

BÀI TẬP LẬP TRÌNH

MÁY ATM

Hiện trong máy ATM chỉ còn các tờ bạc mệnh giá 5 đồng, 10 đồng và 20 đồng với số lượng tương ứng a, b, c ($1 \leq a, b, c \leq 10^5$) tờ. Một người cần rút một số tiền n đồng ($1 \leq n \leq 10^5$). Cho biết máy ATM có bao nhiêu cách khác nhau để trả đủ n đồng cho khách hàng bằng số tiền hiện có trong máy.

Yêu cầu: Cho 4 số nguyên dương a, b, c, n . Cho biết số cách trả tiền khác nhau.

Dữ liệu: Vào từ tập tin văn bản **ATM.INP** chứa 4 số nguyên dương a, b, c, n .

Kết quả: Ghi ra tập tin văn bản **ATM.OUT** số cách trả tiền tìm được.

Ví dụ:

ATM . INP	ATM . OUT
5 3 2 50	6

Giải thích: có 6 cách trả tiền: (0,1,2), (0,3,1), (2,0,2), (2,2,1), (4,1,1), (4,3,0)

MUA VÉ TIẾT KIỆM

Để khuyến khích mọi người sử dụng các phương tiện giao thông trong thành phố, ngoài việc bán lẻ với giá p_1 cho một vé thì ở hầu hết các nước đều có chế độ bán sỉ với giá p_2 cho mỗi tập gồm k vé.

Bạn đến thành phố tham quan và dự kiến sẽ đi n chuyến trên các phương tiện giao thông công cộng. Vấn đề đặt ra là nên mua vé như thế nào để tiết kiệm tiền nhất. Dĩ nhiên, bạn không hề có ý nghĩ là sẽ đi lậ vé một lần nào.

Yêu cầu: Cho các số nguyên dương n, k, p_1, p_2 các số đều có giá trị không vượt quá 10^9 . Nếu $k = 1$ thì $p_1 = p_2$. Hãy tính chi phí tối thiểu cần thiết để mua vé.

Dữ liệu: Vào từ tập tin văn bản **TICKETS.INP** 4 số nguyên n, k, p_1, p_2 .

Kết quả: Ghi ra tập tin văn bản **TICKETS.OUT** chi phí tối thiểu tìm được.

Ví dụ:

TICKETS . INP	TICKETS . OUT
12 10 17 120	154

THỨ SÁU NGÀY 13

Hành tinh Cyberplanet sử dụng lịch cũng tương tự với lịch của hành tinh chúng ta. Mỗi năm ở Cyberplanet có n tháng, mỗi tháng có đúng 30 ngày, mỗi tuần có 7 ngày và ngày không may mắn ở đây cũng là thứ sáu ngày 13. Biết rằng ngày đầu năm mới ở hành tinh Cyberplanet là ngày thứ k ($1 \leq k \leq 7$) trong tuần ($k = 1$: thứ hai; $k = 2$: thứ ba, ..., $k = 7$: chủ nhật)

Yêu cầu: Hãy xác định có bao nhiêu ngày không may mắn trong năm ở Cyberplanet

Dữ liệu: Vào từ tập tin văn bản **FRIDAY.INP** 2 số nguyên dương n, k ($1 \leq n \leq 10^9$; $1 \leq k \leq 7$)

Kết quả: Ghi ra tập tin văn bản **FRIDAY.OUT** số ngày rơi vào thứ sáu ngày 13 trong năm.

Ví dụ:

FRIDAY . INP	FRIDAY . OUT
12 1	2

TÁO QUÂN

Có m ông táo vào n bà táo được Ngọc Hoàng phân công nhiệm vụ trong năm mới. Đầu tiên Ngọc Hoàng chọn k táo (ông hoặc bà) làm những nhiệm vụ đặc biệt tại các Bộ/Ngành, sau đó Ngọc Hoàng sẽ chọn ra các nhóm, mỗi nhóm gồm đúng 2 ông táo và 1 bà táo để phân xuống các gia đình dưới hạ giới.

Yêu cầu: Hãy giúp Ngọc Hoàng xác định số nhóm nhiều nhất để phân xuống các gia đình dưới hạ giới. Ví dụ có $m = 12$ ông táo và $n = 7$ bà táo, có $k = 6$ táo phải làm nhiệm vụ đặc biệt. Ngọc Hoàng có thể chọn 4 ông táo và 2 bà táo làm nhiệm vụ đặc biệt, còn lại chia làm 4 nhóm (1 bà táo không được phân việc). Cách khác là chọn 3 ông táo và 3 bà táo làm nhiệm vụ đặc biệt, còn lại chia làm 4 nhóm (1 ông táo không được phân việc).

Dữ liệu: Vào từ tập tin văn bản **LARES.INP** gồm 1 dòng chứa 3 số nguyên dương $m, n, k \leq 10^9$ cách nhau ít nhất một dấu cách

Kết quả: Ghi ra tập tin văn bản **LARES.OUT** một số nguyên duy nhất là số nhóm nhiều nhất chọn được để phân xuống các gia đình dưới hạ giới.

Ví dụ:

LARES . INP	LARES . OUT
12 7 5	4

ĐỘ DÀI DÃY SỐ LIÊN TIẾP

Người ta viết liên tục các số tự nhiên không chứa khoảng cách từ 1 trở đi tạo thành một số tự nhiên như sau
123456789101112131415...

Yêu cầu: Cho số nguyên dương $n \leq 10^{18}$. Hỏi số tự nhiên tạo được bằng cách viết liên tục từ 1 đến n như trên có chiều dài bao nhiêu.

Dữ liệu: Vào từ tập tin văn bản **SEQLEN.INP** bàn phím số nguyên dương n .

Kết quả: Ghi ra tập tin văn bản **SEQLEN.OUT** chiều dài của số tạo được.

Ví dụ:

SEQLEN . INP	SEQLEN . OUT
15	21

TÍCH CÁC CHỮ SỐ

Cho số nguyên $n (0 \leq n \leq 10^9)$. Hãy tìm số nguyên dương m nhỏ nhất sao cho tích các chữ số của m đúng bằng n .

Dữ liệu: Vào từ tập tin văn bản **PRODIG.INP** số nguyên dương n

Kết quả: Ghi ra tập tin văn bản **PRODIG.OUT** số nguyên dương m . Nếu không tồn tại lời giải thì ghi -1 .

