



Combo

Stai giocando ad un videogioco d'azione controllato da 4 pulsanti: A, B, X, e Y. In questo gioco guadagni gettoni per ogni *mossa combo*, che viene effettuata premendo i pulsanti in una certa sequenza.

Questo gioco ha una sequenza segreta di pulsanti che può essere rappresentata come una sequenza S di questi 4 caratteri. Non conosci la stringa S , tuttavia conosci la sua lunghezza N e **sai anche che il primo carattere di S non ha altre occorrenze in essa**. Ad esempio, S può essere "ABXYY" o "XYAA", ma non "AAAAA" o "BXYBX".

Puoi premere una sequenza di massimo $4N$ pulsanti per effettuare una mossa combo. Sia p la stringa che rappresenta la sequenza di pulsanti che hai premuto. Il numero di gettoni che ottieni per questa mossa è calcolato come la lunghezza del più lungo prefisso di S che è anche sottostringa di p (*una sottostringa di una stringa t è una sequenza contigua di caratteri dentro t , e un prefisso di t è una sottostringa di t vuota o che contiene il primo carattere di t*).

Per esempio, se S è "ABXYY" e p è "XXYYABYABXAY" otterrai 3 gettoni, poiché "ABX" è il più lungo prefisso di S che è anche una sottostringa di p . Individua la stringa segreta S usando il minor numero possibile di mosse combo!

Dettagli implementativi

Devi implementare la seguente funzione:

```
string guess_sequence(int N)
```

- N : la lunghezza della stringa S .
- Questa funzione viene chiamata esattamente una volta per ogni caso di test e deve restituire la stringa S .

Il tuo programma può chiamare la seguente funzione:

```
int press(string p)
```

- p : la sequenza di pulsanti premuti, che deve essere una stringa di lunghezza compresa tra 0 e $4N$ (estremi inclusi). Ogni carattere in p deve essere A, B, X o Y.
- Non puoi chiamare questa funzione più di 8000 volte per ogni test case.

- Questa funzione restituisce il numero di gettoni che ottieni quando premi la sequenza di pulsanti rappresentata da p .

Se qualcuna delle condizioni sopracitate non è soddisfatta, il tuo programma verrà giudicato come **Wrong Answer**. Altrimenti, il tuo programma verrà giudicato come **Accepted** e il tuo score sarà calcolato a seconda del numero di chiamate a `press` effettuato (vedi sezione Subtask).

Esempio

Sia S uguale a "ABXY", per cui il grader chiama `guess_sequence(5)`. Un esempio di comunicazione è mostrato qui sotto.

Chiamata	Valore di ritorno
<code>press("XXYYABYABXAY")</code>	3
<code>press("ABXY")</code>	5
<code>press("ABXYABXY")</code>	5
<code>press("")</code>	0
<code>press("X")</code>	0
<code>press("BXY")</code>	0
<code>press("YYXBA")</code>	1
<code>press("AY")</code>	1

Per la prima chiamata a `press`, "ABX" appare in "XXYYABYABXAY" come sottostringa ma "ABXY" no; quindi viene restituito 3.

Per la terza chiamata a `press`, "ABXY" stesso appare in "ABXYABXY" come sottostringa; quindi viene restituito 5.

Per la sesta chiamata a `press`, nessun prefisso di "ABXY" tranne la stringa vuota appare in "BXY" come sottostringa; quindi viene restituito 0.

Infine, `guess_sequence(5)` deve restituire "ABXY".

Il file `sample-01-in.txt` nell'archivio compresso in allegato corrisponde a questo esempio.

Assunzioni

- $1 \leq N \leq 2000$
- Ogni carattere della stringa S è A, B, X o Y
- Il primo carattere di S non riappare più in S

In questo problema il grader non è adattivo. Questo significa che S è fissata all'inizio dell'esecuzione del grader e non dipende dalle domande effettuate dalla tua soluzione.

Subtask

1. (5 punti) $N = 3$.
2. (95 punti) Nessuna assunzione aggiuntiva. Per questo subtask, il tuo punteggio per ogni caso di test è calcolato nel seguente modo. Sia q il numero di chiamate a `press`.
 - Se $q \leq N + 2$, il tuo punteggio è 95.
 - Se $N + 2 < q \leq N + 10$, il tuo punteggio è $95 - 3(q - N - 2)$.
 - Se $N + 10 < q \leq 2N + 1$, il tuo punteggio è 25.
 - Se $\max\{N + 10, 2N + 1\} < q \leq 4N$, il tuo punteggio è 5.
 - Altrimenti, il tuo punteggio è 0.

Ricorda che il punteggio in ogni subtask è pari al minimo dei punteggi per i casi di test in quel subtask.

Grader di esempio

Il grader di esempio legge l'input nel seguente formato:

- riga 1: S

Se il tuo programma verrà giudicato come **Accepted**, il grader di esempio stamperà `Accepted: q`, dove q è il numero di chiamate alla funzione `press`.

Se il tuo programma verrà giudicato come **Wrong Answer**, il grader di esempio stamperà `Wrong Answer: MSG`, dove `MSG` può essere:

- `invalid press`: se uno dei valori di p dati a `press` non è valido. Nello specifico, la lunghezza di p non è tra 0 e $4N$ (estremi inclusi), o qualche carattere di p non è `A`, `B`, `X` o `Y`.
- `too many moves`: se la funzione `press` è stata chiamata più di 8 000 volte.
- `wrong guess`: se il valore restituito da `guess_sequence` non è S .