

DNR mutacijos

Grace dirba biologe Singapūre esančioje bioinformatikos įmonėje. Darbe ji analizuoja įvairių organizmų DNR sekas. DNR seka apibrėžiama kaip simbolių eilutė, sudaryta iš simbolių „A“, „T“ ir „C“. Atkreipkite dėmesį, kad šioje užduotyje DNR sekose **nėra simbolio „G“**.

DNR sekos mutacija yra operacija, kurios metu sukeičiami du sekos elementai. Pavyzdžiui, viena mutacija gali transformuoti seką „ACTA“ į seką „AATC“, sukeisdama paryškintus simbolius „A“ ir „C“.

Mutavimo atstumas tarp dviejų sekų lygus mažiausiam mutacijų skaičiui, kurioms įvykus viena seka būtų transformuota į antrąją, arba lygus -1 , jei mutuoiant neįmanoma iš vienos sekos gauti kitą.

Grace analizuoja dvi DNR sekas a ir b , kiekvieną kurių sudaro n elementų, sunumeruotų nuo 0 iki $n - 1$. Padėkite Grace atsakyti į q klausimų: kam lygus mutavimo atstumas tarp posekio $a[x..y]$ ir posekio $b[x..y]$? DNR sekos s posekis $s[x..y]$ apibrėžiamas kaip sekos s iš eilės einančių simbolių, kurių indeksai yra nuo x iki y imtinai, seka. Kitaip sakant, $s[x..y]$ yra seka $s[x]s[x + 1] \dots s[y]$.

Realizacija

Parašykite šias procedūras:

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

- a , b : n ilgio simbolių eilutės, apibūdinančios dvi nagrinėjamas DNR sekas.
- Ši procedūra iškviečiama lygiai vieną kartą prieš bet kokį `get_distance` iškvietimą.

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

- x , y : nagrinėjamų posekių pradžios ir pabaigos indeksai.
- Ši procedūra turi grąžinti mutavimo atstumą tarp posekių $a[x..y]$ ir $b[x..y]$.
- Ši procedūra bus iškviesta lygiai q kartų.

Pavyzdys

Panagrinėkime tokį iškvietimą:

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

Tarkime, kad vertinimo programa iškviečia `get_distance(1, 3)`. Šis iškvietimas turi grąžinti mutavimo atstumą tarp $a[1..3]$ ir $b[1..3]$, t.y. tarp sekų „TAC“ ir „CTA“. „TAC“ galima transformuoti į „CTA“ 2 mutacijomis: $TAC \rightarrow CAT$, ir tada $CAT \rightarrow CTA$, ir ši transformacija yra neįmanoma su mažiau nei 2 mutacijomis.

Taigi, šis iškvietimas turi grąžinti 2.

Tarkime, kad vertinimo programa iškviečia `get_distance(4, 5)`. Šis iškvietimas turi grąžinti mutavimo atstumą tarp sekų „AT“ ir „TA“. „AT“ galima transformuoti į „TA“ viena mutacija, ir akivaizdu, kad bent vienos mutacijos tam reikia.

Taigi, šis iškvietimas turi grąžinti 1.

Galiausiai tarkime, kad vertinimo programa iškviečia `get_distance(3, 5)`. Kadangi **neįmanoma** transformuoti sekos „CAT“ į seką „ATA“ jokia mutacijų seka, šis iškvietimas turi grąžinti -1 .

Ribojimai

- $1 \leq n, q \leq 100\,000$
- $0 \leq x \leq y \leq n - 1$
- Galimi a ir b simboliai yra: „A“, „T“ ir „C“.

Dalinės užduotys

1. (21 taškas) $y - x \leq 2$
2. (22 taškai) $q \leq 500$, $y - x \leq 1000$, kiekvienas a ir b simbolis yra arba „A“, arba „T“.
3. (13 taškų) Kiekvienas a ir b simbolis yra arba „A“, arba „T“.
4. (28 taškai) $q \leq 500$, $y - x \leq 1000$
5. (16 taškų) Papildomų ribojimų nėra.

Pavyzdinė vertinimo programa

Pavyzdinė vertinimo programa skaito įvestį tokiu formatu:

- 1-oji eilutė: $n\ q$
- 2-oji eilutė: a
- 3-oji eilutė: b
- $(4 + i)$ -oji ($0 \leq i \leq q - 1$) eilutė: $x\ y\ i$ -ajam `get_distance` iškvietimui.

Pavyzdinė vertinimo programa išveda atsakymą tokiu formatu:

- $(1 + i)$ -oji ($0 \leq i \leq q - 1$) eilutė: i -ojo `get_distance` iškvietimo grąžinta reikšmė.