Ví dụ:

PRODIG . INP	PRODIG . OUT
10	25

GẤP GIẤY

Steve rất thích nghệ thuật gấp giấy origami đang được học ở trường. Một lần cậu ta gấp bài tập thực hành quá khó nên đã nghĩ ra một bài toán gấp giấy đơn giản hơn để tiêu khiển. Steve đang có mảnh giấy hình chữ nhật kích thước $W \times H$ và muốn gấp nó để thành mảnh giấy hình chữ nhật có kích thước mới $w \times h$. Mỗi lần gấp giấy phải gấp theo các đường song song với các cạnh của mảnh giấy và mảnh giấy phải luôn là hình chữ nhật.

Yêu cầu: Cho kích thước của mảnh giấy $W \times H$. Hãy tính số lần gấp tối thiểu để biến thành mảnh giấy kích thước $w \times h$

Dữ liệu: Vào từ tập tin văn bản **FOLDING.INP**

- Dòng đầu tiên chứa hai số nguyên $W, H (1 \leq W, H \leq 10^9)$ – kích thước ban đầu của mảnh giấy.
- Dòng tiếp theo chứa hai số nguyên $w, h (1 \leq w, h \leq 10^9)$ – kích thước cần gấp.

Kết quả: Ghi ra tập tin văn bản **FOLDING.OUT** một số nguyên là số lần gấp tối thiểu. Nếu không thể gấp được thì ghi -1.

Ví dụ:

FOLDING . INP	FOLDING . OUT
2 7 2 2	2
10 6 4 8	2
5 5 1 6	-1

GIAO THÔNG

Trường Nội trú cần đưa n học sinh ra phục lễ khai mạc Hội khỏe Phù Đổng. Nhà trường có thể thuê xe buýt và taxi. Mỗi xe buýt chở được không quá 50 học sinh và giá thuê là a , mỗi taxi chở không quá 4 học sinh và giá thuê là b ($a > b$). Tất cả phải khởi hành đồng thời từ trường.

Yêu cầu: Cho các số nguyên dương n, a và b ($1 \leq n \leq 10^5; 1 \leq b < a \leq 10^3$). Hãy xác định số xe buýt và số taxi cần thuê để tổng chi phí là nhỏ nhất.

Dữ liệu: Vào từ tập tin văn bản **TRANS.INP** gồm một dòng chứa 3 số nguyên n, a và b .

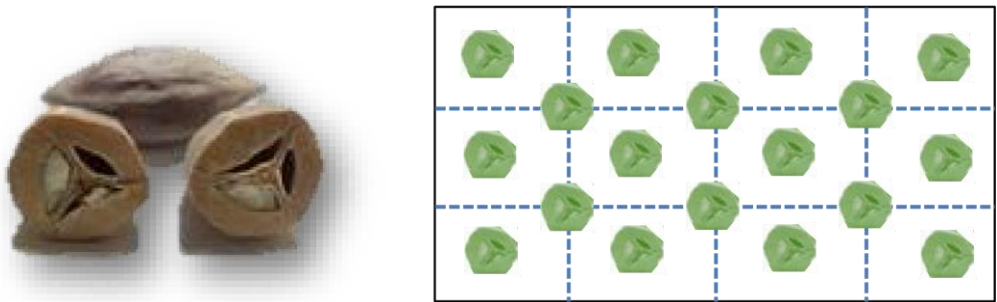
Kết quả: Ghi ra tập tin văn bản **TRANS.OUT** 2 số nguyên – số xe buýt và số taxi cần thuê.

Ví dụ:

TRANS . INP	TRANS . OUT
4 3 2	0 1

TRÁM ĐEN

Hiên, một huyện miền núi phía tây Quảng Nam cũng có trám, tuy không nhiều như ở Bắc Kạn. Các bạn Sinh viên Tình nguyện Mùa hè xanh thấy hột trám vương vãi quanh trường khá nhiều, đã nảy ra sáng kiến “trám hóa” sân trường. Có k hạt trám được thu thập về. Sân trường có hình chữ nhật. Bằng m đường cách đều nhau song song với một cạnh của sân trường và n đường cách đều nhau song song với cạnh kia của sân trường toàn bộ sân được chia thành các hình chữ nhật con giống nhau ($1 \leq m \leq n$). Các hột trám sẽ được chắt đôi. Sau khi ăn nhân bên trong học sinh sẽ đóng nửa hạt này xuống sân tại các điểm giao nhau giữa các đường kẻ và ở tâm điểm các hình chữ nhật con. Tại mỗi điểm chỉ đóng nửa hạt trám. Để không lãng phí số hạt trám đã thu nhặt và hạt trám được đóng phân bố đều trên sân, các bạn sinh viên quyết định chọn m và n sao cho số hạt trám sẽ được dùng hết và hiệu $n - m$ là nhỏ nhất.



Yêu cầu: Với k cho trước, hãy xác định m và n . Nếu không tồn tại m và n thỏa mãn thì đưa ra hai số -1

Dữ liệu: Vào từ tập tin văn bản **CANARIUM.INP** số nguyên dương $k(1 \leq k \leq 10^{14})$.

Kết quả: Ghi ra tập tin văn bản **CANARIUM.OUT** 2 số nguyên m và n (có thể là -1 -1).

Ví dụ:

CANARIUM . INP	CANARIUM . OUT
9 6	2 3 -1 -1

DÃY LIÊN TIẾP

Gọi $S(n)$ là số chữ số của số nguyên dương n . Ta viết một dãy các số nguyên liên tiếp bắt đầu từ m , nghĩa là ta có dãy $m, m + 1, m + 2, \dots$. Biết rằng, khi thêm số nguyên n vào dãy thì ta sẽ phải tốn một chi phí là $S(n) * k$. Chẳng hạn với $m = 7$ và $k = 2$ thì dãy gồm 6 phần tử: 7,8,9,10,11,12 sẽ tốn chi phí 18.

Yêu cầu: Cho biết chiều dài lớn nhất của dãy gồm các số nguyên liên tiếp bắt đầu từ m và với chi phí ban đầu w .

Dữ liệu: Vào từ tập tin văn bản **CONSEQ.INP** gồm 3 số nguyên dương $w(1 \leq w \leq 10^{16})$, $m(1 \leq m \leq 10^{16})$, $k(1 \leq k \leq 10^9)$

Kết quả: Ghi ra tập tin văn bản **CONSEQ.OUT** gồm 1 số nguyên dương là chiều dài lớn nhất của dãy tạo được.

