5 6 7561
3 5 8844
```

Sample Output

```
79169
```

Coder Rating

Time Limit: 1.0s **Memory Limit:** 512M

Cho danh sách N lập trình viên ($1 \leq N \leq 300000$), đánh số lần lượt từ 1 đến N. Mỗi người đều tham gia cả hai giải thi đấu: Giải THPT và giải Mở rộng. Với mỗi lập trình viên, bạn sẽ được cung cấp điểm số của giải Mở rộng A_i và điểm số của giải THPT H_i (Các điểm số đều là số nguyên không âm và không vượt quá 100000). Lập trình viên i được coi là giỏi hơn lập trình viên j khi và chỉ khi cả 2 điểm số của lập trình viên i đều lớn hơn hoặc bằng điểm số tương ứng của lập trình viên j, trong đó có ít nhất 1 điểm số phải lớn hơn. Hãy tính xem với mỗi lập trình viên i thì có bao nhiêu lập trình viên mà i giỏi hơn.

Input

Dòng đầu tiên chứa số nguyên N.

N dòng tiếp theo, dòng thứ $i + 1$ chứa 2 số nguyên A_i và H_i.

Output

Dòng i chứa số lượng lập trình viên mà lập trình viên i giỏi hơn.

Sample Input

```
8
1798 1832
862 700
1075 1089
1568 1557
2575 1984
1033 950
1656 1649
1014 1473
```

Sample Output

```
6
0
2
4
7
1
5
1
```

Tìm số

Time Limit: 0.38s **Memory Limit:** 512M

Cho trước một số n. Hãy tìm số nguyên dương nhỏ nhất có đúng n ước.

Input

- Một số nguyên n duy nhất ($1 \leq n \leq 1000$).
- Giới hạn: 50% số test có $n \leq 250$

Output

- Số nguyên dương nhỏ nhất (không vượt quá 10^{18}) có đúng n ước.
- Biết rằng kết quả của các test luôn nằm trong giới hạn của đề

Sample Input

