

Tổng quan về các bài thi trong đề

TT	Tên bài	File Chương trình	Điểm
1	Sắp xếp xâu	REORDER.***	100
2	Phủ sóng 5G	COVER5G.***	100
3	Đột biến Gen	MUTATION.***	100
4	Thứ tự duyệt DFS	MULTIDFS.***	100

Chú ý: Dữ liệu nhập vào và đưa ra file theo yêu cầu.

Viết chương trình giải các bài toán sau:

Bài 1. Sắp xếp xâu[Reorder]

Cho xâu S chỉ bao gồm các ký tự Latin in thường.

Hãy sắp xếp lại các ký tự của xâu S để tạo ra xâu T thỏa mãn:

- Không có 2 ký tự liên tiếp nào của T giống nhau
- T có thứ tự từ điển nhỏ nhất

Dữ liệu: vào từ file **REORDER.INP** gồm một dòng duy nhất chứa xâu S ($|S| \leq 10^5$)

Kết quả: ghi ra file **REORDER.OUT** xâu T thỏa mãn hoặc in ra -1 nếu không tồn tại.

Ví dụ:

REORDER.INP	REORDER.OUT
badcadb	ababdcd

Subtasks:

- Subtask 1 (30 điểm): $|S| \leq 10$
- Subtask 2 (20 điểm): xâu S chỉ gồm các ký tự a, b, c .
- Subtask 3 (50 điểm): không có ràng buộc gì thêm.

Bài 2. Phủ sóng 5G [RAID]

Một thành phố có một con đường huyết mạch và n điểm dân cư dọc theo con đường đó. Điểm dân cư thứ i ở vị trí x_i . Lãnh đạo thành phố đang cần triển khai phủ sóng 5G để mọi người dân đều có thể kết nối. Qua khảo sát, điểm dân cư thứ i có c_i người. Thành phố dự kiến xây dựng k trạm phát có cùng công suất như nhau. Để đảm bảo an toàn, tất cả các trạm phát đều phải được đặt ở các điểm dân cư. Công suất mỗi trạm phát luôn là một giá trị số nguyên dương. Nếu trạm phát có công suất p ở điểm dân cư thứ i có thể phát tín hiệu cho người ở điểm dân cư thứ j nếu như $|x_i - x_j| \leq p$. Tại mỗi điểm dân cư có thể không xây dựng, hoặc xây dựng 1, hoặc nhiều trạm phát khác nhau.

Yêu cầu: Biết mỗi trạm phát có thể phục vụ cung cấp tín hiệu cho tối đa M người dân cùng lúc. Hãy xác định công suất tối thiểu để trong trường hợp lý tưởng có thể phát tín hiệu cho tất cả dân cư trong toàn thành phố cùng lúc.

Dữ liệu: vào từ file **COVER5G.INP**

- Dòng đầu chứa số 3 nguyên dương N, M, K ($N \leq 10^5$; $M \leq 10^{15}$; $K \leq 10^9$)
- Dòng 2 chứa N số nguyên $x_1, x_2, x_3, \dots, x_N$
- Dòng 3 chứa N số nguyên $c_1, c_2, c_3, \dots, c_N$ ($0 \leq x_i, c_i \leq 10^9$)

Kết quả: ghi ra file **COVER5G.OUT** một số nguyên duy nhất là giá trị p tối thiểu tìm được.

Ví dụ:

COVER5G.INP	COVER5G.OUT
5 10 2 1 2 3 8 9 4 3 3 4 2	1
5 9 2 1 2 3 8 9 4 3 3 4 2	5

Giải thích:

- Trong ví dụ 1, đặt trạm phát ở vị trí 2 và 9. Lúc đó, trạm phát ở vị trí 2 sẽ phục vụ cho tất cả các khách ở các vị trí 1,2,3. Trạm phát ở vị trí 9 phục vụ cư dân ở vị trí 8,9.
- Trong ví dụ 2, đặt trạm phát ở vị trí 2 và 8. Lúc đó trạm phát ở vị trí 2 phục vụ cho cư dân ở vị trí 1,2 và 2 cư dân ở vị trí 3. Trạm phát ở vị trí 8 sẽ phục vụ cho 1 cư dân ở vị trí 3, 4 cư dân ở vị trí 8 và 2 cư dân ở vị trí 9.

Subtasks:

- Subtask 1 (10 điểm): $K = 1$; $N \leq 10$; $c_i \leq 10^3$; $M \geq 10^5$
- Subtask 2 (20 điểm): $x_i = i$; $c_i = 1 \forall i = 1, 2, 3, \dots, n$
- Subtask 3 (35 điểm): $M \geq 10^{14}$
- Subtasks 4 (35 điểm): không có ràng buộc bổ sung.

Bài 3. Đột biến Gen [MUTATION]

Trong quá trình nghiên cứu và phân tích dữ liệu gen, bạn phát hiện ra một loại đột biến đặc biệt xảy ra trên chuỗi DNA của một mẫu sinh vật. Chuỗi DNA ban đầu được biểu diễn bằng một xâu S gồm n kí tự in thường. Các vị trí được đánh số từ 1 tới n .

