# Problem 3 – Post Office

*The post office is running well, after the breakthrough of the technologies, but people demand security. That's why Moist von Lipwig, started encrypting the messages. After all he forgot what was the way to decipher them, and here you go, you have to help him or…. He will be terribly hurt by a pair of high heels.*

**You read a single line of ASCII symbols**, and the message is somewhere inside it, you must find it.

The input consists of three parts separated with **"|"** like this:

**"{firstPart}|{secondPart}|{thirdPart}"**

Each word **starts with capital letter** and **has a fixed length**, you can find those in each different part of the input.

The **first part** carries the capital letters for each word inside the message. You need to find those capital letters **1 or more** **from A to Z**. The capital letters should be surrounded from the both sides with any of the following symbols – **"#, $, %, \*, &".** And those symbols **should match on the both sides.** This means that **$AOTP$ - is a valid** pattern for the capital letters. **$AKTP% - is invalid since the symbols do not match.**

The **second part** of the data contains the **starting letter ASCII code and words** **length** /**between 1 – 20 charactes/**, in the following format: **"{asciiCode}:{length}"**.For example,"**67:05**" – means that '67' - **ascii code** **equal to the capital letter** **"C",** represents a word starting with "C" with following 5 characters: like **"Carrot".** The **ascii code** should be a **capital letter equal to a letter from the first part**. Word's length **should be exactly 2 digits**. Length **less than** **10 will always have a padding zero**, **you don't need to check that.**

The **third part of the message** are **words separated by spaces.** Those **words have to start with Capital letter [A…Z] equal to the ascii code and have exactly the length for each capital letter you have found in the second part. Those words can contain any ASCII symbol without spaces.**

When you find **valid word**, you have to **print it on a new line**.

## Input / Constraints

* On the first line – the text in form of three different parts separated by **"|". There can be any ASCII character inside the input, except '|'.**
* Input will always be valid - you don’t need to check it
* The input will always have three different parts, that will always be separated by "|".

## Output

* Print all extracted words, each on a new line.
* Allowed working **time** / **memory**: **100ms** / **16MB**

## Examples

|  |  |  |
| --- | --- | --- |
| **Input** | **Output** | **Comment** |
| sdsGGasAOTPWEEEdas$AOTP$|a65:1.2s65:03d79:01ds84:02! -80:07++ABs90:1.1|adsaArmyd Gara So La Arm Armyw21 Argo O daOfa Or Ti Sar saTheww The Parahaos | Argo  Or  The  Parahaos | The capital letters are "AOTP"  Then we look for the addition length of the words for each capital letter. For A(65) -> it's 4. For O(79) -> it's 2 For T(84) -> it's 3 For P(80) -> it's 8.  Then we search in the last part for the words.First, start with letter 'A' and we find "Argo". With letter 'O' we find "Or". With letter 'T' we find "The" and with letter 'P' we find "Parahaos". |
| Urgent"Message.TO$#POAML#|readData79:05:79:0!2reme80:03--23:11{79:05}tak{65:11ar}!77:!23--)77:05ACCSS76:05ad|Remedy Por Ostream :Istream Post sOffices Office Of Ankh-Morpork MR.LIPWIG Mister Lipwig | Post  Office  Ankh-Morpork  Mister  Lipwig | The first capital letters are "POAML"  Then we look for the addition length of the words for each capital letter.  P(80) -> it's 4.  O(79) -> it's 6  A(65) -> it's 12  M(77) -> it's 6  L(76) -> it's 6.  Then we search the last part for the words. First, start with the letter 'P' and we find "Post". With letter 'O' we find "Office". With letter 'A' we find "Ankh-Morpork". With letter 'M' we find "Mister" and with letter 'L' we find "Lipwig". |

*“Welcome to fear, said Moist to himself. It's hope, turned inside out. You know it can't go wrong, you're sure it can't go wrong...But it might.”*