```
4
```

Sample Output

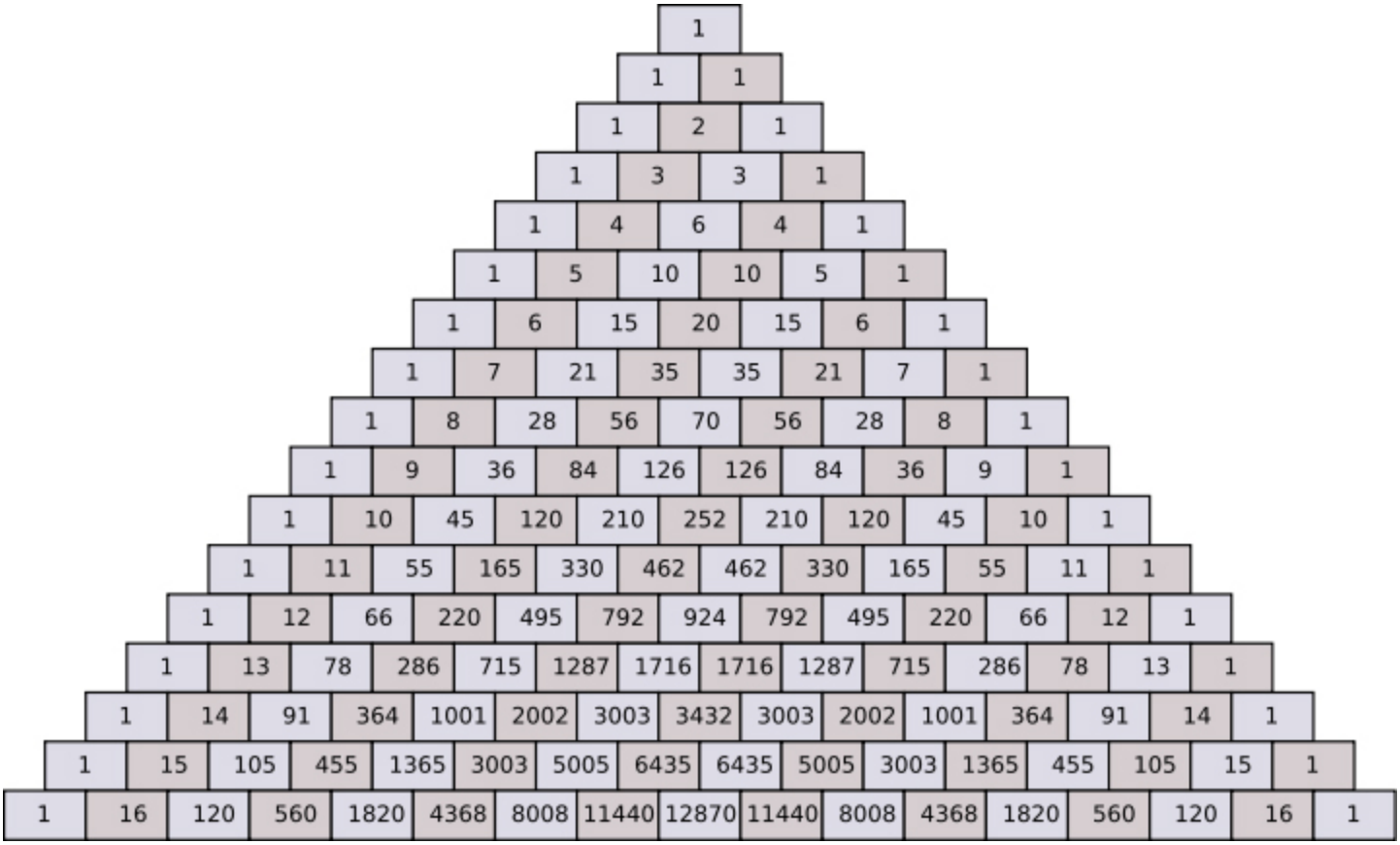
```
6
```

Ước chung lớn nhất trong tam giác Pascal

Time Limit: 1.0s **Memory Limit:** 512M

Tam giác Pascal là một cách sắp xếp hình học của các hệ số nhị thức vào một tam giác. Hàng thứ n $(n \geq 0)$ của tam giác bao gồm các hệ số trong khai triển của đa thức $f(x, y) = (x + y)^n$. Hay nói cách khác, phần tử tại cột thứ k, hàng thứ n của Tam giác Pascal là C_n^k, tức tổ hợp chập k của tập n phần tử $(0 \leq k \leq n)$.

Dưới đây là hình vẽ thể hiện các hàng từ 0 đến 16 của Tam giác Pascal:



Cho số tự nhiên n. Hãy tính $GPT(n)$ là ước chung lớn nhất của các số nằm giữa hai số 1 trên hàng thứ n của Tam giác Pascal.

Input

Dòng đầu ghi T là số lượng test. T dòng tiếp theo, mỗi dòng ghi một số nguyên n.

Output

Gồm T dòng, mỗi dòng ghi $GPT(n)$ tương ứng.

Giới hạn

- $1 \leq T \leq 20$.

- $2 \leq n \leq 10^9$.

Sample Input

```
5

2

3

4

5

6
```

Sample Output

```
2
3
2
5
1
```

Coins Game

Time Limit: 1.0s **Memory Limit:** 512M

Asen và Boyan cùng chơi trò chơi với các đồng xu sau. Chúng chọn 2 số nguyên dương K và L khác nhau và chơi trò chơi với 1 tháp gồm N xu. Asen luôn chơi trước, tiếp theo là Boyan, sau đó lại là Asen...

Mỗi lần chơi, chúng có thể lấy đi 1, K hoặc L đồng xu. Người thực hiện lượt đi cuối cùng là người chiến thắng. Asen nhận thấy có những trường hợp nó luôn có thể thắng, và các trường hợp mà Boyan có thể thắng bất kể nó chơi như nào. Do đó, trước khi bắt đầu chơi, Asen muốn biết kết quả có thể của lần chơi. Hãy viết 1 chương trình dự đoán kết quả với K, L và N cho trước.

Input

Dữ liệu vào mô tả m lần chơi. Dòng đầu tiên gồm các số K, L và m $(1 < K < L < 10, 3 < m < 50)$. Dòng thứ hai gồm m số nguyên $N_1, N_2, \ldots, N_m$ $(1 \leq N_i \leq 10^6, i = 1, 2, \ldots, m)$, là số đồng xu trong từng lần chơi.

Output

Hiển thị một xâu m kí tự 'A' và 'B', 'A' nếu Asen thắng và 'B' nếu ngược lại trong lần chơi thứ i.

Sample Input