Ví dụ:

CONSEQ . INP	CONSEQ . OUT
9 1 1	9

BẢO VỆ THÀNH PHỐ

Đất nước Wonder có n thành phố, các thành phố tọa lạc tại các vị trí có tọa độ nguyên. Chính quyền quan tâm đến sự an toàn của đất nước nên đã cho xây dựng một đường biên bao quanh đất nước sao cho tất cả thành phố đều phải nằm bên trong hoặc trên đường biên đó. Đường biên phải là một hình vuông và có các cạnh song song với các trục tọa độ.

Yêu cầu: Tìm diện tích nhỏ nhất của vùng được bảo vệ.

Dữ liệu: Vào từ tập tin văn bản **CITIES.INP**

- Dòng đầu tiên chứa số nguyên t ($1 \leq t \leq 5$) – số bộ test.
- Dòng đầu tiên của mỗi bộ test chứa số nguyên n ($1 \leq n \leq 10^5$) – số lượng thành phố.
- Mỗi dòng trong n dòng tiếp theo của bộ test chứa 2 số nguyên x_i, y_i ($|x_i|, |y_i| \leq 10^9$) – tọa độ của thành phố thứ i .

Kết quả: Ghi ra tập tin văn bản **CITIES.OUT** gồm t dòng, mỗi dòng ghi một số nguyên là diện tích nhỏ nhất của vùng được bảo vệ tương ứng.

Ví dụ:

CITIES . INP	CITIES . OUT
2	4
4	4
-1 -1	
1 1	
1 -1	
-1 1	
3	
0 0	
1 1	
2 2	

TUYẾN XE BUS

Bi lần đầu đến Mega city và muốn đón duy nhất 1 tuyến xe bus để đi từ địa điểm A sang thăm người thân sống tại địa điểm B . Hệ thống vận tải Mega city gồm n ($n \leq 500$) tuyến xe bus, mỗi tuyến có một lộ trình riêng biệt gồm từ 2 địa điểm trở lên và không quay lại địa điểm đã đi qua. Giá cước của mỗi tuyến được tính trọn gói cho hành khách lên và xuống xe tại bất kỳ 2 địa điểm nào của lộ trình (giá cước không giảm cho dù hành khách đi đoạn đường ngắn hơn trên lộ trình).

Yêu cầu: Tìm chi phí thấp nhất mà Bi phải trả để đi từ A đến B .

Dữ liệu: Vào từ tập tin văn bản **BUSROUTE.INP**

- Dòng đầu tiên chứa 3 số nguyên dương A, B, n
- Trong $2 \times n$ dòng tiếp theo mô tả từng lộ trình của các tuyến xe bus, mỗi tuyến trên 2 dòng: dòng đầu tiên chứa số nguyên c ($1 \leq c \leq 1000$) là giá cước của lộ trình và m ($1 \leq m \leq 500$) là số địa điểm của lộ trình; dòng tiếp theo chứa dãy số nguyên x_1, x_2, \dots, x_m ($1 \leq x_i \leq 100000$) là danh sách các địa điểm mà tuyến xe đi qua theo đúng thứ tự đó.

Kết quả: Ghi ra tập tin văn bản **BUSROUTE.OUT** một số nguyên là giá cước thấp nhất mà Bi phải trả hoặc -1 nếu như không tìm được lộ trình nào để đi từ A đến B chỉ bằng 1 tuyến xe duy nhất.

Ví dụ:

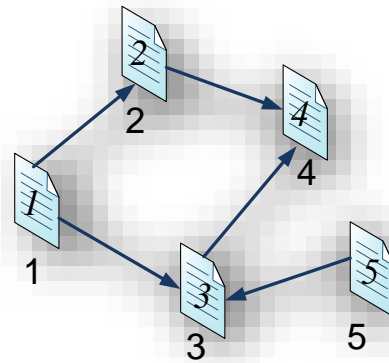
BUSROUTE . INP	BUSROUTE . OUT
1 2 3 3 3 3 2 1 4 4 2 1 4 3 8 5 4 1 7 8 2	8

BÀI TẬP VỀ NHÀ

Steve rất không thích làm bài tập ở nhà. Nhưng trong giờ Tin học thầy giáo cho tới n bài tập về nhà, trong đó có những bài chỉ có thể giải được sau khi làm xong một số bài khác.

Steve đọc đề bài, ước lượng thời gian giải cho từng bài và thấy rõ rằng mình không kịp làm được hết tất cả bài tập. Khi đó Steve quyết định sẽ bỏ một bài. Hy vọng rằng nếu chỉ thiếu có một bài thầy giáo sẽ không mắng nhiều. Vấn đề là phải chọn bài nào không làm sao cho tổng thời gian làm các bài còn lại là nhỏ nhất.

Ví dụ, với $n = 5$, thời gian làm bài thứ i là i phút và các bài 2, 3 phải làm sau khi đã làm xong bài 1, bài 3 phải làm sau bài 5. Như vậy Steve có thể bỏ bài 4 và thời gian là các bài còn lại sẽ là $1 + 2 + 3 + 5 = 11$ phút.



Yêu cầu: Cho các số nguyên n, m, t_i – thời gian làm bài thứ i và m cặp quan hệ dạng (a, b) cho biết bài b phải làm sau bài a . Hãy xác định thời gian tối thiểu cần thiết để Steve thực hiện được kế hoạch của mình.

Dữ liệu: Vào từ tập tin văn bản **HOMEWORK.INP**

- Dòng đầu tiên chứa 2 số nguyên n và m ($1 \leq n \leq 2 \times 10^5$; $0 \leq m \leq 2 \times 10^5$),
- Dòng thứ hai chứa n số nguyên t_1, t_2, \dots, t_n ($1 \leq t_i \leq 10^5$),
- Mỗi dòng trong m dòng sau chứa 2 số nguyên a và b ($1 \leq a, b \leq n$; $a \neq b$).

Kết quả: Ghi ra tập tin văn bản **HOMEWORK.OUT** thời gian tối thiểu tìm được.

Ví dụ:

HOMEWORK . INP	HOMEWORK . OUT
5 5 1 2 3 4 5 1 2 5 3 1 3 3 4 2 4	11

TỔNG

Steve mới được quà sinh nhật từ bố mẹ. Đó là một chiếc máy tính bấm tay Casio mới tinh. Sau khi được hướng dẫn cách thực hiện liên hoàn các phép tính Steve chạy về phòng mình ngồi hàng giờ để tính tổng các số nguyên liên tiếp

$$a + (a+1) + (a+2) + . . . + b$$

Steve hãnh diện cho bố mẹ xem tổng S nhận được và ngẩn người khi được hỏi tổng S được tính từ đâu tới đâu!

