

1. Đếm bội số(LCMs3_5.*)

Cho số nguyên dương N . Hãy đếm các số là bội của 3 hoặc 5 trong phạm vi không quá N .

Đầu vào:

- Dòng đầu chứa số nguyên T cho biết số bộ dữ liệu, mỗi bộ dữ liệu gồm 1 dòng chứa một số nguyên N .

Ràng buộc:

$$1 \leq T \leq 100; 1 \leq N \leq 10^7$$

Đầu ra:

Ứng với mỗi bộ dữ liệu test, in ra một dòng chứa kết quả bài toán.

Ví dụ:

Input	Output
3	1
3	2
5	5
10	

2. Tuổi cha và tuổi con(FatherAndSon.*)

Biết tuổi cha là x , tuổi con là y . Hỏi sau bao nhiêu năm nữa thì tuổi cha gấp đôi tuổi con?

Đầu vào:

- Dòng đầu chứa số nguyên T cho biết số bộ dữ liệu, mỗi bộ dữ liệu gồm một dòng chứa hai số nguyên x và y .

Ràng buộc: $1 \leq T \leq 10000; 1 \leq y + 20 \leq x \leq 100$

Đầu ra:

Ứng với mỗi bộ dữ liệu test, in ra một dòng là kết quả của bài toán, nếu không xảy ra thì in ra -1.

Ví dụ:

Input	Ouput
4	-1
60 40	35
71 18	20
38 9	33
41 4	

3. Thiết kế công trình Tên chương trình: PRDESIGN.*

Simbon làm việc cho một công ty chuyên về thiết kế các bản vẽ cho các công trình lớn. Một lần, Simbon nhận được một dự án lớn với một bản vẽ đã có sẵn một số dữ liệu trên đó. Dự án này muốn chọn một khu đất hình vuông lớn nhất có thể để xây dựng một công trình lớn. Tuy nhiên, trên bản vẽ có mô tả nhiều hệ thống kênh rạch ngang dọc trên bản vẽ nên rất khó khăn xác định chính xác diện tích xây dựng công trình như mong muốn.

Để tiện cho quá trình xử lý dữ liệu trên máy tính, bản vẽ có thể được xem như mặt phẳng tọa độ Oxy và hệ thống kênh rạch là n đường thẳng song song với hai trục tọa độ.

Yêu cầu: Hãy tìm khu đất hình vuông có kích thước lớn nhất trên bản vẽ đó.

Dữ liệu: Vào từ file **PRDESIGN.INP**:

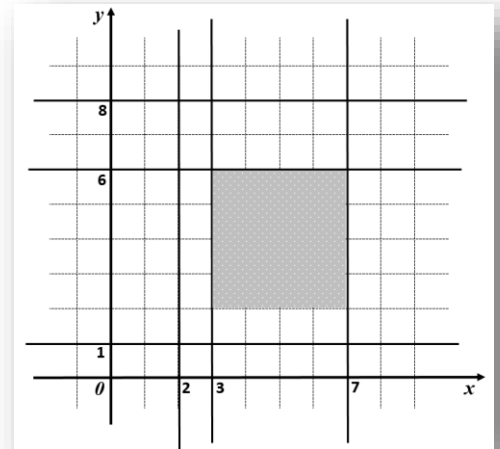
- Dòng đầu là số nguyên dương n ($3 \leq n \leq 2 \cdot 10^5$).
- Dòng thứ i trong n dòng sau chứa 2 số nguyên t_i và v_i . Trong đó:
 - $t_i = 0$ xác định đường thẳng $x = v_i$
 - $t_i = 1$ xác định đường thẳng $y = v_i$($0 \leq v_i \leq 10^9$; $i = 1, 2, \dots, n$)

Kết quả: Ghi vào file **PRDESIGN.OUT** độ dài cạnh của khu đất hình vuông lớn nhất tìm được.

Ví dụ:

PRDESIGN.INP
6
0 2
1 1
0 3
0 7
1 8
1 6

PRDESIGN.OUT
4



4. Siêu nguyên tố - Tên chương trình: SPRIME.*

Số **siêu nguyên tố** là một số nguyên tố và khi ta bỏ lần lượt các chữ số từ phải sang trái của số đó thì phần còn lại vẫn tạo thành một số nguyên tố. **Ví dụ:** 7331 là một số siêu nguyên tố có 4 chữ số vì khi bỏ 1 thì 733 là số nguyên tố, khi bỏ 13 thì 73 là số nguyên tố, khi bỏ 133 thì 7 cũng là số nguyên tố.

Cho dãy gồm N số nguyên dương a_1, a_2, \dots, a_N . Hãy lập trình cho biết trong dãy có bao nhiêu số siêu nguyên tố. Nếu trong dãy không có số nào là số siêu nguyên tố thì ghi ra 0.

Dữ liệu vào: Từ tệp **SPRIME.INP**

- Dòng đầu tiên là số nguyên dương N ($1 \leq N \leq 10^6$).
- Dòng tiếp theo ghi N số a_1, a_2, \dots, a_N ($1 \leq a_i \leq 10^6$).