```
2 3 5
3 12 113 25714 88888
```

Sample Output

```
ABAAB
```

Majstor

Time Limit: 1.0s **Memory Limit:** 512M

"Oẳn tù tì" là 1 trò chơi đối kháng nổi tiếng giữa 2 người. Ở mỗi cuộc đấu, người chơi được phép ra kéo, giấy hoặc nắm đấm. 2 người sẽ hoà nếu ra cùng loại; nếu không, kéo thắng giấy, giấy thắng nắm đấm và nắm đấm thắng kéo.

Svan đã học về tâm lý học trong nhiều năm và trở nên vô đối trong trò chơi này. Do vậy cậu quyết định tổ chức cuộc đấu đồng loạt với N bạn. Cuộc đấu diễn ra trong R vòng.

Điểm số của Svan với mỗi bạn là tổng điểm số cho R vòng. Điểm số cho mỗi vòng được tính: thắng 2, hoà 1 và thua 0.

Yêu cầu: Viết chương trình tính tổng điểm Svan có thể nhận được sau khi đấu với N bạn. Đồng thời, Svan có thể giành được tối đa bao nhiêu điểm nếu đoán trước được các bạn của mình sẽ ra cái gì?

Input

- Dòng 1 ghi số R $(1 \leq R \leq 50)$ là số vòng
- Dòng 2 là 1 xâu R kí tự 'S' (kéo), 'R' (nắm đấm), 'P' (giấy)
- Dòng 3 ghi số N $(1 \leq N \leq 50)$ là số bạn đấu với Svan
- N dòng tiếp theo, dòng thứ i là 1 xâu R kí tự, thể hiện biểu tượng mà bạn i sẽ ra

Output

- Dòng 1: số điểm Svan nhận được theo lối chơi lúc đầu
- Dòng 2: số điểm tối đa Svan có thể nhận được, nếu đưa ra sự điều chỉnh phù hợp (các bạn của Svan vẫn giữ nguyên lối chơi)

Sample Input 1

```
5
SSPPR
1
SSPPR
```

Sample Output 1

```
5
10
```

Sample Input 2

```
5
SSPPR
2
PPRRS
RRSSP
```

Sample Output 2

```
10
15
```

Sample Input 3

```
4
SPRS
4
RPRP
SRRR
SSPR
PSPS
```

Sample Output 3

```
12
21
```

Note

Giải thích test 2:

- Theo cách chơi ban đầu, Svan sẽ giành được 10đ khi đấu với bạn thứ nhất, và 0đ khi đấu với bạn thứ 2. Tổng cộng là 10đ.
- Để giành được số điểm tối đa, Svan sẽ chơi 'PPRRS' để giành được 5đ khi đấu với bạn thứ nhất, và 10đ khi đấu với bạn thứ hai. Tổng cộng là 15đ

Biến đổi chuỗi

Time Limit: 1.0s **Memory Limit:** 512M

Cho hai chuỗi s và t. Bạn phải biến đổi hai chuỗi này thành cùng một chuỗi. Mỗi lần biến đổi, bạn được đổi một ký tự thuộc một trong hai chuỗi trở thành ký tự trước hoặc ngay sau nó trong bảng chữ cái. Bảng chữ cái tiếp nối theo vòng tròn, nên bạn cũng có thể đổi a thành z hoặc z thành a.

Bạn cần biến đổi với số lần ít nhất có thể. Hãy trả về chuỗi kết quả. Nếu có nhiều đáp án, trả về chuỗi có thứ tự từ điển nhỏ nhất.

Input

- Mỗi test bắt đầu bằng thẻ [CASE], các test cách nhau bởi một dòng trắng. Thẻ [END] báo hiệu kết thúc file input.
- Tiếp theo là chuỗi s.
- Tiếp theo là chuỗi t.

Giới hạn:

- Chuỗi s và t có từ 1 đến 50 ký tự, và chỉ bao gồm các chữ cái in thường a..z.

Output

- Với mỗi test in ra chuỗi kết quả tìm được.

Sample Input

