

Mục lục

Phương pháp sinh xâu nhị phân — BINARYGEN	1
Phương pháp sinh chuỗi tổ hợp — COMBINATIONGEN	2
Phương pháp sinh hoán vị — PERMUTATIONGEN	3
Liệt kê xâu nhị phân — BINARYLIST	4
Liệt kê tổ hợp — COMBINATIONLIST	5
Liệt kê hoán vị — PERMUTATIONLIST	6
Bản vanxơ Fibonacci — FIBVAL	7
Dãy Fibonacci — FIBSEQ	8
Dãy đèn — LIGHT	9
Tìm lũy thừa — POWER	11

Bài A. Phương pháp sinh xâu nhị phân

File dữ liệu vào: `stdin`
File kết quả: `stdout`
Hạn chế thời gian: 0.1 giây
Hạn chế bộ nhớ: 256 MB

Cho 1 một xâu nhị phân S độ dài n . Tìm xâu nhị phân kế tiếp của xâu S trong thứ tự từ điển.

Dữ liệu vào

20 test

Dòng đầu 1 số nguyên dương $n \leq 10^4$ Dòng thứ 2 ghi n số 0 hoặc 1 liên tiếp nhau.

Kết quả

Ghi ra xâu nhị phân kế tiếp của xâu S trên một dòng duy nhất. Nếu không tồn tại thì ghi ra -1.

Ví dụ

<code>stdin</code>	<code>stdout</code>
5 00100	00101

Bài B. Phương pháp sinh chuỗi tổ hợp

File dữ liệu vào: `stdin`
File kết quả: `stdout`
Hạn chế thời gian: 0.1 giây
Hạn chế bộ nhớ: 256 MB

Cho 1 một chuỗi tổ hợp C độ dài m với các thành phần nằm trong tập $\{1, 2, \dots, n\}$. Tìm chuỗi tổ hợp kế tiếp của chuỗi C trong thứ tự từ điển.

Dữ liệu vào

20 test

Dòng đầu 2 số nguyên dương $n, m \leq 10^4$ Dòng thứ 2 ghi m số nguyên dương $\leq n$ cách nhau bởi dấu cách.

Kết quả

Ghi ra chuỗi C trên một dòng duy nhất, các thành phần cách nhau bởi dấu cách. Nếu không tồn tại thì ghi ra -1.

Ví dụ

stdin	stdout
5 3 2 3 5	2 4 5

Bài C. Phương pháp sinh hoán vị

File dữ liệu vào: `stdin`
File kết quả: `stdout`
Hạn chế thời gian: 1 giây
Hạn chế bộ nhớ: 256 MB

Cho 1 một hoán vị H độ dài n với các thành phần nằm trong tập $\{1, 2, \dots, n\}$. Tìm hoán vị kế tiếp của hoán vị H trong thứ tự từ điển.

Dữ liệu vào

20 test

Dòng đầu ghi 1 số nguyên dương $n \leq 10^4$ Dòng thứ 2 ghi n số nguyên dương $\leq n$ cách nhau bởi dấu cách là hoán vị H .

Kết quả

Ghi ra hoán vị H trên một dòng duy nhất, các thành phần cách nhau bởi dấu cách. Nếu không tồn tại thì ghi ra -1.

Ví dụ

stdin	stdout
5 3 2 1 5 4	3 2 4 1 5

Bài D. Liệt kê xâu nhị phân

File dữ liệu vào: `stdin`
File kết quả: `stdout`
Hạn chế thời gian: 0.1 giây
Hạn chế bộ nhớ: 256 MB

Cho 1 số nguyên dương n . Đưa ra xâu nhị phân độ dài n thứ k trong thứ tự từ điển mà không có i số 0 liên tiếp.

Dữ liệu vào

Dòng đầu ghi 3 số nguyên dương $n, k, i \leq 10^4$ cách nhau bởi dấu cách

Kết quả

Ghi ra xâu nhị phân độ dài n thứ k mà không có i số 0 liên tiếp trên một dòng duy nhất, các thành phần cách nhau bởi dấu cách. Nếu không tồn tại thì ghi ra -1.

Ví dụ

<code>stdin</code>	<code>stdout</code>
6 4 2	0 1 1 0 1 0

Bài E. Liệt kê tổ hợp

File dữ liệu vào: `stdin`
File kết quả: `stdout`
Hạn chế thời gian: 0.1 giây
Hạn chế bộ nhớ: 256 MB

Cho 2 số nguyên dương n, m . Đưa ra chuỗi tổ hợp chập m của n phần tử trong tập $\{1, 2, \dots, n\}$ thứ k trong thứ tự từ điển.

