

Mutasi DNA

Grace adalah seorang ahli biologi yang bekerja di perusahaan bioinformatika di Singapura. Sebagai bagian dari pekerjaannya, dia menganalisa rangkaian DNA dari berbagai macam makhluk. Sebuah rangkaian DNA didefinisikan sebagai sebuah *string* yang terdiri dari karakter "A", "T", dan "C". Perhatikan bahwa dalam kasus ini, DNA **tidak memiliki karakter "G"**.

Sebuah mutasi didefinisikan sebagai sebuah operasi pada rangkaian DNA dimana dua elemen dari rangkaian tersebut ditukar. Sebagai contoh, sebuah mutasi dapat mengubah "ACTA" menjadi "AATC" dengan menukar karakter "A" dan "C" yang ditebalkan.

Jarak mutasi antara dua buah rangkaian adalah jumlah mutasi paling sedikit yang diperlukan untuk mengubah salah satu rangkaian menjadi yang lainnya, atau -1 jika tidak memungkinkan untuk mengubah salah satu rangkaian menjadi yang lainnya dengan menggunakan mutasi.

Grace sedang menganalisa dua rangkaian DNA a dan b , dimana keduanya terdiri dari n elemen dengan indeks dari 0 hingga $n - 1$. Tugas anda adalah membantu Grace menjawab q pertanyaan dalam bentuk: berapakah jarak mutasi antara *substring* $a[x..y]$ dan *substring* $b[x..y]$? Sebuah *substring* $s[x..y]$ dari sebuah rangkaian DNA s didefinisikan sebagai beberapa karakter berurutan s , dengan indeks dari x ke y inklusif. Dengan kata lain, $s[x..y]$ adalah sebuah rangkaian $s[x]s[x + 1] \dots s[y]$.

Detail Implementasi

Anda perlu mengimplementasikan fungsi berikut:

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

- a , b : *string* yang masing-masing memiliki panjang n , mendeskripsikan dua buah rangkaian DNA yang akan dianalisa.
- Fungsi ini akan dipanggil tepat sekali, sebelum pemanggilan apapun terhadap fungsi `get_distance`.

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

- x , y : indeks awal dan akhir dari *substrings* yang akan dianalisa.
- Fungsi ini mengembalikan jarak mutasi antara *substrings* $a[x..y]$ and $b[x..y]$.
- Fungsi ini akan dipanggil tepat sebanyak q kali.

Contoh

Perhatikan pemanggilan fungsi berikut:

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

Jika *grader* memanggil `get_distance(1, 3)`. Pemanggilan ini perlu mengembalikan jarak mutasi antara $a[1..3]$ dan $b[1..3]$, yaitu string "TAC" dan "CTA". "TAC" bisa diubah menjadi "CTA" dengan menggunakan 2 mutasi: **TAC** \rightarrow **CAT**, dilanjutkan dengan **CAT** \rightarrow **CTA**, dan perubahan tidak mungkin terjadi dengan menggunakan kurang dari 2 mutasi.

Maka dari itu, pemanggilan ini akan mengembalikan 2.

Jika *grader* memanggil `get_distance(4, 5)`. Pemanggilan ini perlu mengembalikan jarak mutasi antara "AT" dan "TA". "AT" bisa diubah menjadi "TA" dengan menggunakan sebuah mutasi, dan tentu saja minimal satu mutasi diperlukan.

Maka dari itu, pemanggilan ini akan mengembalikan 1.

Jika *grader* memanggil `get_distance(3, 5)`. Karena **tidak ada** cara untuk "CAT" berubah menjadi "ATA" dengan menggunakan rangkaian mutasi apapun, pemanggilan ini akan mengembalikan -1 .

Batasan

- $1 \leq n, q \leq 100\,000$
- $0 \leq x \leq y \leq n - 1$
- Setiap karakter dari a dan b merupakan salah satu dari "A", "T", dan "C".

Subsoal

1. (21 poin) $y - x \leq 2$
2. (22 poin) $q \leq 500$, $y - x \leq 1000$, setiap karakter pada a dan b terdiri dari "A" atau "T".
3. (13 poin) setiap karakter a dan b merupakan "A" atau "T".
4. (28 poin) $q \leq 500$, $y - x \leq 1000$
5. (16 poin) Tidak ada batasan tambahan.

Contoh *grader*

Contoh *grader* membaca masukan dengan format berikut:

- baris 1: $n\ q$
- baris 2: a
- baris 3: b
- baris $4 + i$ ($0 \leq i \leq q - 1$): $x\ y$ untuk pemanggilan ke- i terhadap fungsi `get_distance`.

Contoh *grader* mencetak jawaban anda dengan format berikut:

- baris $1 + i$ ($0 \leq i \leq q - 1$): nilai yang dikembalikan dari pemanggilan ke- i fungsi `get_distance`.