```
[CASE]
cat
dog

[CASE]
abcdefghijklmnopqrstuvwxyz
bcdefghijklmnopqrstuvwxyza

[CASE]
programmingcompetitionsrule
programmingcompetitionsrule

[CASE]
topcoderopen
onlinerounds

[END]
```

Sample Output

```
caa
abcdefghijklmnopqrstuvwxya
programmingcompetitionsrule
onlcndaoondn
```

Xây hàng rào

Time Limit: 0.38s **Memory Limit:** 512M

Nông dân John muốn xây một cái hàng rào có 4 mặt vây lấy đàn bò. Ông ta có một thanh gỗ có độ dài là 1 số nguyên N ($4 \leq N \leq 2500$), ông ta muốn cắt thanh gỗ này tại 3 điểm để chia thành 4 miếng nhỏ hơn, mỗi miếng có độ dài là 1 số nguyên.

4 miếng này dài ngắn thế nào cũng được miễn là có thể giúp nông dân John đóng được 1 cái hàng rào hình tứ giác là được. Hỏi có bao nhiêu cách khác nhau cắt thanh gỗ ban đầu để tạo thành được hàng rào?

- Hai cách cắt gọi là khác nhau nếu một cách có 1 nhát cắt tại 1 điểm mà cách kia không có.
- Đảm bảo rằng hàng rào này xây dựng có diện tích lớn hơn 0.
- Chú ý đáp án luôn nằm trong phạm vi 1 số nguyên 32 bit có dấu.

Input

- Dòng 1: 1 số nguyên duy nhất: N

Output

- Một số nguyên duy nhất là số cách mà nông dân John có thể cắt thanh gỗ thành 4 miếng nhỏ hơn mà có thể tạo được 1 tứ giác.

Sample Input

```
6
```

Sample Output

```
6
```

Note

Nông dân John có thể cắt thanh gỗ theo 10 cách: (1, 1, 1, 3); (1, 1, 2, 2); (1, 1, 3, 1); (1, 2, 1, 2); (1, 2, 2, 1); (1, 3, 1, 1); (2, 1, 1, 2); (2, 1, 2, 1); (2, 2, 1, 1); or (3, 1, 1, 1). Trong đó 4 cách -- (1, 1, 1, 3), (1, 1, 3, 1), (1, 3, 1, 1), và (3, 1, 1, 1) -- không thể sử dụng để tạo thành 1 tứ giác.

Xây dựng đường

Time Limit: 0.38s **Memory Limit:** 512M

Vua Peaceful vừa khai hoang một vùng đất để lập ra đất nước Peace, lúc đầu chỉ có N thành phố (được đánh số từ 1 đến N) và không có con đường nào.

Vua Peace chọn ra 4 thành phố đặc biệt để làm trung tâm kinh tế và 4 thành phố này phải được liên thông với nhau. Chi phí xây dựng các con đường không phải nhỏ vì thế nhà vua muốn sử dụng chi phí ít nhất để xây dựng các con đường sao cho 4 thành phố đặc biệt đó vẫn liên thông.

Bạn được biết chi phí ước tính để xây dựng một số con đường và bạn hãy chọn một số con đường để xây dựng để theo đúng ý nhà vua biết rằng luôn tồn tại ít nhất một phương án xây dựng đường sao cho 4 thành phố đặc biệt liên thông.

Input

Dòng đầu tiên ghi số nguyên dương N là số lượng các thành phố. ($1 \leq N \leq 100$)

Dòng thứ hai ghi 4 số nguyên là số hiệu của 4 thành phố đặc biệt.

Trong một số dòng tiếp theo, mỗi dòng ghi 3 số nguyên u, v và c với ý nghĩa muốn xây dựng một con đường hai chiều nối trực tiếp giữa 2 thành phố u và v thì chi phí là c. ($1 \leq c \leq 5000$)

Output

Gồm 1 dòng duy nhất là tổng chi phí nhỏ nhất để xây dựng hệ thống đường.

Sample Input

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

Sample Output

