

beCP 2023

Taak 2.3: Password (password)

Auteur: Damien Galant

Vorbereiding: Damien Galant

Maximale uitvoeringsduur: 1 s Geheugenlimiet: 512 MB

Aangezien u traint voor de Internationale Informatica Olympiade, is uw vriend, een informaticus, zo vriendelijk u toegang te geven tot zijn online collectie C++ algoritmen. Deze is beveiligd met een wachtwoord. Om u uit te dagen stelt uw vriend voor dat u zelf het wachtwoord vindt door een van zijn algoritmische problemen op te lossen. Kunt u zijn uitdaging aan?

Zijn probleem gaat over strings bestaande uit de 26 letters van het alfabet, in kleine letters (bijvoorbeeld `becp`).

Er bestaan twee bewerkingen:

1. Alle voorkomens van de letter l_1 in de string verwijderen.
2. Alle voorkomens van de letter l_1 in de string vervangen door de letter l_2 .

Dus als de bewerking 1 wordt toegepast op de string `aeeeaecae`, met $l_1 = \mathbf{a}$, is de resulterende string `eeeece`.

Als de bewerking 2 wordt toegepast op de string `abbbabcaba` met $l_1 = \mathbf{b}$ en $l_2 = \mathbf{e}$, is de resulterende string `aeeeaecaea`.

Uw vriend legt uit hoe u het wachtwoord van zijn verzameling algoritmen kunt vinden:

- U krijgt een string S . Dit is de initiële string.
- Een opeenvolging van M bewerkingen van type 1 en 2 is gegeven.
- K keer achter elkaar, moet u de M bewerkingen toepassen, in de volgorde waarin ze zijn opgegeven. Als de bewerkingen O_1, \dots, O_M zijn, past u achtereenvolgens toe

$$\underbrace{O_1, O_2, \dots, O_M}_{\text{1ste iteratie}}, \underbrace{O_1, O_2, \dots, O_M}_{\text{2de iteratie}}, \dots, \underbrace{O_1, O_2, \dots, O_M}_{\text{kde iteratie}}.$$

Vertrekkende van S past u O_1 toe en verkrijgt u een string S_1 , dan past u O_2 toe op S_1 en krijgt u een string S_2 , enzovoort.

De string die na deze $K \cdot M$ bewerkingen wordt verkregen (dus $S_{K \cdot M}$) is het wachtwoord dat u zoekt. Aan u om het te vinden!

Input

De eerste regel bevat de drie gehele getallen N , M en K . De tweede regel bevat de initiële string S , die alleen kleine letters bevat. De volgende M regels zijn van het type

- 1 l_1 , die een bewerking van het type 1 beschrijft;
- 2 $l_1 l_2$, die een bewerking van het type 2 beschrijft.

Output

De uitvoer bestaat uit een enkele regel met een string van kleine letters: het wachtwoord van uw vriend. Er wordt gegarandeerd dat het te raden wachtwoord niet leeg is.

Algemene limieten

- S bestaat uit N kleine letters uit `a-z`;
- voor elke bewerking van het type 2 geldt $l_1 \neq l_2$;
- $1 \leq N$;
- $1 \leq M$;
- $N \cdot M \leq 10^6$;
- $1 \leq K \leq 10^{18}$.

Bijkomende beperkingen

Subtaak	Punten	Beperkingen
A	15	$K = M = 1$
B	20	Er zijn enkel bewerkingen van het type 1
C	20	$K \cdot N \cdot M \leq 10^6$
D	20	Er zijn enkel bewerkingen van het type 2
E	25	Geen bijkomende beperkingen

Belangrijke opmerking: sommige getallen kunnen de capaciteit van een 32-bit getal overschrijden, dus gebruik het type `long long`.

Voorbeeld 1

sample1.in	sample1.out
10 1 1 aaaaeacaea 1 a	eeeece

Dit voorbeeld kan voorkomen in de subtaken A, B, C en E.

Voorbeeld 2

sample2.in	sample2.out
10 1 1 abbbabcaba 2 b e	aaaaeacaea

Dit voorbeeld kan voorkomen in de subtaken A, C, D en E.

Voorbeeld 3

sample3.in	sample3.out
10 3 2 uvbwewucvp 2 u v 1 v 2 w u	becp

Dit voorbeeld kan voorkomen in de subtaken C en E. In dit voorbeeld moet een opeenvolging van drie bewerkingen, van het type 2, 1 en 2, tweemaal worden herhaald. De oorspronkelijke string is `uvbwewucvp`. Door de bewerkingen toe te passen, verkrijgen we achtereenvolgens `uvbwewvcvp`, `bwewcp`, `bueucp`, `bvevcv`, `becp` en `becp`. Het wachtwoord is dus `becp`!