Yêu cầu: Cho số nguyên $S (1 \leq S \leq 10^{12})$. Hãy xác định các cặp số nguyên dương $a, b (a \leq b)$ tương ứng với S đã cho.

Dữ liệu: Vào từ tập tin văn bản **SUM.INP** gồm một dòng chứa số nguyên S .

Kết quả: Ghi ra tập tin văn bản **SUM.OUT**

- Dòng đầu tiên chứa số nguyên k – số lượng cặp số tìm được,
- Mỗi dòng trong k dòng sau chứa một cặp số nguyên a, b .

Các cặp số đưa ra theo thứ tự tăng dần của a .

Ví dụ:

SUM . INP	SUM . OUT
25	3 3 7 12 13 25 25

TRÒ CHƠI VỚI DÃY SỐ

Tí rất thích môn Số học nên thường nghĩ ra các câu đố có liên quan đến những con số để đố Tèo là em trai mình. Lần này Tí có n số nguyên dương a_1, a_2, \dots, a_n , cậu ta đố em trai thực hiện các thao tác sau trên dãy số:

- Chọn 2 số bất kỳ có giá trị khác nhau trong dãy
- Giảm giá trị của số lớn hơn đi một đại lượng bằng giá trị của số nhỏ hơn còn lại.

Tèo có thể thực hiện các thao tác trên với số lần tùy thích sao cho tổng các phần tử của dãy là nhỏ nhất có thể có.

Dữ liệu: Vào từ tập tin văn bản **NUMQUIZ.INP**

- Dòng đầu chứa số nguyên dương $n (n \leq 10^5)$
- Dòng tiếp theo chứa n số nguyên dương $a_1, a_2, \dots, a_n (1 \leq a_i \leq 10^5)$

Kết quả: Ghi ra tập tin văn bản **NUMQUIZ.OUT** số nguyên duy nhất là giá trị tổng nhỏ nhất

Ví dụ:

NUMQUIZ . INP	NUMQUIZ . OUT
2 1 2	2

LÊN DỐC

Steve cảm thấy rất mệt mỗi khi phải đi bộ lên cầu thang, trạng thái thừa cân đã ảnh hưởng nhiều đến sức khỏe. Steve quyết định tăng cường rèn luyện thể lực bằng môn thể thao mà mình ưa thích – đi xe đạp vượt địa hình.

Vùng ngoại ô gần nhà của Steve có một con đường đồi, trên đó có thể xác định n điểm, từ điểm i đến điểm $i + 1$ là đoạn chỉ xuống dốc, nằm ngang hoặc chỉ lên dốc ($1 \leq i \leq n - 1$). Điểm 1 và điểm n là các điểm đầu và cuối của con đường, h_i là độ cao tại điểm thứ i ($1 \leq i \leq n$). Steve không quan tâm lắm đến độ dài của mỗi đoạn đường mà chỉ lưu ý đến chênh lệch độ cao và cũng phải chọn các đoạn đường leo dốc, tức là đi từ thấp lên cao mới có tác dụng rèn luyện tốt. Vì vậy đường đạp xe phải là các đoạn lên dốc liên tục và chênh lệch độ cao ở các điểm đầu và cuối phải lớn nhất. Ví dụ, con đường đồi có 8 điểm với các h_i tương ứng là 12, 3,5,7,10, 6, 1,11. Các đoạn đường lên dốc liên tục là từ điểm 2 đến điểm 5 với chênh lệch độ cao là 7 và đoạn từ điểm 7 đến điểm 8 với chênh lệch độ cao là 10 và Steve sẽ chọn đường đi từ điểm 7 tới điểm 8.

Yêu cầu: Cho n và các h_i ($2 \leq n \leq 10^5$; $1 \leq h_i \leq 10^6$). Hãy xác định chênh lệch độ cao lớn nhất của đường đạp xe mà Steve có thể chọn.

Dữ liệu: Vào từ tập tin văn bản **CLIMB.INP**

- Dòng đầu tiên chứa số nguyên n ,
- Dòng thứ 2 chứa n số nguyên h_1, h_2, \dots, h_n .

Kết quả: Ghi ra tập tin văn bản **CLIMB.OUT** một số nguyên – chênh lệch độ cao lớn nhất của đường đạp xe. Nếu không tồn tại đoạn đường lên dốc nào thì đưa ra số 0.

Ví dụ:

CLIMB . INP	CLIMB . OUT
8 12 3 5 7 10 6 1 11	10

DÃY SỐ

Xét dãy số nguyên $a_1, a_2, \dots, a_n, \dots$ trong đó $a_1 = 1$, a_n được xác định như sau: đảo ngược thứ tự viết các chữ số của a_{n-1} và cộng thêm 2 vào số nhận được.

Phần đầu của dãy số này có giá trị như sau:

<i>Chỉ số</i>	1	2	3	4	5	6	7	8	...
<i>Dãy a</i>	1	3	5	7	9	11	13	33	...

Yêu cầu: Cho số nguyên dương n . Hãy xác định a_n ($1 \leq n \leq 10^{12}$).

Dữ liệu: Vào từ tập tin văn bản **SEQUENCE.INP** gồm nhiều tests, mỗi test cho trên một dòng chứa một số nguyên n .

Kết quả: Ghi ra tập tin văn bản **SEQUENCE.OUT**, kết quả mỗi test đưa ra trên một dòng.

Ví dụ:

SEQUENCE . INP	SEQUENCE . OUT
1 12	1 77

BẦU CỬ

Chính quyền đất nước Metacity đang tổ chức lấy phiếu bầu cho n ứng viên, được đánh thứ tự từ 1 đến n . Megacity có m thành phố và hệ thống bầu cử ở đất nước này được chia thành 2 giai đoạn khá lạ lùng như sau.

Giai đoạn đầu của đợt bầu cử, các lá phiếu chỉ tính cho từng thành phố, nghĩa là ứng viên được cử tri của thành phố nào bỏ nhiều phiếu nhất sẽ chiến thắng tại thành phố đó. Nếu có nhiều ứng viên cùng số phiếu cao nhất của cùng một thành phố thì ứng viên nào có thứ tự nhỏ hơn sẽ chiến thắng.

Ở giai đoạn tiếp theo, người chiến thắng được xác định theo nguyên tắc như sau: người thắng cử là người chiến thắng tại nhiều thành phố nhất. Nếu có nhiều ứng viên cùng thắng tại nhiều thành phố nhất thì ứng viên có thứ tự nhỏ hơn sẽ thắng cử.

