

Mutating DNA

Grace là một nhà sinh học làm việc trong một công ty tin sinh học tại Singapore. Một phần công việc hàng ngày của cô là phân tích chuỗi ADN của các tổ chức sinh học. Một chuỗi ADN được định nghĩa là một xâu gồm các ký tự "A", "T", và "C". Chú ý rằng, trong bài này, các chuỗi ADN **không chứa ký tự "G"**.

Chúng ta định nghĩa một phép đột biến là thao tác hoán đổi hai phần tử trên một chuỗi ADN. Ví dụ một phép đột biến chuyển đổi chuỗi "ACTA" thành "AATC" bằng cách hoán đổi hai ký tự in đậm "A" và "C".

Khoảng cách đột biến giữa hai chuỗi là số lượng phép đột biến ít nhất cần thực hiện để chuyển đổi từ chuỗi này sang chuỗi kia, hoặc -1 nếu không tồn tại cách chuyển đổi từ chuỗi này sang chuỗi kia bằng việc sử dụng các phép đột biến.

Grace đang tiến hành phân tích hai chuỗi ADN a và b , cả hai đều chứa n phần tử được đánh chỉ số từ 0 đến $n - 1$. Nhiệm vụ của bạn là giúp Grace trả lời q câu hỏi dạng: khoảng cách đột biến giữa chuỗi con $a[x..y]$ và chuỗi con $b[x..y]$ là bao nhiêu? Trong trường hợp này, một chuỗi con $s[x..y]$ của một chuỗi ADN s được định nghĩa là một dãy các ký tự liên tiếp của s , có chỉ số từ x đến y bao gồm cả hai đầu mút. Nói cách khác, $s[x..y]$ là dãy $s[x]s[x + 1] \dots s[y]$.

Chi tiết cài đặt

Bạn cần cài đặt các hàm sau:

```
void init(std::string a, std::string b)
```

- a , b : các xâu độ dài n , biểu diễn hai chuỗi ADN cần phân tích.
- Hàm được gọi đúng một lần, trước bất kỳ lời gọi nào tới hàm `get_distance`.

```
int get_distance(int x, int y)
```

- x , y : các chỉ số đầu và cuối của các xâu con cần phân tích.
- Hàm trả về khoảng cách đột biến giữa xâu con $a[x..y]$ và xâu con $b[x..y]$.
- Hàm được gọi đúng q lần.

Ví dụ

Xét lời gọi hàm sau:

```
init("ATACAT", "ACTATA")
```

Giả thiết trình chấm mẫu gọi hàm `get_distance(1, 3)`. Kết quả trả về của lời gọi hàm là khoảng cách đột biến giữa $a[1..3]$ và $b[1..3]$, tương ứng với chuỗi "TAC" và chuỗi "CTA". "TAC" có thể chuyển đổi thành "CTA" bằng cách thực hiện 2 phép đột biến: **TAC** \rightarrow **CAT**, tiếp đó là **CAT** \rightarrow **CTA**, và sự chuyển đổi này không thể thực hiện được nếu dùng ít hơn 2 phép đột biến.

Do đó, lần gọi này hàm trả về giá trị 2.

Giả thiết trình chấm mẫu gọi hàm `get_distance(4, 5)`. Kết quả trả về của lời gọi hàm là khoảng cách đột biến giữa chuỗi "AT" và chuỗi "TA". "AT" có thể chuyển đổi thành "TA" bằng cách thực hiện một phép đột biến, và rõ ràng cần ít nhất một phép đột biến trường hợp này.

Do đó, lần gọi này hàm trả về giá trị 1.

Cuối cùng, giả thiết trình chấm mẫu gọi hàm `get_distance(3, 5)`. Vì **không tồn tại cách** chuyển đổi chuỗi "CAT" thành "ATA" bằng cách thực hiện dãy các phép đột biến bất kì, lần gọi này hàm trả về giá trị -1 .

Các ràng buộc

- $1 \leq n, q \leq 100\,000$
- $0 \leq x \leq y \leq n - 1$
- Mỗi kí tự của a và b là một trong các kí tự "A", "T", và "C".

Các subtask

1. (21 điểm) $y - x \leq 2$
2. (22 điểm) $q \leq 500$, $y - x \leq 1000$, mỗi kí tự của a và b là kí tự "A" hoặc "T".
3. (13 điểm) mỗi kí tự của a và b là kí tự "A" hoặc "T".
4. (28 điểm) $q \leq 500$, $y - x \leq 1000$
5. (16 điểm) Không có ràng buộc nào thêm.

Trình chấm mẫu

Trình chấm mẫu đọc dữ liệu vào theo định dạng sau:

- dòng 1: $n\ q$
- dòng 2: a
- dòng 3: b
- dòng $4 + i$ ($0 \leq i \leq q - 1$): $x\ y$ cho lời gọi hàm `get_distance` lần thứ i .

Trình chấm mẫu in các câu trả lời của bạn theo định dạng sau:

- dòng $1 + i$ ($0 \leq i \leq q - 1$): giá trị trả về của lời gọi hàm `get_distance` lần thứ i .