

# PREVOI ONLINE TRAINNING 2022 - CONTEST #4

Teacher: Đỗ Phan Thuận  
dophanthuan@gmail.com

Teaching Assitant: Nguyễn Đức Thuận  
thuanbn03@gmail.com

Khoa Học Máy Tính  
Đại học Bách Khoa Hà Nội



Ngày 18 tháng 10 năm 2022



Bài 1. FOS đếm hoán vị

Bài 2. Tạo xâu bất

Bài 3. Tàu điện ngầm

## FOS đếm hoán vị

FOS có một dãy  $P = p_1, p_2, \dots, p_n$  là hoán vị của  $n$  số nguyên từ 1 đến  $n$ . FOS viết ra giấy dãy đặc trưng của hoán vị trên gồm  $n$  số nguyên  $c_1, c_2, \dots, c_n$  được định nghĩa như sau:  $c_i = j$  với  $j$  là số nguyên dương nhỏ nhất  $> i$  và  $p_j > p_i$ . Nếu không tồn tại  $j > i$  sao cho  $p_j > p_i$  thì  $c_i = n + 1$ . FOS tự hỏi có bao nhiêu hoán vị có dãy đặc trưng giống dãy đặc trưng của hoán vị  $P$  (tính cả hoán vị  $P$ ).

**Yêu cầu:** Hãy giúp FOS tính con số này.

## Subtask 1: $n \leq 10$

- ▶ Vì  $n \leq 10$  nên ta có thể sinh ra tất cả các hoán vị của các số từ 1 đến  $n$  và kiểm tra dãy đặc trưng của hoán vị đó có bằng dãy đặc trưng của hoán vị ban đầu không.
- ▶ Sử dụng Stack để tính dãy đặc trưng.

## Subtask 2: $n \leq 10^3$

- ▶ Ta có nhận xét sau: nếu có 2 vị trí  $i, j$  sao cho  $i < j$  và  $c_i = c_j$  thì  $p_i > p_j$ , vì  $c_i$  là vị trí bé nhất sao cho  $p_i < p_{c_i}$
- ▶ Ta dựng cây bằng cách nối đỉnh con  $i$  với đỉnh cha  $c_i$
- ▶ Gọi  $dp_u$  là số cách để sắp xếp các số từ 1 đến  $size_u$  trong cây con gốc  $u$  sao cho đỉnh con bé hơn đỉnh cha và nếu 2 đỉnh cùng tầng thì đỉnh có chỉ số nhỏ hơn sẽ nhận giá trị lớn hơn.
- ▶ Nhận xét rằng nếu 2 cây con gốc  $u$  và  $v$  ( $u \leq v$ ) cùng tầng và là con của  $p$  thì ta phải đảm bảo giá trị ở đỉnh  $u$  lớn hơn giá trị ở đỉnh  $v \Rightarrow$  để xếp các số từ 1 đến  $size_u + size_v$  vào 2 cây con gốc  $u$  và  $v$  thì giá trị ở đỉnh  $u$  phải bằng  $size_u + size_v$  và các giá trị còn lại có thể gán tùy ý. Ta có số cách xếp là  $dp_u * dp_v * C_{size_u + size_v - 1}^{size_v}$
- ▶ Khi gộp 2 cây con cùng tầng vào ta có thể coi nó là một cây con và gộp nó với các cây con phía trước.

## Subtask 2: $n \leq 10^3$

```
void dfs(int node, int parent) {  
  
    dp[node] = 1;  
    sizeTree[node] = 0;  
  
    for (auto &u : adj[node]) {  
        if (u != parent) {  
  
            dfs(u, node);  
            sizeTree[node] += sizeTree[u];  
            dp[node] = (dp[node] * dp[u] % MOD)  
                      * C(sizeTree[node] - 1, sizeTree[u] - 1) % MOD;  
        }  
    }  
    sizeTree[node]++;  
}
```

- ▶ Ta có thể tính tổ hợp bằng mảng 2 chiều trong  $O(n^2)$
- ▶ Kết quả của bài toán là  $dp_{n+1}$
- ▶ Độ phức tạp  $O(n^2)$

### Subtask 3: $n \leq 10^5$

- ▶ Việc tính tổ hợp trong subtask 2 mất  $O(n^2)$ , ta có thể cải tiến bằng cách sử dụng nghịch đảo mô-đun
- ▶ Định lý Fermat nhỏ:  $x^0 \equiv x^{mod-1}(\%mod)$
- ▶ Ta có  $\frac{1}{x} \equiv \frac{1}{x} \cdot x^{mod-1} \equiv x^{mod-2}(\%mod)$
- ▶ 
$$C_n^k = \frac{n!}{k! \cdot (n-k)!} = n! \cdot \frac{1}{k!} \cdot \frac{1}{(n-k)!} \equiv n! \cdot (k!)^{mod-2} \cdot ((n-k)!)^{mod-2}(\%mod)$$



Bài 1. FOS đếm hoán vị

Bài 2. Tạo xâu bít

Bài 3. Tàu điện ngầm

# Tạo chuỗi bit — BITSTR

- ▶ Sơn muốn tạo ra một chuỗi  $S$  nhị phân 0/1 gồm  $N$  bit  $S_1, S_2, \dots, S_N$  thoả mãn  $K$  ràng buộc khác nhau, mỗi ràng buộc được xác định thông qua ba số nguyên  $A_i, B_i$  và  $C_i$ . Số chữ số 1 trong chuỗi con  $S_{A_i}, S_{A_i+1}, \dots, S_{B_i}$  phải đúng bằng  $C_i$ .
- ▶ Yêu cầu: giúp Sơn chọn chuỗi có thứ tự từ điển đứng thứ  $P$  tính từ 1 trong tập các chuỗi thoả mãn.

