# Problem 3. Phoenix Grid

The Phoenix Grid is an ancient artifact created by the Linguistics miracle – Mozilla, The “Fire Bird”. It is used to translate Phoenix language. You are the newest scientist, researching the Grid and as the research team was almost out of hope, you came up with the genius idea to use Regular Expressions! You saved the day! You are a Hero!

You will begin **receiving encoded messages**. You must **CHECK** each **one** of **them** and if it’s a **VALID**.

A **valid encoded message** consists of **one** **phrase** or **more phrases**, separated by **DOTS** (‘.’).

* A **phrase** consists of exactly **3 characters**.
* A **phrase CANNOT** contain **whitespace** characters or the ‘\_’ (underscore) character.

**Valid** messages: “asd.dsa”, “123.312”, “[3@a.231](mailto:3@a.231)”, “111”, “@sd”, “132.31$.ddd” . . .

**Invalid** messages: “123asdasd.dsa”, “\_@a. sd”, “a.s.d” . . .

When you have found a valid message, you must **check** if it a **PALINDROME** – if it reads the same backward as forward.

**Palindrome** messages: “asd.dsa”, “123.321”, “cat.php.tac” . . .

If the **message** is **VALID** and is a **PALINDROME** print “**YES**”. In any other case, print “**NO**”.

The input ends when you receive the command “ReadMe”.

### Input

* As input you will receive several input lines containing encoded messages.

### Output

* As output you must print **for each** **message** “**YES**” or “**NO**” if its **valid** or **not**.

### Constrains

* The input lines may contain **any ASCII character**.
* There will be no more than **1000 input lines**.
* Allowed working time / memory: **100ms / 16MB**.

### Examples

|  |  |
| --- | --- |
| **Input** | **Output** |
| asd  asd.asd  asd.dsa  123.323.321  \_ds.\_sad.sds  jss.csh.php.hsc.ssj  ReadMe | NO  NO  YES  YES  NO  YES |
| asa  igi.igi  \_\_\_.\_\_\_  .  sds.dsd.sds.dsd.sds.dsd.sds  xha.ahx  ReadMe | YES  YES  NO  NO  YES  YES |