

# Virkņu vienādojums

Input file: EQUATION.IN  
Output file: EQUATION.OUT

exe-file: EQUATION.EXE  
Time limit: 0.25 sec.

Nodefinēsim jebkurām divām netukšām virknēm, kas sastāv no maziem latīņu burtiem, divas operācijas:

1. *Saskaitīšana*. Tiek apzīmēta ar zīmi "+". Rezultāts ir virkne, ko iegūst, pierakstot otro virkni pirmās virknes galā. Piemēram, `run+site=runsited`.
2. *Reizināšana*. Tiek apzīmēta ar zīmi "\*". Rezultāts ir virkne, ko iegūst, iespraužot otro virkni aiz katra pirmās virknes simbola. Piemēram, `hd*uba=hubaduba`.

Virknju vienādojums ir izteiksme, kas sastāv no divām daļām, kas atdalītas ar "=" zīmi. Labā vienādojuma puse ir netukša virkne, kas sastāv no maziem latīņu burtiem. Kreisā vienādojuma puse sastāv no netukšām virknēm (kas sastāv no maziem latīņu burtiem), viena vai vairākiem nezināmā simboliem X un zīmēm "+" un "\*". Tiek garantēts, ka kreisajā pusē nezināmais X parādās vismaz vienu reizi. Operācijas tiek izpildītas no kreisās uz labo pusi – abas operācijas ir ar vienādu prioritāti.

Nepieciešams atrisināt doto virkņu vienādojumu: atrast tādu netukšu virkni, ka pēc tās ievietošanas nezināmā X vietā vienādojums kļūst par patiesu vienādību. Tiek garantēts, ka dotajam vienādojumam eksistē vismaz viens atrisinājums.

Piemēram, vienādojuma `abc+X*X=astbstcstsssttst` atrisinājums ir virkne `st`.

**Ievaddati.** Vienīgajā ievaddatu faila rindā ir dots virkņu vienādojums. Labā vienādojuma puse satur ne vairāk par 100000 simboliem.

**Izvaddati.** Vienīgajā izvaddatu faila rindā jāizvada dotā virkņu vienādojuma atrisinājums. Ja vienādojumam ir vairāk par vienu atrisinājumu, jāizvada jebkurš viens no tiem.

Piemēri	EQUATION.IN X*ab=cabcabdab	EQUATION.OUT ccd
	EQUATION.IN X=abc	EQUATION.OUT abc
	EQUATION.IN abc+X*X=astbstcstsssttst	EQUATION.OUT st

---

# Строковое уравнение

Input file: EQUATION.IN  
Output file: EQUATION.OUT

exe-file: EQUATION.EXE  
Time limit: 0.25 sec.

Определим для любых двух непустых строк, состоящих из маленьких латинских букв, две операции:

1. *Сложение*. Обозначается знаком "+". В результате получается строка, получаемая приписыванием второй строки к концу первой строки. Например, `run+site=runsite`.
2. *Умножение*. Обозначается знаком "\*". В результате получается строка, получаемая путем вставки второй строки вслед за каждым символом первой строки. Например, `hd*uba=hubaduba`.

Строковое уравнение представляет из себя выражение, состоящее из двух частей, разделенных знаком "=". Правая часть уравнения представляет из себя непустую последовательность маленьких латинских букв. Левая часть уравнения состоит из непустых строк (состоящих из маленьких латинских букв), одного или нескольких символов неизвестного X и знаков "+" и "\*". Гарантируется, что в левой части неизвестное X встречается по крайней мере один раз. Операции выполняются слева направо – обе операции имеют одинаковый приоритет.

Требуется решить данное строковое уравнение: найти такую непустую строку, что при подстановке ее в уравнение вместо неизвестного X, уравнение превращается в верное равенство. Гарантируется, что заданное уравнение имеет хотя бы одно решение.

Например, решением уравнения `abc+X*X=astbstcstsssttst` является строка `st`.

**Входные данные.** В единственной строке входного файла находится строковое уравнение. Правая часть уравнения содержит не более 100000 символов.

**Выходные данные.** Выходной файл должен в единственной строке содержать решение данного строкового уравнения. Если уравнение имеет более одного решения, то следует выводить любое одно из них.

Примеры	EQUATION.IN	EQUATION.OUT
	X*ab=cabcabdab	ccd
	EQUATION.IN	EQUATION.OUT
	X=abc	abc
	EQUATION.IN	EQUATION.OUT
	abc+X*X=astbstcstsssttst	st