Bài A. ANDGAME

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

tags: NIM, combinatorics

Xem mỗi số nguyên có k bit 1 là một đống sỏi có k viên. Bài toán trở thành bốc sỏi thông thường, trạng thái thua khi và chỉ khi tổng nim (xor) của số bit 1 của các số trong dãy là bằng 0.

Chứng minh:
Gọi k_i là số bit 1 của a_i, đặt $K = k_1 \wedge k_2 \wedge \ldots \wedge k_n$.

- Nếu $K = 0$ thì hoặc là không có cách đi hợp lệ nữa, ngược lại thì một cách đi hợp lệ bất kỳ đều sẽ thay đổi đúng một số k_i, do đó K thay đổi và trở thành khác 0.
- Nếu $K \neq 0$, gọi t là bit 1 cao nhất của K, khi đó tồn tại k_i có bit 1 ở vị trí t. Ta sẽ giảm k_i để đưa K về 0 bằng cách thay k_i bằng $k'_i = k_i \wedge K$, vì t là bit 1 cao nhất của K nên k_i bị tắt bit t và các bit cao hơn t giữ nguyên, do đó $k'_i < k_i$. Đây là một cách đi hợp lệ và đưa về $K' = 0$.

Phần còn lại trong bài là bài toán tổ hợp: Có những cách nào chọn k_i, và có bao nhiêu cách chọn x để thu được k'_i.

Bài B. LNET

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

tags: max clique, graph theory, heavy light

Đây là bài toán tô màu trên đồ thị subtree intersection. Số màu cần dùng để tô cho đồ thị này bằng số đỉnh của đồ thị con đầy đủ lớn nhất (perfect graph). Do đó bài toán đưa về tìm đồ thị con đầy đủ lớn nhất online.

Nhận xét: Nếu k kết nối đôi một xung đột thì có ít nhất một đỉnh chung của cả k kết nối này.

Gọi $cap(x)$ là số kết nối đi qua đỉnh x, khi đó mỗi thay đổi là một phép tăng/giảm cap của các đỉnh trên đường đi từ s đến t, và truy vấn là hỏi max cap tất cả các đỉnh. $\Rightarrow$ Dùng prioritty_queue để quản lý max cap, mỗi nút của prioritty_queue là một segment tree quản lý một block phân rã bởi heavy light. (Cũng có thể thay thế rioritty_queue bằng một cấu trúc dữ liệu khác có thể thay đổi một phần tử và hỏi max, ví dụ segment tree)

Bài C. HALFGAME

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

tags: game theory, grundy

Tổng quát hơn bài ANDGAME, k_i bây giờ là giá trị hàm grundy của đống sỏi có a_i viên, ký hiệu là $g(a_i)$. Hàm g được tính như sau: $g(x) = \text{MEX}(\{g(x-y) | 1 \le y \le (x+1)/2\})$, với $\text{MEX}(S)$ là số tự nhiên nhỏ nhất không xuất hiện trong tập S.

Lúc này trạng thái thua khi và chỉ khi $g(a_1) \wedge g(a_2) \wedge \ldots \wedge g(a_n) = 0$ (chứng minh tương tự bài ANDGAME).

Phần còn lại trong bài là tìm cách tính nhanh hàm g, dựa vào quy luật của nó (và chứng minh).

Bài D. ARRCNT

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

tags: combinatorics, math, FFT

Suntask 1: Dễ thấy nếu trong a có hai số khác 0 mà cách nhau quá n thì kết quả là 0. Vậy nếu trong dãy có một số $\leq$ n thì mọi số đều $<$ 2n. Còn nếu các số khác 0 trong dãy đều lớn hơn n thì trừ đều các số đó đi một lượng (sao cho số nhỏ nhất bằng n), lúc này các số trong a bị giới hạn trong 1..2n. Gọi dp(i, p) là số dãy mà đến vị trí i có số liền trước là $a_{i-1} = p$, bài toán được giải quyết trong $n^2 \times 3$

Subtask 2: Tách dãy số thành các đoạn 0 liên tiếp và xử lý trên từng đoạn. Kết quả là tích của các kết quả này. Mỗi đoạn số 0 liên tiếp sẽ ứng với một bài toán tổ hợp: Đếm số dãy x độ dài L, chỉ gồm $-1, 0, +1$, có tổng bằng S. Do $a_i \geq n$ nên ta không cần quan tâm đến ràng buộc $a_i + x > 0$.

Subtask 3: Mỗi dãy 0 liên tiếp đưa về bài toán đếm số đường đi không chạm vào 0. Do tính chất bằng 1, số đường đi sau khi biết các vị trí đi ngang là số catalan.

Subtask 4: Mỗi dãy 0 liên tiếp đưa về bài toán đếm số đường đi không chạm vào 0. Ánh xạ tập các đường vi phạm sang một tập khác để đếm: Mỗi đường vi phạm, ta đảo ngược dấu từ vị trí chạm 0 đầu tiên về sau, sẽ thu được một đường khác. Đây là một song ánh.

Subtask 5: Có hai đoạn 0 ở đầu và cuối dãy, hai subtask trên không giải quyết được. Đặt $P(x) = x^s(1 + x + 1/x)^L$ với s là a_i đầu đoạn, L là độ dài đoạn. Ta có thể dùng FFT để tính nhanh đa thức này (FFT log lần, đpt 2nlogn). Sau đó có thể dùng các hệ số của đa thức để đếm nhanh số ánh xạ như đề cập ở subtask 4. Kết quả là tổng các hệ số dương trừ đi tổng các hệ số âm của đa thức.