```
5
```

Cây P đỉnh (Cơ bản)

Time Limit: 1.0s **Memory Limit:** 512M

Cho một cây gồm N đỉnh mỗi đỉnh có 1 nhãn C_i gọi là trọng số của đỉnh i. Hãy tìm 1 cây con gồm P đỉnh sao cho tổng trọng số của cây con này là lớn nhất. Hiểu 1 cách đơn giản là tìm P đỉnh sao cho P đỉnh này liên thông và tổng trọng số là lớn nhất.

Vì bài này là ở mức độ khó tuy nhiên lại có nhiều ứng dụng nên các bạn có thể xem lời giải, download test. Nói chung cũng có thể xếp bài này vào nhóm bài "cơ bản" .

Download test và solution tại đây.

Input

- Dòng 1: 2 số nguyên dương N và P. $(1 \leq P \leq N \leq 200)$.
- Dòng 2: N số nguyên dương C_1, ..., C_n. $(-1000 \leq C_i \leq 1000)$.
- N - 1 dòng tiếp theo, mỗi dòng gồm 2 số nguyên dương u, v mô tả 1 cạnh của đồ thị.

Output

- Gồm 1 dòng ghi ra P số nguyên là chỉ số của P đỉnh được chọn.

Sample Input

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

Sample Output

```
2 3
```

Wooden Sticks

Time Limit: 1.0s **Memory Limit:** 256M

Có N đoạn gỗ. Để xử lý chúng cần thời gian để chuẩn bị :

- Thời gian chuẩn bị cho đoạn gỗ đầu tiên là 1 phút.
- Sau khi xử lý xong đoạn gỗ có chiều dài l và trọng lượng w, không mất thời gian xử lý nếu đoạn gỗ tiếp theo có độ dài l' và trọng lượng w' thỏa $l \leq l'$ and $w \leq w'$. Ngược lại mất 1 phút để chuẩn bị.

Tìm thời gian chuẩn bị ít nhất cho N đoạn gỗ.

Ví dụ có 5 đoạn $(9, 4), (2, 5), (1, 2), (5, 3)$ và $(4, 1)$, thì thời gian ít nhất là 2 vì có thể xử lý theo thứ tự như sau $(4, 1), (5, 3), (9, 4), (1, 2), (2, 5)$.

Input

Dòng đầu là số lượng test, $T \leq 100$. Mỗi test gồm:

- Dòng đầu là số nguyên dương $N \leq 5000$.
- Sau đó là N dòng gồm N cặp số nguyên dương $(l_1, w_1), (l_2, w_2), \ldots, (l_N, w_N)$, $l_i, w_i \leq 10000$, là độ dài và trọng lượng của các khối gỗ.

Output

In ra T dòng, mỗi dòng là thời gian ít nhất để chuẩn bị các đoạn gỗ của test đó.

Sample Input

