



Combo

Je speelt een videospelletje vol met actie. De spelcontroller heeft 4 knoppen, A, B, X, en Y. In dit spel kun je munten verdienen met zogenaamde combo moves. Je kunt een combo move maken door knoppen op een bepaalde volgorde in te drukken.

Dit spel heeft een geheime serie knoppen, die kan worden weergegeven als een string S van deze 4 tekens. Je kent deze string S niet, maar je weet de lengte ervan, N .

Je weet ook dat het eerste teken van S nooit ook op een andere positie van de geheime string voorkomt. Zo kan S bijvoorbeeld "ABXYY" zijn of "XYAA", maar niet "AAAAA" of "BXYBX".

Een combo move bestaat uit het na elkaar indrukken van maximaal $4N$ knoppen. Als p de string is met de knoppen die je hebt ingedrukt, dan wordt het aantal munten dat je verdient berekend door de lengte te nemen van de langste prefix van S die ook een substring is van p . Een substring van een string t is een aaneengesloten (mogelijk lege) serie tekens binnen t . Een prefix van t is een substring van t die leeg is of het eerste teken van t bevat.

Als S bijvoorbeeld "ABXYY" is en p is "XXYYABYABXAY", dan krijg je 3 munten omdat "ABX" de langste prefix van S is die ook een substring is van p .

Het is aan jou om de geheime string S te bepalen waarbij je weinig combo moves gebruikt.

Implementatie details

Implementeer de volgende functie:

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

- N : de lengte van string S .
- Deze functie wordt voor ieder testgeval precies eenmaal aangeroepen.
- De functie retourneert de string S .

Je programma kan de volgende functie aanroepen:

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

- p : een serie knoppen die je na elkaar indrukt.
- p moet een string zijn waarvan de lengte minstens 0 is en en hoogstens N . Alle tekens van p zijn een A, B, X, of Y.
- Je mag deze functie voor ieder testgeval niet meer dan 8 000 keer aanroepen.
- De functie retourneert het aantal munten dat je krijgt als je de serie knoppen indrukt zoals staat aangegeven in p .

Als je je niet aan deze voorwaarden houdt, wordt je programma beoordeeld met **Wrong Answer**. In alle andere gevallen wordt je programma beoordeeld als **Accepted** en wordt je score berekend op basis van het aantal aanroepen van `press` (zie onder Subtaken).

Voorbeeld

Stel dat S "ABXYY" is. De grader roept `guess_sequence(5)` aan. Een voorbeeld van de communicatie zie je hieronder.

Aanroep	Return
<code>press("XXYYABYABXAY")</code>	3
<code>press("ABXYY")</code>	5
<code>press("ABXYYABXYY")</code>	5
<code>press("")</code>	0
<code>press("X")</code>	0
<code>press("BXYY")</code>	0
<code>press("YYXBA")</code>	1
<code>press("AY")</code>	1

Bij de eerste aanroep van `press` komt "ABX" in "XXYYABYABXAY" voor als substring maar "ABXY" niet; daarom wordt 3 geretourneerd.

Bij de derde aanroep van `press` komt "ABXYY" zelf als substring voor in "ABXYYABXYY", dus wordt 5 geretourneerd.

Bij de zesde aanroep van `press` komt alleen de lege string als prefix van "ABXYY" voor, en omdat de lege string ook een substring is van "BXYY" wordt 0 geretourneerd.

Tenslotte moet `guess_sequence(5)` "ABXYY" retourneren.

De file `sample-01-in.txt` in het zipbestand bij de opgave hoort bij dit voorbeeld.

Randvoorwaarden

- $1 \leq N \leq 2000$

- Elk teken van de string S is een A,B, X, or Y.
- Het eerste teken van S komt verder niet in S voor.

Voor deze opgave is de grader NIET adaptief. Dat betekent dat S een vaste waarde heeft als de grader wordt gestart en niet afhangt van de queries die jouw oplossing formuleert.

Subtaken

1. (5 punten) $N = 3$
2. (95 punten) Geen aanvullende randvoorwaarden. Voor deze subtask wordt je score voor ieder testgeval als volgt berekend. Als q het aantal aanroepen is van `press`.
 - Als $q \leq N + 2$, is jouw score 95.
 - Als $N + 2 < q \leq N + 10$, is jouw score $95 - 3(q - N - 2)$.
 - Als $N + 10 < q \leq 2N + 1$, is jouw score 25.
 - Als $\max\{N + 10, 2N + 1\} < q \leq 4N$, is jouw score 5.
 - Anders is jouw score 0.

Merk op dat je score voor iedere subtask het minimum is van de scores voor de testgevallen in deze subtask.

Voorbeeldgrader

De voorbeeldgrader leest de invoer in het volgende format:

- regel 1: S

Als je programma wordt beoordeeld als **Accepted**, voert de grader `Accepted: q` uit, waarbij q het aantal aanroepen is van de functie `press`.

Als je programma wordt beoordeeld als **Wrong Answer**, voert de grader uit `Wrong Answer: MSG`. `MSG` kan de volgende waarden hebben:

- `invalid press`: De waarde van `p` die wordt meegegeven aan `press` is ongeldig. De lengte van `p` is namelijk niet minstens 0 en hoogstens $4N$, of een teken van `p` is ongelijk aan A, B, X, of Y.
- `too many moves`: De functie `press` is meer dan 8 000 keer aangeroepen.
- `wrong guess`: De geretourneerde waarde van `guess_sequence` is ongelijk aan S .