Yêu cầu: Hãy xác định ứng viên thắng cử

Dữ liệu: Vào từ tập tin văn bản **ELECTION.INP**

- Dòng đầu tiên chứa 2 số nguyên n, m ($1 \leq n, m \leq 1000$)
- Dòng thứ i trong m dòng tiếp theo chứa n số nguyên $a_{i1}, a_{i2}, \dots, a_{in}$ ($0 \leq a_{ij} \leq 10^9$) cho biết số phiếu bầu của các ứng viên thứ ở thành phố thứ i .

Kết quả: Ghi ra tập tin văn bản **ELECTION.OUT** chỉ số của ứng viên thắng cử.

Ví dụ:

ELECTION . INP	ELECTION . OUT
3 3 1 2 3 2 3 1 1 2 1	2

SỐ THÂN THIỆN

Số tự nhiên có rất nhiều tính chất thú vị. Ví dụ với số 23, số đảo ngược của nó là 32. Hai số này có ước chung lớn nhất là 1. Những số như thế được gọi là số thân thiện, tức là số 23 được gọi là số thân thiện, số 32 cũng được gọi là số thân thiện.

Yêu cầu: Cho 2 số nguyên $a, b (10 \leq a \leq b \leq 10^5)$. Đếm số lượng số thân thiện trong đoạn giá trị $[a, b]$.

Dữ liệu: Vào từ tập tin văn bản **NUMFRE.INP** chứa 2 số a, b .

Kết quả: Ghi ra tập tin văn bản **NUMFRE.OUT** số lượng số thân thiện tìm được.

Ví dụ:

NUMFRE . INP	NUMFRE . OUT
20 30	3

BIỂU THỨC NHÂN, CỘNG

Cho n số nguyên dương $a_i, i = 1..n$, bạn phải đặt giữa n số nguyên dương này 2 phép nhân và $n - 3$ phép cộng sao cho kết quả biểu thức là lớn nhất.

Ví dụ: với $n = 5$ và dãy a_i là 4, 7, 1, 5, 3 thì bạn có thể có các biểu thức:

$$4 + 7 * 1 + 5 * 3$$

$$4 * 7 * 1 + 5 + 3$$

Chú ý: Không được thay đổi thứ tự xuất hiện của $a_i, i = 1..n$ trong biểu thức thu được.

Dữ liệu: Vào từ tập tin văn bản **MAEXP.INP**

- Dòng đầu tiên chứa số nguyên dương n ($4 \leq n \leq 10^3$)
- Dòng thứ i trong n dòng tiếp theo chứa số nguyên dương a_i ($1 \leq a_i \leq 10^4$)

Kết quả: Ghi ra tập tin văn bản **MAEXP.OUT** giá trị lớn nhất của biểu thức tìm được.

Ví dụ:

MAEXP . INP	MAEXP . OUT
5 4 7 1 5 3	44

Giải thích: biểu thức tìm được là $4 \times 7 + 1 + 5 \times 3 = 44$

MUA SÁCH

Một lời quảng cáo chào hàng trong một hiệu sách “mua 3, tặng 1, trả tiền 2”. Vì vậy, mỗi khách mua ba quyển sẽ được tặng một quyển có giá rẻ nhất trong ba quyển. Và tất nhiên, khách hàng có thể mua nhiều sách, phụ thuộc vào việc sắp xếp các quyển sách vào mỗi nhóm ba quyển để được miễn phí quyển có giá rẻ nhất trong nhóm đó.

Ví dụ, khách hàng lấy các quyển sách có giá 10, 3, 2, 4, 6, 4, 9. Nếu các quyển sách được sắp thành các nhóm: (10, 3, 2), (4, 6, 4) và (9) thì khách hàng ấy sẽ được tặng cuốn sách có giá là 2 trong nhóm một, 4 trong nhóm hai, và không có quyển sách nào được tặng trong nhóm ba vì nhóm này chỉ có 1 quyển.

Cô bán hàng là một người tốt bụng vì vậy cô ấy luôn muốn mỗi khách hàng trả ít tiền nhất có thể.

Yêu cầu: Cho giá các quyển sách, hãy giúp cô bán hàng sắp xếp các quyển sách vào các nhóm sao cho tổng số tiền khách hàng phải trả là ít nhất có thể. Chú ý cô bán hàng có thể sắp xếp các quyển sách vào các nhóm có ít nhất 1 quyển hoặc nhiều nhất 3 quyển.

Dữ liệu: Vào từ tập tin văn bản **BOOK.INP**

- Dòng đầu tiên chứa số nguyên $n (1 \leq n \leq 10^5)$ – là số sách khách hàng mua.
- Mỗi dòng trong n dòng tiếp theo chứa số nguyên $c_i (1 \leq c_i \leq 10^5)$ – là giá mỗi quyển sách.

Kết quả: Ghi ra tập tin văn bản **BOOK.OUT** một số nguyên là giá tiền nhỏ nhất mà khách hàng phải trả.

Ví dụ:

BOOK . INP	BOOK . OUT
6 3 3 4 4 3 2	14

KHÔI PHỤC NGOẶC

Một dãy dấu ngoặc hợp lệ là một dãy các ký tự "(" và ")" được định nghĩa như sau:

- Dãy rỗng (không có ký tự nào) là một dãy dấu ngoặc hợp lệ
- Nếu A là một dãy dấu ngoặc hợp lệ thì (A) là dãy dấu ngoặc hợp lệ. Dấu ngoặc mở và dấu ngoặc đóng hai bên dãy A được gọi là tương ứng với nhau
- Nếu A và B là hai dãy dấu ngoặc hợp lệ thì AB là dãy dấu ngoặc hợp lệ.

Ví dụ: ((())) (()) () () là một dãy dấu ngoặc hợp lệ. các dấu mở ngoặc ở các vị trí: 1, 2, 3, 7, 8, 11, 13 tương ứng lần lượt với các dấu đóng ngoặc ở các vị trí: 6, 5, 4, 10, 9, 12, 14.

Ban đầu có một dãy dấu ngoặc hợp lệ, người ta viết vào dưới mỗi dấu ngoặc mở một số là số dấu ngoặc (cả đóng và mở) nằm giữa dấu ngoặc mở đó và dấu ngoặc đóng tương ứng:

((()))	(())	()	()
4	2	0				2	0			0		0	

Sau đó xoá đi dãy ngoặc.

Yêu cầu: Cho biết dãy số còn lại, hãy khôi phục lại dãy ngoặc ban đầu

Dữ liệu: Vào từ tập tin văn bản **BRACKETS.INP**

- Dòng 1: Ghi số n là số phần tử của dãy số còn lại ($n \leq 10^5$)
- Dòng 2: Ghi lần lượt các số trong dãy

Kết quả: Ghi ra tập tin văn bản **BRACKETS.OUT** dãy dấu ngoặc khôi phục được.

