

Słowa

Jaś znalazł ostatnio na ulicy dwa długie słowa. Oba te słowa miały długość n liter i były za długie jak na gust naszego bohatera. Jaś postanowił je skrócić i trochę się przy tym zabawić, wymyślił więc dość nietypowy sposób skrócenia tych słów. Chłopiec wypisał sobie pary liczb a i b ($a \leq b$) i teraz po kolei sprawdza, czy w pierwszym słowie pod słowo zaczynające się na indeksie a i kończące na indeksie b jest identyczne jak pod słowo zaczynające się i kończące na tych samych indeksach w słowie drugim. Jeśli pod słowa są identyczne, to Jaś w obu słowach zmienia najbliższe litery znajdujące się na lewo od pozycji a (o ile takie litery istnieją) na znak $\#$ i przechodzi do kolejnej pary liczb. Znak $\#$ oczywiście nie jest uznawany za literę. Jeśli pod słowa są różne to Jaś od razu przechodzi do następnej pary liczb.

Po przeanalizowaniu wszystkich par liczb, Jaś wypisał sobie oba słowa bez znaków $\#$. Teraz zastanawia się, czy przypadkiem gdzieś się nie pomylił? Pomóż Jasiowi i napisz program, który dla danych słów, oraz par liczb wyznaczy wynikowe słowa.

Wejście

W pierwszych dwóch liniach wejścia znajdują się słowa, które znalazł Jaś. Każde z nich umieszczone jest w osobnej linii i składa się z $n \in [1;10^6]$ małych liter alfabetu łacińskiego. Słowa indeksowane są od 0. W trzeciej linii wejścia znajduje się liczba $m \in [1;10^6]$ oznaczająca ilość par liczb do przeanalizowania. W każdej z kolejnych m linii znajdują się dwie liczby całkowite $a \in [0;n)$ i $b \in [a;n)$ opisane w treści zadania.

Wyjście

Na wyjściu należy wypisać wynikowe słowa, w takiej samej kolejności w jakiej zostały wczytane albo -1 jeżeli słowa te są puste.

Przykład

Wejście:

```
abddqwez  
wbdceweq  
5  
0 1  
1 2  
5 6  
5 5  
1 6
```

Wyjście:

```
bdwez  
bdweq
```

Wyjaśnienie do przykładu:

Oto jak wyglądają słowa po kolejnych przeanalizowanych parach liczb:

1. abddqwez
wbdceweq
2. #bddqwez
#bdceweq
3. #bdd#wez
#bdc#weq
4. #bd##wez
#bd##weq
5. #bd##wez
#bd##weq