Dữ liệu vào

20 test

Dòng đầu ghi 3 số nguyên dương n, m, k cách nhau bởi dấu cách, $n, m \leq 10^4; k \leq 10^9$.

Kết quả

Ghi ra chuỗi tổ hợp chập m của n phần tử thứ k trên một dòng duy nhất, các thành phần cách nhau bởi dấu cách. Nếu không tồn tại thì ghi ra -1.

Ví dụ

<code>stdin</code>	<code>stdout</code>
7 3 6	1 3 4

Bài F. Liệt kê hoán vị

File dữ liệu vào: `stdin`
File kết quả: `stdout`
Hạn chế thời gian: 0.1 giây
Hạn chế bộ nhớ: 256 MB

Cho 1 một số n . Đưa ra hoán vị độ dài n thứ k trong thứ tự từ điển.

Dữ liệu vào

20 test

Dòng đầu ghi 2 số nguyên dương n, k ($n \leq 10^4, k \leq 10^9$) cách nhau bởi dấu cách.

Kết quả

Ghi ra hoán vị thứ k trên một dòng duy nhất, các thành phần cách nhau bởi dấu cách. Nếu không tồn tại thì ghi ra -1.

Ví dụ

<code>stdin</code>	<code>stdout</code>
3 4	2 3 1

Bài G. Bản vanxơ Fibonacci

File dữ liệu vào: `stdin`
File kết quả: `stdout`
Hạn chế thời gian: 0.1 giây
Hạn chế bộ nhớ: 256 MB

Bản vanxơ Fibonacci là một bản nhạc mà giai điệu của nó bắt nguồn từ một trong những dãy số nổi tiếng nhất trong Lý thuyết số - dãy số Fibonacci. Hai số đầu tiên của dãy là số 1 và số 2, các số tiếp theo được xác định bằng tổng của hai số liên tiếp ngay trước nó trong dãy.

Bản vanxơ Fibonacci thu được bằng việc chuyển dãy số Fibonacci thành dãy các nốt nhạc theo qui tắc chuyển một số nguyên dương thành nốt nhạc sau đây:



- số 1 tương ứng với nốt Đô (C),
- số 2 tương ứng với nốt Rê (D),
- số 3 tương ứng với nốt Mi (E),
- số 4 tương ứng với nốt Fa (F),
- số 5 tương ứng với nốt Sol (G),
- số 6 tương ứng với nốt La (A),
- số 7 tương ứng với nốt Si (B),
- số 8 tương ứng với nốt Đô (C),
- số 9 tương ứng với nốt Rê (D)

và cứ tiếp tục như vậy. Ví dụ, dãy gồm 6 số Fibonacci đầu tiên 1, 2, 3, 5, 8 và 13 tương ứng với dãy các nốt nhạc C, D, E, G, C và A. Để xây dựng nhịp điệu vanxơ người ta đi tìm các đoạn nhạc có tính chu kỳ trong bản vanxơ Fibonacci. Đoạn nhạc được gọi là có tính chu kỳ nếu như có thể chia nó ra thành $k \geq 2$ đoạn giống hệt nhau. Ví dụ, đoạn nhạc GCAGCA là đoạn có tính chu kỳ, vì nó gồm hai đoạn giống nhau GCA. Yêu cầu: Cho trước hai số nguyên dương u, v ($u < v$), hãy xác định độ dài đoạn nhạc dài nhất có tính chu kỳ của bản nhạc gồm dãy các nốt nhạc của bản vanxơ Fibonacci bắt đầu từ vị trí u kết thúc ở vị trí v .

Dữ liệu vào

Dòng thứ nhất chứa số nguyên dương k ($k \leq 100$) là số lượng test;

Dòng thứ i trong số k dòng tiếp theo chứa hai số nguyên dương u_i, v_i được ghi cách nhau bởi dấu cách ($u_i < v_i \leq 10^9$) là vị trí bắt đầu và kết thúc của một bản nhạc;

Kết quả

Ghi ra k dòng, dòng thứ i chứa một số nguyên là độ dài đoạn nhạc tìm được tương ứng với test thứ i . Nếu không tìm được đoạn nào có tính chu kỳ thì ghi ra số -1.

Ví dụ

<code>stdin</code>	<code>stdout</code>
2	-1
1 3	2
4 10	

Hạn chế

Ràng buộc: 50% số tests ứng với 50% số điểm của bài có $u_i < v_i \leq 100$.