Ví dụ:

BRACKETS . INP	BRACKETS . OUT
7 4 2 0 2 0 0 0	((())) (()) () ()

DÃ NGOẠI

Nhóm n người tổ chức một cuộc du lịch dã ngoại trên sông bằng thuyền đôi. Người thứ i trong nhóm có trọng lượng m_i . Mỗi thuyền du lịch đôi có thể chở 1 hoặc 2 người với tổng trọng lượng không quá d . Số thuyền mà hãng du lịch có là không hạn chế.

Yêu cầu: Hãy xác định số lượng thuyền ít nhất cần thuê.

Dữ liệu: Vào từ tập tin văn bản **TRIP.INP** gồm không quá 10 tests, mỗi test trên 2 dòng

- Dòng đầu tiên chứa 2 số nguyên n, d ($1 \leq n, d \leq 10^5$)
- Dòng thứ hai chứa n số nguyên m_1, m_2, \dots, m_n ($1 \leq m_i \leq d$)

Kết quả: Ghi ra tập tin văn bản **TRIP.OUT** mỗi test trên một dòng dưới dạng số nguyên

Ví dụ

TRIP . INP	TRIP . OUT
4 135	2
50 74 60 82	4
6 135	
50 120 74 60 100 82	

KHOẢNG CÁCH MANHATTAN

Cho n điểm trong mặt phẳng tọa độ $A_i = (x_i, y_i)$. Khoảng cách Manhattan giữa 2 điểm $A(x_A, y_A)$ và $B(x_B, y_B)$ có giá trị $d = |x_A - x_B| + |y_A - y_B|$.

Yêu cầu: Tính khoảng cách Manhattan lớn nhất giữa các cặp điểm đã cho.

Dữ liệu: Vào từ tập tin văn bản **MAHDIST.INP**

- Dòng đầu tiên chứa số nguyên dương $n (2 \leq n \leq 10^5)$.
- Dòng thứ i trong n dòng tiếp theo chứa 2 số nguyên là tọa độ của điểm $(x_i, y_i) (|x_i|, |y_i| \leq 10^9)$.

Kết quả: Ghi ra tập tin văn bản **MAHDIST.OUT** số nguyên là kết quả tìm được.

Ví dụ:

MAHDIST . INP	MAHDIST . OUT
5 1 2 2 3 5 9 0 -7 -1 4	21

TIỀN THƯỞNG

Sau khi chiến thắng được Boss cuối màn, Bòm được vào màn bonus. Màn hình hiện lên n thùng tiền vàng, thùng thứ i có c_i đồng tiền vàng.

Tuy nhiên, màn bonus vẫn tiếp tục thử thách Bòm theo cách khi Bòm chạm vào thùng đầu tiên có x đồng tiền vàng thì những tất cả những thùng có cùng x đồng tiền vàng đều bị biến mất khỏi màn hình, những thùng tiền còn lại là phần thưởng dành cho Bòm.

Yêu cầu: Cho n thùng chứa c_1, c_2, \dots, c_n đồng tiền vàng tương ứng. Hãy tìm thùng đầu tiên cần chọn để tổng số tiền thưởng là lớn nhất.

Dữ liệu: Vào từ tập tin văn bản **BONUS.INP**

- Dòng đầu tiên chứa số nguyên n ($1 \leq n \leq 10^5$).
- Dòng tiếp theo chứa dãy số nguyên c_1, c_2, \dots, c_n ($1 \leq c_i \leq 10^9$)

Kết quả: Ghi ra tập tin văn bản **BONUS.OUT** một số nguyên là số tiền thưởng lớn nhất.

Ví dụ:

BONUS . INP	BONUS . OUT
5 1 6 2 1 1	9

OLYMPIAD

Đội tuyển dự thi Olympiad có n học sinh. Mỗi học sinh thứ i được đặc trưng bởi 2 tham số: hệ số kỹ năng a_i và chỉ số thông minh b_i .

Giờ học được tiến hành theo kiểu sau: Giáo viên phụ trách đội tuyển lần lượt làm việc với học sinh, thảo luận, giải quyết những vấn đề nảy sinh. Kết quả là hệ số kỹ năng của học sinh được tăng thêm một lượng bằng chỉ số thông minh. Như vậy học sinh càng thông minh bao nhiêu thì hệ số kỹ năng càng tăng nhiều bấy nhiêu.

Do hạn chế về thời gian, trong suốt quá trình học, giáo viên chỉ có thể thực hiện được c lần làm việc riêng với học sinh. Một học sinh có thể được làm việc nhiều lần với giáo viên.

Số liệu thống kê cho thấy, muốn được giải trong kỳ thi thì hệ số kỹ năng phải không nhỏ hơn k .

Yêu cầu: Cho các số nguyên n, c, k và a_i, b_i ($1 \leq n \leq 10^6$; $1 \leq c, k \leq 10^9$; $0 \leq a_i, b_i \leq 10^9$). Hãy xác định số lượng tối đa học sinh sẽ được giải.

Dữ liệu: Vào từ tập tin văn bản **OLYMP.INP**

- Dòng đầu tiên chứa 3 số nguyên n, c, k
- Dòng thứ i trong n dòng sau chứa 2 số nguyên a_i, b_i .

Kết quả: Ghi ra tập tin văn bản **OLYMP.OUT** một số nguyên – số lượng tối đa học sinh sẽ được giải.

Ví dụ:

OLYMP . INP	OLYMP . OUT
3 5 6 1 1 2 1 4 2	2

VLTK

Bờm là một học sinh rất mê game. Cậu tích cực ngày đêm cày cuốc để nhân vật của mình ngày càng mạnh lên. Gần đây do thầy giáo cho khá nhiều bài tập tin học nên cậu ta buộc phải giảm bớt cày cuốc game. Tuy nhiên cậu ta lại không muốn nhân vật của mình yếu đi và có thể bị tiêu diệt trong khi làm bài tập nên phải mua vật phẩm để tăng sức mạnh cho nhân vật.

Có 2 loại vật phẩm giúp tăng sức mạnh cho nhân vật: tăng trực tiếp và tăng theo tỉ lệ phần trăm. Cụ thể, nếu nhân vật có sức mạnh là b và có n vật phẩm tăng trực tiếp có giá trị tương ứng d_1, d_2, \dots, d_n và m vật phẩm tăng theo tỉ lệ phần trăm có giá trị tương ứng p_1, p_2, \dots, p_m thì sức mạnh của nhân vật sẽ tăng lên thành $(b + d_1 + d_2 + \dots + d_n) * (100 + p_1 + p_2 + \dots + p_m) / 100$.