Kết quả ra: Ghi ra tệp **SPRIME.OUT**

Một số duy nhất là số lượng số siêu nguyên tố có trong dãy.

Ví dụ:

SPRIME.INP	SPRIME.OUT	Giải thích
4 31 18 19 7331	2	Có 2 số siêu nguyên tố là 31 và 7331
5 13 19 18 21 52	0	Không có số siêu nguyên tố nào

Ràng buộc:

- Có 40% số test tương ứng 40% số điểm của bài với: $n \leq 10^3$; $a_i \leq 10^3$
- Có 30% số test tương ứng 30% số điểm của bài với: $n \leq 10^4$; $a_i \leq 10^6$
- Có 30% số test tương ứng 30% số điểm của bài với: $10^4 \leq n \leq 10^6$; $10^3 \leq a_i \leq 10^6$.

5. Ghép số.

Cho N số nguyên dương a_1, a_1, \dots, a_n ($1 < n \leq 100$), mỗi số không vượt quá 10^9 . Từ các số này người ta tạo ra số nguyên bằng cách ghép liên tiếp tất cả các số đã cho tạo thành một số. Hãy xác định số lớn nhất có thể tạo ra khi ghép tất cả các số đã cho tạo thành số mới.

Dữ liệu vào file NUMJOIN.INP: Dòng đầu là số n ; dòng tiếp theo là n số nguyên dương a_1, a_1, \dots, a_n .

Kết quả ra file NUMJOIN.OUT: Gồm một dòng là số lớn nhất đã ghép được.

Ví dụ:

NUMJOIN.INP	NUMJOIN.OUT
4 123 124 56 90	9056124123

6. Tìm kiếm nhị phân

Cho dãy số A gồm n phần tử ($1 < n < 10^9$) nguyên có giá trị tuyệt đối nhỏ hơn 10^9 và giá trị x nguyên có giá trị tuyệt đối nhỏ hơn 10^9 . Hãy kiểm tra xem x có trong A hay không? Dữ liệu vào file TKNP.INP: Dòng đầu số n và số x; dòng thứ hai là các giá trị của A. Kết quả ra file TKNP.OUT: Ghi vị trí xuất hiện tương ứng với câu trả lời theo yêu cầu trên; nếu không có giá trị x thì ghi 0;

TKNP.INP	TKNP.OUT
10 -2 -20 -19 -12 -2 7 17 23 29 34 41	4

7. Kết quả giải bóng đá.

Bảng kết quả của một giải vô địch bóng đá thi đấu 1 lượt trận được cho bởi ma trận vuông cấp n ($1 < n < 100$), trong đó tất cả các phần tử thuộc đường chéo chính bằng 0, các phần tử khác bằng 3, 1 hoặc 0 (đó là số điểm của trận đấu: 3 - thắng, 1 - hoà, 0 - thua). Dòng thứ i là kết quả thi đấu của đội thứ i.

a. In ra các đội có tổng điểm lớn nhất với các thông tin số thứ tự đội, tổng số điểm, số trận thắng, số trận hoà và số trận thua?

b. In ra thứ tự các đội có số trận thắng nhiều hơn số trận thua? Nếu không có in ra số 0.

c. In ra thứ tự các đội không thua trận nào? Nếu không có in ra số 0.

Dữ liệu vào file 'KQBONGDA.INP': Dòng đầu là số n; n dòng tiếp theo là bảng kết quả bóng đá.

Kết quả ra file 'KQBONGDA.OUT': Mỗi kết quả ghi trên một dòng theo thứ tự các phần đã cho.

Ví dụ:

5 0 1 0 0 0 1 0 1 3 3 3 1 0 1 0 3 0 1 0 0 3 0 3 3 0	5 9 3 0 1 2 5 2
--	-----------------------

8. Đi xe bus(Bus.*)

An thường đi học bằng xe bus. Biết rằng giá vé thông thường đi xe bus có giá a đồng/vé. Bên cạnh đó, An biết rằng công ty xe bus có bán loại vé đặc biệt cho phép đi xe bus m lần (và có thể mua nhiều lần) với giá b đồng/vé. An cần sử dụng xe bus n lần.

Yêu cầu: Hãy giúp An tính toán cách mua vé xe bus tiết kiệm nhất.

Dữ liệu vào: Đọc vào từ file Bus.inp gồm:

- + Một dòng duy nhất chứa bốn số nguyên cách nhau bởi dấu cách n, m, a, b tương ứng với số lần đi xe bus An cần đi, số lần đi xe bus bằng một vé đặc biệt, giá một vé bình thường và giá một vé đặc biệt.

Ràng buộc:

$$1 \leq n, m, a, b \leq 1000$$

Kết quả: Ghi ra file Bus.out: một số nguyên duy nhất – số tiền tối thiểu mà An sẽ phải chi tiêu.

Ví dụ:

Bus.inp	Bus.out
6 2 1 2	6
5 2 2 3	8

9. Alibaba lấy vàng(Alibaba.*)

Hôm nay, Alibaba lại vào hang của 40 tên cướp để lấy trộm vàng. Trong hang có n túi vàng, trong túi vàng thứ i có $a[i]$ đồng tiền vàng. Anh ta muốn lấy nhiều tiền vàng nhất có thể. Tuy nhiên, lần này, Thần giữ cửa hang yêu cầu Alibaba yêu cầu anh ta chỉ được lấy số tiền vàng là một số chẵn.