```
3

5
4 9
5 2
2 1
3 5
1 4

3
2 2
1 1
2 2

3
1 3
2 2
3 1
```

Sample Output

```
2
1
3
```

VOI 07 Bài 3 - Robot cứu hỏa

Time Limit: 0.38s **Memory Limit:** 512M

Trên một mạng lưới giao thông có n nút, các nút được đánh số từ 1 đến n và giữa hai nút bất kỳ có không quá một đường nối trực tiếp (đường nối trực tiếp là một đường hai chiều). Ta gọi đường đi từ nút s đến nút t là một dãy các nút và các đường nối trực tiếp có dạng:

$s = u_1, e_1, u_2, ..., u_i, e_i, u_{i+1}, ..., u_{k-1}, e_{k-1}, u_k = t,$

trong đó $u_1, u_2, ..., u_k$ là các nút trong mạng lưới giao thông, e_i là đường nối trực tiếp giữa nút u_i và u_{i+1} (không có nút u_j nào xuất hiện nhiều hơn một lần trong dãy trên, $j = 1, 2, \ldots, k$).

Biết rằng mạng lưới giao thông được xét luôn có ít nhất một đường đi từ nút 1 đến nút n.

Một robot chứa đầy bình với w đơn vị năng lượng, cần đi từ trạm cứu hoả đặt tại nút 1 đến nơi xảy ra hoả hoạn ở nút n, trong thời gian ít nhất có thể. Thời gian và chi phí năng lượng để robot đi trên đường nối trực tiếp từ nút i đến nút j tương ứng là $t_{i,j}$ và $c_{i,j}$ $(1 \leq i, j \leq n)$. Robot chỉ có thể đi được trên đường nối trực tiếp từ nút i đến nút j nếu năng lượng còn lại trong bình chứa không ít hơn $c_{i,j}$ $(1 \leq i, j \leq n)$. Nếu robot đi đến một nút có trạm tiếp năng lượng (một nút có thể có hoặc không có trạm tiếp năng lượng) thì nó tự động được nạp đầy năng lượng vào bình chứa với thời gian nạp coi như không đáng kể.

Yêu cầu: Hãy xác định giá trị w nhỏ nhất để robot đi được trên một đường đi từ nút 1 đến nút n trong thời gian ít nhất.

Input

Dòng đầu tiên chứa một số nguyên dương n $(2 \leq n \leq 500)$;

Dòng thứ hai chứa n số, trong đó số thứ j bằng 1 hoặc 0 tương ứng ở nút j có hoặc không có trạm tiếp năng lượng $(j = 1, 2, \ldots, n)$;

Dòng thứ ba chứa số nguyên dương m $(m \leq 30\,000)$ là số đường nối trực tiếp có trong mạng lưới giao thông;

Dòng thứ k trong số m dòng tiếp theo chứa 4 số nguyên dương i, j, $t_{i,j}$, $c_{i,j}$ ($t_{i,j}, c_{i,j} \leq 10\,000$) mô tả đường nối trực tiếp từ nút i đến nút j, thời gian và chi phí năng lượng tương ứng.

Hai số liên tiếp trên một dòng trong file dữ liệu cách nhau ít nhất một dấu cách.

Output

Ghi ra số nguyên dương w tìm được.

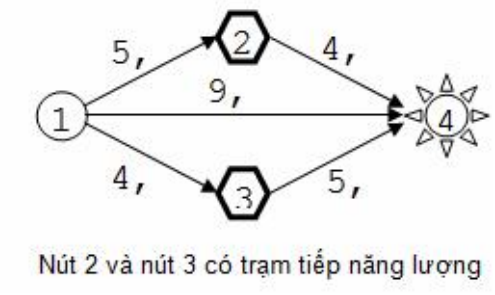
Sample Input

```
4
0 1 1 0
5
1 2 5 4
1 3 4 3
1 4 9 4
2 4 4 1
3 4 5 2
```

Sample Output

```
3
```

Note



Nút 2 và nút 3 có trạm tiếp năng lượng

Diện tích hình chữ nhật

Time Limit: 0.38s **Memory Limit:** 512M

Trên mặt phẳng toạ độ người ta vẽ ra N hình chữ nhật. Hãy tính diện tích che phủ bởi N hình chữ nhật này, biết rằng N hình chữ nhật này song song với 2 trục Ox và Oy.

Input

Dòng 1: số nguyên N ($1 \leq N \leq 10000$).

N dòng tiếp theo, mỗi dòng gồm 4 số nguyên x_1, y_1, x_2, y_2 tương ứng là toạ độ góc trái dưới và góc phải trên của hình chữ nhật thứ i. ($0 \leq x_1 \leq x_2 \leq 30000$, $0 \leq y_1 \leq y_2 \leq 30000$).

Output

Gồm 1 dòng ghi ra diện tích phủ bởi N hình chữ nhật

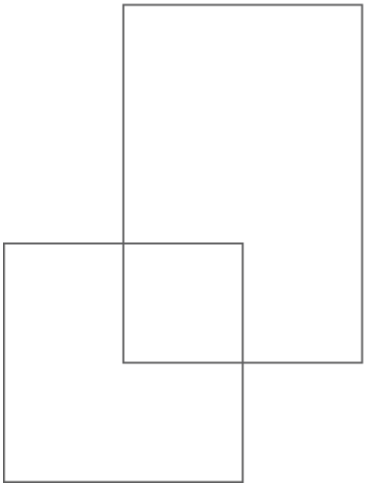
Sample Input