Nhưng không may là game giới hạn số vật phẩm được phép sử dụng tối đa là k cho cả 2 loại và mỗi vật phẩm chỉ được phép sử dụng không quá 1 lần. HungDM đang đau đầu vì không biết nên chọn mua những vật phẩm nào để có lợi nhất.

Yêu cầu: Hãy giúp Bờm chọn vật phẩm để chỉ số sức mạnh của nhân vật được tăng tối đa.

Dữ liệu: Vào từ tập tin văn bản **VLTK.INP**

- Dòng đầu tiên chứa 3 số nguyên dương b, k, n, m ($0 \leq b, k, n, m \leq 50000$)
- Dòng thứ hai chứa dãy n số nguyên d_1, d_2, \dots, d_n ($0 \leq d_i \leq 50000$)
- Dòng thứ ba chứa dãy m số nguyên p_1, p_2, \dots, p_m ($0 \leq p_i \leq 50000$)

Kết quả: Ghi ra tập tin văn bản **VLTK.OUT**

- Dòng đầu tiên ghi 2 số nd, np ($0 \leq nd \leq n; 0 \leq np \leq m; 0 \leq nd + np \leq k$) tương ứng với số vật phẩm tăng trực tiếp và tăng theo tỉ lệ phần trăm được chọn.
- Dòng thứ hai ghi chỉ số các vật phẩm loại tăng trực tiếp được chọn
- Dòng thứ ba ghi chỉ số các vật phẩm loại tăng theo tỉ lệ phần trăm được chọn

Ví dụ:

VLTK . INP	VLTK . OUT
70 3 2 2	2 1
40 30	2 1
50 40	1

NỐI DÂY

Có n đoạn dây xanh, n đoạn dây đỏ, n đoạn dây tím và n đoạn dây vàng. Độ dài các đoạn dây được cho trước.

Yêu cầu: cho số nguyên L , hãy cho biết có bao nhiêu cách chọn đúng 1 đoạn dây xanh, 1 đoạn dây đỏ, 1 đoạn dây tím và 1 đoạn dây vàng để nối lại thành một sợi dây trang trí có độ dài bằng L . Hai cách chọn được gọi là khác nhau nếu có đoạn dây được chọn trong một cách nhưng không được chọn trong cách còn lại.

Dữ liệu: Vào từ tập tin văn bản **TERA.INP**

- Dòng 1 chứa hai số nguyên dương $n \leq 1000$; $L \leq 10^9$.
- Dòng 2 chứa n số nguyên dương là độ dài n đoạn dây xanh.
- Dòng 3 chứa n số nguyên dương là độ dài n đoạn dây đỏ.
- Dòng 4 chứa n số nguyên dương là độ dài n đoạn dây tím.
- Dòng 5 chứa n số nguyên dương là độ dài n đoạn dây vàng.

Độ dài các đoạn dây không quá 10^9 .

Kết quả: Ghi ra tập tin văn bản **TERA.OUT** một số nguyên duy nhất là số cách chọn tính được.

Ví dụ:

TERA . INP	TERA . OUT
3 28 1 1 1 1 1 1 10 11 12 13 14 15	18

CHƯƠNG TRÌNH TRUYỀN HÌNH

Khi có quá nhiều kênh truyền hình với rất nhiều chương trình giải trí thú vị, bạn sẽ chọn lựa xem những chương trình nào? Đây quả là một câu hỏi khó.

Có n chương trình giải trí, chương trình thứ i ($1 \leq i \leq n$) có thời điểm bắt đầu là s_i và thời điểm kết thúc là t_i . Chương trình giải trí thứ i và chương trình giải trí thứ j ($1 \leq i < j \leq n$) được gọi là không phù hợp với nhau về lịch phát sóng nếu người xem không thể xem trọn vẹn nội dung của cả 2 chương trình giải trí này.

Nếu thời điểm kết thúc t_i của chương trình i là thời điểm bắt đầu s_j của chương trình j thì 2 chương trình này vẫn được xem là có lịch phát sóng phù hợp với nhau.

Ví dụ có 3 chương trình giải trí như sau: chương trình 1 ($s_1 = 7, t_1 = 10$), chương trình 2 ($s_2 = 12, t_2 = 15$), chương trình 3 ($s_3 = 10, t_3 = 20$). Chương trình 1 và 2 có lịch phát sóng phù hợp với nhau. Tương tự chương trình 1 và 3 cũng được xem là có lịch phát sóng phù hợp với nhau. Tuy nhiên, chương trình 2 và 3 có lịch phát sóng không phù hợp với nhau.

Yêu cầu: Cho biết kế hoạch phát sóng của n chương trình giải trí, hãy xác định có bao nhiêu cặp chương trình có lịch phát sóng không phù hợp với nhau.

Dữ liệu: Vào từ tập tin văn bản **TVSHOW.INP**

- Dòng đầu tiên chứa một số nguyên dương n ($n \leq 50000$).
- Dòng thứ i trong n dòng tiếp theo, mỗi dòng gồm 2 số nguyên dương s_i, t_i là thời điểm bắt đầu và kết thúc của chương trình giải trí thứ i ($1 \leq s_i < t_i \leq 10^5$).

Kết quả: Ghi ra tập tin văn bản **TVSHOW.OUT** một số nguyên là số lượng cặp chương trình có lịch phát sóng không phù hợp với nhau.

Ví dụ:

TVSHOW . INP	TVSHOW . OUT
3 7 10 12 15 10 20	1

THƯỞNG KẸO

Thầy giáo viết lên bảng 2 dãy số nguyên a_1, a_2, \dots, a_n và b_1, b_2, \dots, b_m . Các giá trị của mỗi dãy là phân biệt nhau. Thầy yêu cầu các học sinh chỉ ra các cặp (a_i, b_j) thỏa $a_i < b_j$, cứ mỗi cặp tìm được thì thầy sẽ thưởng cho 1 cây kẹo. Bôm rất thích ăn kẹo của thầy nên cố gắng kiểm thật nhiều kẹo.

Yêu cầu: Hãy xác định số kẹo tối đa được thưởng.

Dữ liệu: Vào từ tập tin văn bản **PTS.INP**

- Dòng đầu tiên chứa số $n, m (n, m \leq 10^5)$
- Dòng thứ hai chứa dãy số $a_1, a_2, \dots, a_n (|a_i| \leq 10^9; a_i \neq a_j, \forall i \neq j)$.
- Dòng thứ ba chứa dãy số $b_1, b_2, \dots, b_m (|b_j| \leq 10^9; b_i \neq b_j, \forall i \neq j)$.