Trong quá trình tiến hóa và sao chép tế bào, hệ thống ghi nhận có hai loại sự kiện liên tục xảy ra đối với chuỗi DNA này:

- **Sự kiện 1 (Đột biến nhân đôi):** Một đoạn gen từ vị trí l đến r xảy ra lỗi trong quá trình phiên mã, khiến mỗi nucleotide (kí tự) trong đoạn này bị nhân đôi. (Ví dụ: nếu chuỗi hiện tại là *abac* và xảy ra đột biến từ vị trí 2 đến 3, chuỗi mới sẽ bị kéo dài và trở thành *abbaac*).
- **Sự kiện 2 (Truy vấn trích xuất):** Để đánh giá đặc điểm di truyền tại thời điểm hiện tại, bạn cần trích xuất thông tin của chuỗi gen. Bạn muốn xác định xem nucleotide ở vị trí thứ i trong chuỗi gen lúc này đang là kí tự nào.

Với số lượng các sự kiện đột biến xảy ra liên tục, chuỗi gen có thể trở nên rất dài. Nhiệm vụ của bạn là viết một chương trình mô phỏng lại quá trình biến đổi này và trả lời chính xác các truy vấn trích xuất dữ liệu.

Dữ liệu: vào từ file **MUTATION.INP**

- Dòng đầu tiên chứa hai số nguyên n và q ($1 \leq n, q \leq 3 \cdot 10^5$) — độ dài của chuỗi DNA ban đầu s và tổng số sự kiện được ghi nhận.
- Dòng thứ hai chứa chuỗi s gồm n ký tự là các chữ cái tiếng Anh in thường — cấu trúc của chuỗi DNA ban đầu.
- q dòng tiếp theo, mỗi dòng mô tả một sự kiện theo một trong hai định dạng:
 - o 1 $l\ r$ ($1 \leq l \leq r \leq 10^{18}$): Các nucleotide trong đoạn $[l, r]$ của chuỗi hiện tại bị nhân đôi.
 - o 2 i ($1 \leq i \leq 10^{18}$): Yêu cầu xác định kí tự tại vị trí i .

Tất cả giá trị l, r, i đều đảm bảo không vượt quá độ dài xâu S tại thời điểm sự kiện diễn ra.

Kết quả: ghi ra file **MUTATION.OUT** với mỗi sự kiện truy vấn loại 2, in ra kí tự nucleotide tìm được trên một dòng riêng biệt.

Ví dụ:

MUTATION.INP	MUTATION.OUT
4 6	b
abac	a
2 2	b
2 3	a
1 2 3	c
2 3	
2 4	
2 6	

Subtasks:

- Subtask 1 (15 điểm) : Tất cả các sự kiện loại 1 đều có $l = 1$ và $r = |S|$ (với $|S|$ là độ dài xâu S tại thời điểm hiện tại).
- Subtask 2 (30 điểm) : $n, q \leq 2000$.
- Subtask 3 (35 điểm) : $n, q \leq 5 \times 10^4$.
- Subtask 4 (20 điểm) : Không có ràng buộc gì thêm.

Bài 4. Thứ tự duyệt [MULTIDFS]

Trong tính toán trên đồ thị, quá trình duyệt DFS trên một cây xuất phát từ gốc, khi đứng tại một đỉnh u , thuật toán sẽ đi thăm các đỉnh con trực tiếp của u theo một thứ tự hoán vị **bất kỳ**. Dãy thứ tự DFS là dãy lưu lại các đỉnh theo đúng thứ tự mà chúng được ghé thăm lần đầu tiên.

Yêu cầu: Cho m dãy thứ tự duyệt theo chiều sâu (DFS order) của cùng một cây gồm n đỉnh (các đỉnh được đánh số từ 1 đến n), bạn là đếm xem có tất cả bao nhiêu cấu trúc cây phân biệt có khả năng sinh ra tất cả m dãy DFS đã cho. Hai cây được coi là phân biệt nếu tồn tại ít nhất một đỉnh có tập hợp các đỉnh con trực tiếp trong hai cây khác nhau.

Dữ liệu: vào từ file **MULTIDFS.INP**

- Dòng đầu tiên chứa hai số nguyên n và m ($1 \leq n, m \leq 500$) — biểu thị số lượng đỉnh của cây và số lượng dãy DFS đã được ghi nhận.
- m dòng tiếp theo, mỗi dòng chứa một hoán vị gồm n số nguyên phân biệt từ 1 đến n , biểu diễn một dãy thứ tự DFS.

Dữ liệu đảm bảo phần tử đầu tiên của cả m dãy đều giống nhau.

Kết quả: ghi ra file **MULTIDFS.OUT** một số nguyên duy nhất là số lượng cấu trúc cây phân biệt thỏa mãn điều kiện. Đưa ra số dư trong phép chia cho 998244353.

Ví dụ

MULTIDFS.INP	MULTIDFS.OUT
5 2 1 2 3 4 5 1 4 5 2 3	4

Subtasks

- Subtask 1 (10 điểm): $n \leq 9$; $m \leq 10$.
- Subtask 2 (20 điểm): $m = 1$.
- Subtask 3 (30 điểm): $n, m \leq 50$.
- Subtask 4 (40 điểm): Không có ràng buộc gì thêm.

-----HẾT-----