Bài H. Dãy Fibonacci

File dữ liệu vào: **stdin**
File kết quả: **stdout**
Hạn chế thời gian: 1 giây
Hạn chế bộ nhớ: 256 MB

Năm 1202, Leonardo Fibonacci, nhà toán học người Ý, tình cờ phát hiện ra tỉ lệ vàng 0.618 được tiệm cận bằng thương của hai số liên tiếp trong một loại dãy số vô hạn được một số nhà toán học Ấn Độ xét đến từ năm 1150. Sau đó dãy số này được đặt tên là dãy số Fibonacci $\{F_i : i = 1, 2, \dots\}$, trong đó $F_1 = F_2 = 1$ và mỗi số tiếp theo trong dãy được tính bằng tổng của hai số ngay trước nó. Đây là 10 số đầu tiên của dãy Fibonacci: 1, 1, 2, 3, 5, 8, 13, 21, 34, 55. Người ta đã khám phá ra mối liên hệ chặt chẽ của số Fibonacci và tỉ lệ vàng với sự phát triển trong tự nhiên (cánh hoa, cành cây, vân gỗ), trong vũ trụ (hình xoáy tròn ốc dãi ngân hà, khoảng cách giữa các hành tinh), hay sự cân đối của cơ thể con người. Đặc biệt số Fibonacci được ứng dụng mạnh mẽ trong kiến trúc (Kim tự tháp Ai Cập, tháp Eiffel), trong mỹ thuật (các bức tranh của Leonardo da Vinci), trong âm nhạc (các bản giao hưởng của Mozart) và trong nhiều lĩnh vực khoa học kỹ thuật.

Trong toán học, dãy Fibonacci là một đối tượng tổ hợp quan trọng có nhiều tính chất đẹp. Có nhiều phương pháp hiệu quả liệt kê và tính các số Fibonacci như phương pháp lặp hay phương pháp nhân ma trận.

Sau khi được học về dãy số Fibonacci, Sơn rất muốn phát hiện thêm những tính chất của dãy số này. Vì thế Sơn đặt ra bài toán sau đây: Hỏi rằng có thể tìm được một tập con các số trong n số Fibonacci liên tiếp bắt đầu từ số thứ i , sao cho tổng của chúng chia hết cho một số nguyên dương k ($k \leq n$) cho trước hay không? Nhắc lại, một tập con q số của một dãy n số là một cách chọn ra q số bất kỳ trong số n số của dãy đó, mỗi số được chọn không quá một lần.

Yêu cầu: Hãy giúp Sơn giải quyết bài toán đặt ra.

Dữ liệu vào

- Dòng thứ nhất ghi số nguyên dương T ($T \leq 10$) là số lượng bộ dữ liệu;
- Mỗi dòng trong T dòng tiếp theo chứa ba số nguyên dương n, i và k là thông tin của một bộ dữ liệu.

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

Kết quả

Ghi ra T dòng tương ứng với kết quả của T bộ dữ liệu đầu vào, mỗi dòng có cấu trúc như sau: Đầu tiên ghi số nguyên q là số lượng các số trong tập con tìm được, tiếp đến ghi q số nguyên là các số thứ tự trong dãy Fibonacci của q số trong tập con tìm được. Nếu không tìm được tập con thỏa mãn điều kiện đặt ra thì ghi ra một số 0.

Nếu có nhiều cách chọn thì chỉ cần đưa ra một cách chọn bất kỳ.

Ví dụ

stdin	stdout
1 10 3 9	2 5 7

Giải thích

Giải thích: Trong ví dụ trên một tập con thỏa mãn điều kiện đặt ra là tập gồm 2 số $F_5 = 5, F_7 = 13$ với tổng bằng 18.

Hạn chế

- Có 20% số lượng test thỏa mãn điều kiện: $n \leq 20, i \leq 10^6$;
- Có 20% số lượng test khác thỏa mãn điều kiện: $n \leq 10^3, i \leq 10^6$;
- Có 20% số lượng test khác thỏa mãn điều kiện: $n \leq 10^6, i \leq 10^6$;
- Có 10% số lượng test khác thỏa mãn điều kiện: $n \leq 20, i \leq 10^{15}$;
- Có 10% số lượng test khác thỏa mãn điều kiện: $n \leq 10^3, i \leq 10^{15}$;
- 20% số lượng test còn lại thỏa mãn điều kiện: $n \leq 10^6, i \leq 10^{15}$.

