# Problem 3 – Basic Mark-up Language

HTML is old and clumsy so a group of highly motivated Softuni students decided to create a new language for the web. It’s basically the same in terms of clumsiness, but supports several revolutionary tags:

* **Inverse –** transforms its content’s lowercase letters to uppercase and vice-versa.
  + <inverse content="**HelloWorlD**"/> outputs **hELLOwORLd**
* **Reverse –** reverses its content
  + **<**reverse content="**helloworld**"/> outputs **dlrowolleh**
* **Repeat –** repeats its content a specified amount of times
  + **<**repeat value="**2**" content="**helloworld**"/> outputs **helloworldhelloworld**
* **Stop –** <stop/> - marks the end of the **BML** file.

Your task is to write a program that correctly parses all currently supported BML tags and outputs the result to the console.

You should **not** output empty lines. For the content tag to be considered non-empty, it must contain **at least one character**.

### Input

* Until the **stop** tag is reached you receive **one** basic mark-up language tag per line
* There are **no invalid** tags or attributes. There may be whitespace **anywhere** in the input

### Output

* Print the correctly formatted output to the console according to the above described rules.
* Each line must have a number, starting from 1, in format “<number>. <output>”
* The <repeat/> tag outputs each string on a new line

### Constraints

* There are no more than 15 lines of input
* The **content** length is in range [0 … 50]. It will not contain double quotes ‘ " ‘
* The **value** attribute is in range [0 … 10]
* Allowed time/memory: 100ms/16MB

|  |  |
| --- | --- |
| **Input** | **Output** |
| <inverse content="HelloWorlD"/>  **<**reverse content="helloworld"**/>**  **<**reverse content="helloworld"**/>**  **<**repeat value="2" content="helloworld"/>  <stop/> | 1. hELLOwORLd  2. dlrowolleh  3. dlrowolleh  4. helloworld  5. helloworld |

|  |  |
| --- | --- |
| **Input** | **Output** |
| **<**repeat value="2" content="12345"/>  **<**repeat value="5" content="12346"/>  <stop/> | 1. 12345  2. 12345  3. 12346  4. 12346  5. 12346  6. 12346  7. 12346 |