```
2
10 10 20 20
15 15 25 30
```

Sample Output

```
225
```

Note



Tiền tố và hậu tố

Time Limit: 0.38s **Memory Limit:** 512M

Xâu a được gọi là tiền tố của xâu b nếu xâu a trùng với phần đầu của xâu b. Ví dụ pre là tiền tố của prefix

Xâu a được gọi là hậu tố của xâu b nếu xâu a trùng với phần cuối của xâu b. Ví dụ fix là hậu tố của suffix

yenthanh132 vừa mới học về tiền tố và hậu tố nên hôm nay anh ta sẽ đố các bạn một bài toán đơn giản về tiền tố và hậu tố như sau:

- Cho 2 xâu a, b gồm các kí tự latin thường ('a' đến 'z')
- Tìm 1 xâu c thỏa mãng:
    1. Xâu a là tiền tố của xâu c
    2. Xâu b là hậu tố của xâu c
    3. Độ xài xâu c là ngắn nhất.

Input

- Dòng 1: Xâu a
- Dòng 2: Xâu b

Output

- Một dòng duy nhất là xâu c.

Giới hạn

- 40% số test có độ dài 2 xâu $a, b \leq 1000$ kí tự
- Trong toàn bộ test, độ dài 2 xâu $a, b \leq 10^5$ kí tự

Sample Input 1

```
abca
cab
```

Sample Output 1

```
abcab
```

Sample Input 2

```
abc
abc
```

Sample Output 2

```
abc
```

Chặt cây

Time Limit: 0.38s **Memory Limit:** 512M

Bạn cần chặt một thanh gỗ ra thành n đoạn, mỗi đoạn có độ dài a_i .

Các đoạn được chặt phải có độ dài theo đúng thứ tự a_1 , a_2 , ..., a_n từ trái sang phải.

Tại mỗi bước, bạn có thể chặt một nhát chia một thanh gỗ làm hai, và chi phí cho nhát chặt này bằng độ dài của thanh gỗ trước khi chặt.

Thứ tự chặt khác nhau sẽ cho ra tổng chi phí khác nhau khi chặt thanh gỗ thành n đoạn yêu cầu.

Ví dụ: bạn cần chặt một thanh gỗ độ dài 20 ra thành 4 đoạn độ dài 3, 5, 2 và 10 theo thứ tự.

Khi chặt từ trái sang phải:

- 20 chặt thành 3 và 17, chi phí 20.
- 17 chặt thành 5 và 12, chi phí 17.
- 12 chặt thành 2 và 10, chi phí 12.

Tổng chi phí: 49

Khi chặt từ phải sang trái:

- 20 chặt thành 10 và 10, chi phí 20.
- 10 chặt thành 8 và 2, chi phí 10.
- 8 chặt thành 3 và 5, chi phí 8.

Tổng chi phí: 38

Bạn hãy tìm cách chặt có tổng chi phí nhỏ nhất.

Input

- Dòng 1: n $(1 \leq n \leq 2000)$
- Dòng 2: n số nguyên dương a_1, a_2, ..., a_n, biết rằng độ dài của thanh gỗ $a_1 + a_2 + ... + a_n \leq 500000$

Output

- Một số nguyên duy nhất là chi phí nhỏ nhất tìm được.

Sample Input

```
4
3 5 2 10
```

Sample Output

```
37
```