Kết quả: Ghi ra tập tin văn bản **PTS.OUT** một số nguyên là số kẹo tối đa được thưởng.

Ví dụ:

PTS . INP	PTS . OUT
2 3 5 7 5 6 9	3

THUÊ PHÒNG

Có n khách đăng ký thuê phòng khách sạn. Hiện khách sạn đang có m phòng trống. Mỗi khách đến thuê đều đặt ra yêu cầu về kích thước phòng mong muốn, nếu không đáp ứng được yêu cầu thì khách sẽ từ chối thuê. Yêu cầu của khách thứ i gồm có kích thước phòng a_i và biên độ k . Khi đó khách thứ i chỉ chấp nhận thuê các phòng có kích thước từ $a_i - k$ đến $a_i + k$.

Yêu cầu: Cho danh sách các yêu cầu về kích thước của từng khách và kích thước của các phòng. Hãy tìm phương án phân phòng sao cho số khách thuê nhận được phòng theo đúng yêu cầu là nhiều nhất.

Dữ liệu: Vào từ tập tin văn bản **APARTMENTS.INP**

- Dòng đầu tiên chứa 3 số nguyên n, m, k ($1 \leq n, m \leq 2 \times 10^5$; $0 \leq k \leq 10^9$) – số lượng khách thuê, số phòng trống và biên độ kích thước phòng mà khách thuê chấp nhận được.
- Dòng thứ hai chứa dãy gồm n số nguyên a_1, a_2, \dots, a_n ($1 \leq a_i \leq 10^9$) – kích thước phòng thuê theo yêu cầu của từng khách.
- Dòng thứ ba chứa dãy gồm m số nguyên b_1, b_2, \dots, b_m ($1 \leq b_i \leq 10^9$) – kích thước các phòng cho thuê.

Kết quả: Ghi ra tập tin văn bản **APARTMENTS.OUT** một số nguyên là số lượng nhiều nhất các khách chấp nhận thuê phòng.

Ví dụ:

APARTMENTS . INP	APARTMENTS . OUT
4 3 5 60 45 80 60 30 60 75	2

BÁNH SINH NHẬT

Trong ngày sinh nhật của mình, Mai - một hoa khôi của trường, được các bạn nam trong trường hâm mộ tặng cho rất nhiều bánh kem. Vì số lượng bánh là rất lớn nên Mai phải sắp chồng các bánh lên nhau mới có đủ chỗ để.

Biết rằng một cái bánh kem có thể xếp chồng lên trên một cái bánh kem khác nếu như trọng lượng của nó nhỏ hơn hoặc bằng nửa trọng lượng của bánh đặt bên dưới. Vì những chiếc bánh kem rất mềm mại và mỏng manh nên Mai chỉ có thể xếp mỗi chồng tối đa 2 bánh kem (có thể có những chồng chỉ gồm 1 cái bánh kem). Chẳng có 8 bánh kem lần lượt có khối lượng 2, 5, 7, 6, 9, 8, 4, 2.



Một phương án dồn để có số chồng còn lại là ít nhất bằng 5.



Yêu cầu: Tìm phương án xếp chồng các bánh lên nhau sao cho số chồng bánh sau khi dồn lại là ít nhất.

Dữ liệu: Vào từ tập tin bản **BIRTHCAKES.INP**

- Dòng đầu tiên chứa số nguyên dương n ($n \leq 10^6$) – số bánh kem.
- Dòng tiếp theo chứa n số nguyên dương a_1, a_2, \dots, a_n ($a_i \leq 10^9$) là trọng lượng của n chiếc bánh kem tương ứng, các số cách nhau khoảng trắng.

Kết quả: Ghi ra tập tin văn bản **BIRTHCAKES.OUT** số chồng bánh ít nhất còn lại sau khi dồn.

Ví dụ:

BIRTHCAKES . INP	BIRTHCAKES . OUT
8 2 5 7 6 9 8 4 2	5

DÂY DẪN

Cho n đoạn dây điện ($1 \leq n \leq 10^5$). Đoạn thứ i có độ dài $l_i \text{ cm}$ ($0 < l_i < 10^9$). Cần phải cắt các đoạn đã cho thành các đoạn sao cho có được k đoạn dây bằng nhau. Có thể không cần cắt hết các đoạn dây đã cho. Mỗi đoạn dây bị cắt có thể có phần còn thừa khác 0.

Yêu cầu: Xác định độ dài lớn nhất của đoạn dây có thể nhận được. Nếu không có cách cắt thì đưa ra số 0.

Dữ liệu: Vào từ tập tin văn bản **WIRES.INP** gồm nhiều tests, mỗi test trên một nhóm dòng

- Dòng đầu tiên chứa 2 số nguyên n và k
- Dòng thứ i trong n dòng tiếp theo chứa số nguyên l_i

Kết quả: Ghi ra tập tin văn bản **WIRES.OUT**, kết quả mỗi test đưa ra trên một dòng dưới dạng số nguyên

Ví dụ:

WIRES . INP	WIRES . OUT
4 11 802 743 547 539	200

DÃY SỐ HAMMING

Dãy số nguyên dương tăng dần, trong đó ước nguyên tố của mỗi số không quá 5 được gọi là dãy Hamming. Như vậy, $10 = 2 \times 5$ sẽ là một số trong dãy Hamming, còn $26 = 2 \times 13$ – không thuộc dãy Hamming.

Phần đầu của dãy Hamming là 1, 2, 3, 4, 5, 6, 8, 9, 10, 12, 15, ...

Yêu cầu: Cho số nguyên $x (1 \leq x \leq 10^{18})$. Hãy xác định số thứ tự của x trong dãy Hamming.

Dữ liệu: vào từ tập tin văn bản **HAMMING.INP**:

- Dòng đầu tiên chứa số nguyên t – số lượng tests ($1 \leq t \leq 10^5$)
- Mỗi dòng tiếp theo chứa một số nguyên x

Kết quả: ghi ra tập tin văn bản **HAMMING.OUT**, kết quả mỗi test đưa ra trên một dòng dưới dạng số nguyên hoặc thông báo **Not in sequence**.

Ví dụ:

HAMMING . INP	HAMMING . OUT
11	1
1	2
2	6
6	Not in sequence
7	7
8	8
9	9
10	Not in sequence
11	10
12	Not in sequence
13	Not in sequence
14	