Bài I. Dây đèn

File dữ liệu vào: `light.inp`
File kết quả: `light.out`
Hạn chế thời gian: 0.1 giây
Hạn chế bộ nhớ: 256 MB

Hoàng mới mua được một dây đèn trang trí gồm n bóng đèn được mắc nối tiếp. Các bóng đèn được đánh số từ 1 đến n theo thứ tự từ bóng đèn gần phích cắm đến bóng đèn xa phích cắm hơn. Sau một thời gian sử dụng, Hoàng phát hiện ra qui luật hoạt động của dây đèn. Khi đang cắm điện chỉ có đúng k bóng đèn là sáng. Cứ sau mỗi giây, dây đèn lại chuyển trạng thái theo qui tắc sau đây:

- Thoạt tiên lúc mới cắm dây đèn vào ổ điện, các bóng đèn từ 1 đến k sáng, các bóng đèn còn lại tắt. Đây là trạng thái đầu tiên của dây đèn.
- Trạng thái cuối cùng của dây đèn là trạng thái trong đó k bóng đèn cuối cùng sáng, các bóng đèn còn lại tắt. Từ trạng thái này, ở giây tiếp theo dây đèn lại lặp lại từ trạng thái đầu tiên.
- Từ một trạng thái trung gian chưa phải là trạng thái cuối cùng, dây đèn chuyển trạng thái theo qui tắc sau:

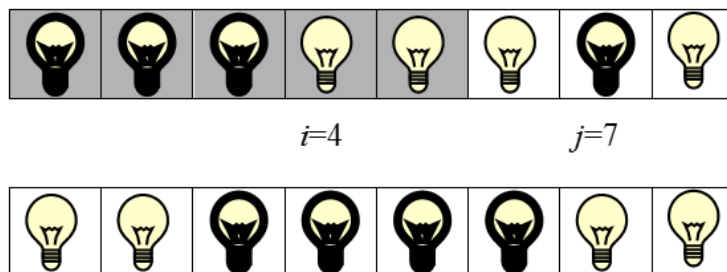
Bước 1. Tìm i là vị trí bóng đèn sáng gần phích cắm nhất.

Bước 2. Tìm $j > i$ là vị trí bóng đèn tắt gần i nhất.

Bước 3. Lật ngược trạng thái bật tắt của hai bóng đèn thứ $j - 1$ và j .

Bước 4. Nếu $j > 3$, trạng thái của dây bóng đèn từ 1 đến $j - 2$ thu được bằng cách đảo ngược thứ tự dây trạng thái hiện tại của dây bóng đèn từ 1 đến $j - 2$.

Ví dụ: Hình vẽ sau đây minh họa hai trạng thái liên tiếp của dây gồm $n = 8$ bóng đèn và $k = 4$:



Phát hiện ra qui luật hoạt động của dây bóng đèn, Hoàng muốn xác định trạng thái của dây bóng đèn ở giây thứ t kể từ lúc bắt đầu hoạt động. Yêu cầu: Cho ba số nguyên dương n, k và t , hãy đưa ra trạng thái của dây bóng đèn ở giây thứ t kể từ lúc bắt đầu hoạt động.

Dữ liệu vào

Dòng đầu tiên chứa p là số lượng bộ dữ liệu;

Mỗi dòng trong p dòng tiếp theo mô tả một bộ dữ liệu gồm ba số nguyên dương n, k và t ($k < n \leq 100; t \leq 2^{100}$). Các số trên cùng dòng ghi cách nhau bởi dấu cách.

Kết quả

Ghi ra p dòng, mỗi dòng là một xâu S độ dài n mô tả trạng thái của dây bóng đèn ở giây thứ t là kết quả tương ứng với một bộ dữ liệu theo thứ tự xuất hiện trong file dữ liệu vào, trong đó ký tự thứ j của xâu S là B nếu bóng đèn thứ j sáng, và là T nếu bóng đèn thứ j tắt.

Ví dụ

light.inp	light.out
2	TBTTB
5 2 8	BTTTTB
10 4 2011	BTTTBT

Bài J. Tìm lũy thừa

File dữ liệu vào: `stdin`
File kết quả: `stdout`
Hạn chế thời gian: 0.1 giây
Hạn chế bộ nhớ: 256 MB

Số nào là lũy thừa của a đầu tiên mà có k chữ số trong hệ cơ số b ?

Dữ liệu vào

Trên một dòng duy nhất 3 số a, k, b cách nhau một dấu cách .

Kết quả

Ghi duy nhất một số nguyên là kết quả tìm được. Nếu không có đáp án thì in ra -1.

Ví dụ

<code>stdin</code>	<code>stdout</code>
17 2 10	1

Hạn chế

- Subtask 1: $a \leq 20; 0 < k < 500; 2 \leq b \leq 100$.
- Subtask 2: $a \leq 20; 0 < k < 100000; 2 \leq b \leq 100$.