# Problem 2 – Phone Numbers

You are given a string, holding ASCII characters. Find all **name -> phone number** pairs, format them and print them in **ordered list** as list items.

The **name** should be at least **one letter long, can contain only letters** and should **always start with uppercase letter**.

The **phone** **number** should be at least **two digits long**, might **start with +** and might **contain** the following symbols in it: **“(“ ”)” ”/” ”.” ”-“ ” “** (left and right bracket, slash, dot, dash and whitespace).

Between a name and the phone number there might be **any number of symbols, other than letters and “+”**.

Between the name -> phone number pairs there might be **any number of ASCII symbols**.

The output format should be **<b>[name]:</b> [phone number]** (there is **one** **space between** the name and the phone number).

### Input

The input will be read from an **HTTP GET** **request** holding a parameter named **numbersString**.

### Output

The output should hold the **resulted ordered list**, or a single **paragraph,** holding“**No matches!**”.

### Constraints

* The **numbers string** will hold only **ASCII** characters (no Unicode).
* Allowed working time: 0.2 seconds. Allowed memory: 16 MB.

### Examples

|  |  |  |
| --- | --- | --- |
| **Input** | | |
| numbersString | **Angel**$(\*^#**029661234**!@#**Pesho** ,.' **+3592/9653241**;'":{},. **Ivan** **0888 123 456** **John**-=\_**555.123.4567** Stoian!@#$#@ Gosho )=\_\* **Steven** #$(\*&**+1-(800)-555-2468** | |
| **Output** | | **Output in the Browser** |
| <ol><li><b>Angel:</b> 029661234</li> <li><b>Pesho:</b> +35929653241</li> <li><b>Ivan:</b> 0888123456</li> <li><b>John:</b> 5551234567</li> <li><b>Steven:</b> +18005552468</li></ol> | |  |

|  |  |  |
| --- | --- | --- |
| **Input** | | |
| numbersString | There are no phone numbers here!!! | |
| **Output** | | **Output in the Browser** |
| <p>No matches!</p> | |  |