## Phương pháp chung:

1. Lắp theo thứ tự từ điển (0 và 1) cho bit đầu tiên  $i$ .
2. Đếm số lượng  $X$  chuỗi hợp lệ với tiền tố  $i$ . Nếu  $X \leq P$ , ta có  $i$  là bit phù hợp cho vị trí đầu tiên. Nếu không, tiếp tục vòng lặp và trừ dần  $X$  khỏi  $P$ .
3. Quay lại bước 1 với vị trí tiếp theo cho đến vị trí cuối cùng.

## Subtask 1: $A_i = B_i, \forall 1 \leq i \leq K$

- ▶  $A_i = C_i, \forall 1 \leq i \leq K$ .
- ▶ Số lượng xâu hợp lệ là  $2^Y$ , với  $Y$  là số lượng vị trí không có điều kiện ràng buộc.
- ▶ Ví dụ, số lượng xâu hợp lệ không tính yêu cầu tiền tố là  $2^{N-K}$ .

## Subtask 2 ( $B_i - A_i \leq 15, \forall 1 \leq i \leq K$ ): QHD bitmask

- ▶ Gọi  $f(x, last)$ , với  $x \in [1, N]$  và  $last$  là số nguyên 15 bit (15 là giá trị lớn nhất có thể có của  $B_j - A_j$ ), là số cách:
  - ▶ gán các số 0 và 1 cho chuỗi bit từ vị trí thứ  $x$  đến vị trí thứ  $N$  (nói cách khác, giả sử rằng chuỗi bit đã được "lấp đầy" cho đến vị trí  $(x - 1)$  -th),
  - ▶ 15 bit cuối cùng (từ vị trí thứ  $(x - 15)$  đến vị trí thứ  $(x - 1)$ ) được thể hiện ở  $last$ , và
  - ▶ phép gán phải thỏa mãn tất cả các ràng buộc  $i$  với  $B_i \geq x$ .

## Subtask 2 ( $B_i - A_i \leq 15, \forall 1 \leq i \leq K$ ): QHD bitmask

- ▶ Lặp  $i \in \{0, 1\}$  cho vị trí thứ  $x$ . Ta có được 16 ký tự cuối cùng. Vì  $B_j - A_j \leq 15$ , ta có thể kiểm tra tất cả các ràng buộc  $j$  với  $B_j = x$ .
- ▶ Nếu một trong các ràng buộc này bị vi phạm thì  $i$  không phải là bit hợp lệ có thể đặt vào vị trí thứ  $x$ . Nếu không, việc đặt  $i$  ở vị trí thứ  $x$  sẽ đóng góp vào  $f(x + 1, last')$  thành  $f(x, last)$ , trong đó  $last'$  thu được từ mặt nạ  $last$  bằng cách bỏ qua bit đầu tiên của  $last$  (do ta sẽ không cần bit từ vị trí  $(x - 15)$  nữa) và thêm một bit  $i$  vào cuối.
- ▶ Với tiền tố  $s$ , số lượng xâu hợp lệ là  $f(length(s) + 1, last\_s)$ , với  $last\_s$  là 15 bit cuối cùng của  $s$ .
- ▶ Có  $N$  khả năng cho  $x$ ,  $2^{15}$  khả năng cho  $last$  và mỗi phép tính  $f(x, y)$  cần một vòng lặp để kiểm tra các ràng buộc, thời gian chạy của thuật toán này là  $2^{15} \times O(N + K)$ .

Bài 1. FOS đếm hoán vị

Bài 2. Tạo xâu bất

Bài 3. Tàu điện ngầm

## Tàu điện ngầm

Thành phố mà Hải đang sống có hệ thống tàu điện ngầm bao gồm  $N$  trạm và  $M$  tuyến đường ngầm mỗi tuyến nối trực tiếp hai trạm khác nhau. Hai trạm được gọi là kề nhau nếu như có tuyến đường nối trực tiếp hai trạm. Không có 2 tuyến nào nối cùng 1 cặp trạm và giữa hai trạm bất kỳ đều có cách đi giữa chúng thông qua các tuyến đường ngầm. Trạm  $i$  có số lượng khách trung bình hàng ngày là  $p_i$ .

Hải mới được giao nhiệm vụ xây dựng và phát triển các chi nhánh công ty quảng cáo của mình tại các trạm tàu điện ngầm. Sau khi khảo sát, Hải nhận thấy rằng hệ thống tàu điện ngầm là có không quá 15 trạm mà mỗi trạm kề với ít nhất 3 trạm khác.

**Yêu cầu:** Giúp Hải xây dựng trên một tập con các trạm sao cho không có hai trạm nào trong tập con được chọn là kề nhau, và tổng số lượng khách trung bình hàng ngày của các trạm trong tập đó phải là lớn nhất.

Subtask 1 ( $N \leq 20$ ):  $O(2^N)$

Duyệt toàn bộ tập con.



## Subtask 2,3: Đồ thị chỉ gồm mạch thẳng hoặc vòng: $O(N)$

Quy hoạch động đơn giản cho bài toán tập độc lập trọng số lớn nhất trên mạch thẳng và mạch vòng.

## Subtask 4 (100%)

Do có không quá 15 đỉnh bậc  $\geq 3$ , các đỉnh còn lại bậc 1 hoặc 2 tạo nên các mạch thẳng và vòng. Chia bài toán thành các phần giải quyết độc lập như sau:

- ▶ Tách đồ thị thành các phần mạch thẳng và mạch vòng xử lý riêng như trong Subtask 2,3;
- ▶ Duyệt  $\mathcal{O}(2^{15})$  với phần còn lại cho các đỉnh bậc  $\geq 3$ .