Yêu cầu: Hãy giúp Alibaba tính toán để lấy được số tiền vàng nhiều nhất.

Dữ liệu: đọc vào từ file Alibaba.inp gồm:

- + Dòng đầu tiên chứa một số nguyên n là số túi vàng trong hang.
- + Dòng tiếp theo chứa n số nguyên cách nhau bởi dấu cách $a[1], a[2], \dots, a[n]$ lần lượt là số tiền vàng trong mỗi túi.

Ràng buộc:

$$1 \leq n \leq 10^5; 1 \leq a[i] \leq 10^9; i = 1..n$$

Kết quả: ghi ra file Alibaba.out: Số đồng tiền vàng tối đa và là số chẵn mà Alibaba có thể lấy.

Ví dụ:

Alibaba.inp	Alibaba.out
3 1 2 3	6
5 999999999 999999999 999999999 999999999 999999999	3999999996

10. Đồng Phục Thi Đấu (UNIFORM.*)

Lê làm việc ở kênh thể thao của đài truyền hình VTV. Anh ấy thường xuyên xem bóng đá các nước. Khi xem bóng đá, anh ấy thường chú ý màu áo của mỗi đội. Anh ấy

nhận ra rằng, mỗi đội có hai màu áo, màu sân nhà và màu sân khách. Khi đội chơi ở sân nhà, các cầu thủ sẽ mặc áo màu sân nhà, ngược lại khi chơi trên sân khách sẽ mặc áo màu sân khách. Có 1 ngoại lệ: khi màu áo sân nhà của đội nhà trùng với màu áo sân khách của đội khách, đội chủ nhà phải mặc áo màu sân khách. Đồng phục sân nhà và đồng phục sân khách phải khác màu.

Có n đội tham gia giải vô địch quốc gia. Giải gồm $n*(n-1)$ trận đấu: mỗi đội sẽ gặp các đội khác trên sân của họ. Lúc này, Lê tự hỏi: có bao nhiêu lần trong suốt giải đấu: đội chủ nhà sẽ phải mặc đồng phục sân khách? Chú ý là thứ tự trận đấu sẽ không ảnh hưởng đến số lần này.

Bạn biết được màu đồng phục sân nhà và sân khách của mỗi đội. Các màu được biểu diễn bởi các số nguyên. Hai màu khác nhau khi giá trị số khác nhau.

Yêu cầu: Bạn hãy giúp Lê tìm ra câu trả lời nhé!

Dữ liệu vào: từ file **UNIFORM.INP** gồm:

- Dòng đầu: số n ($2 \leq n \leq 30$)
- N dòng sau: mỗi dòng chứa 2 số nguyên khác nhau h_i, a_i – màu áo sân nhà và sân khách của đội thứ i tương ứng. ($1 \leq h_i, a_i \leq 100$)

Kết quả: Ghi ra file **UNIFORM.OUT**: số lượng trận đấu mà đội chủ nhà phải mặc áo sân khách.

Ví dụ:

UNIFORM.INP	UNIFORM.OUT	
3 1 2 2 4 3 4	1	Giải vô địch có 6 trận, có một trận thỏa yêu cầu đề bài là trận giữa đội 2 và đội 1 trên sân đội 2.
4 100 42 42 100 5 42 100 5	5	
2 1 2 1 2	0	

11. Phương Và Những Bông Hoa(FLOWERS.*)

Phương đang làm những bó hoa cho buổi lễ Halloween sắp tới vào tháng 10. Cô ấy có một số bông hoa: r bông hoa đỏ, g bông hoa màu xanh dương và b bông hoa màu xanh

lá cây. Cô ấy muốn sử dụng những bông hoa này để làm một số bó hoa theo quy tắc. Có 4 loại bó hoa:

- Để tạo ra một "**bó hoa hồng**": cần 3 bông hoa màu đỏ.
- Để tạo ra một "**bó hoa xanh lá cây**": cần 3 hoa màu xanh lá cây.
- Để tạo ra "**bó hoa xanh lục**": cần 3 bông hoa màu xanh dương.
- Để tạo ra một "**bó mix**": cần 1 bông hoa màu đỏ, 1 màu xanh lá cây và 1 màu xanh dương

Yêu cầu: Giúp Phương tìm số lượng bó hoa tối đa mà cô ấy có thể thực hiện.

Dữ liệu vào: Đọc từ file **FLOWERS.INP**:

- Dòng đầu tiên chứa ba số nguyên **r, g** và **b** ($0 \leq r, g, b \leq 10^9$) - số hoa màu đỏ, xanh lá cây và xanh dương.

Kết quả: Ghi ra file **FLOWERS.OUT**:

- Số lượng tối đa của bó hoa Phương có thể tạo được.

Ví dụ:

FLOWERS.INP	FLOWERS.OUT
3 6 9	6