Elastic Words

In this challenge, you must think about words as elastics. What happens when do you tend an elastic applying a constant traction force at both ends?

Every part (or letter, in this case) of the elastic will expand, with the minimum expansion at the ends, and the maximum expansion in the center.

If the word has an odd length, the effective central character of the word will be the pivot that splits the word into two halves.

```
"ABC" -> Left = "A" | Center = "B" | Right = "C"
```

If the word has an even length, you will consider two parts of equal length, with the last character of the left half and the first character of the right half being the center.

```
"ABCD" -> Left = "AB" | Right = "CD"
```

You will represent the expansion of a letter repeating it as many times as its numeric position (so counting the indexes from/to 1, and not from 0 as usual) in its half, with a crescent order in the left half and a decrescent order in the right half.

```
Word = "ANNA"

Left = "AN"
Right = "NA"

Left = "A" * 1 + "N" * 2 = "ANN"
Right = "N" * 2 + "A" * 1 = "NNA"

Word = Left + Right = "ANNNNA"
```

If the word has an odd length, the pivot (the central character) will be the peak (as to say, the highest value) that delimits the two halves of the word.

```
Word = "KAYAK"

Left = "K" * 1 + "A" * 2 = "KAA"
Pivot = "Y" * 3 = "YYY"
Right = "A" * 2 + "K" * 1 = "AAK"

Word = Left + Pivot + Right = "KAAYYYAAK"
```

Given a `word`, implement a function that returns the elasticized version of the word as a string.

Examples

```
elasticize("ANNA") ➙ "ANNNNA"

elasticize("KAYAK") ➙ "KAAYYYAAK"

elasticize("X") ➙ "X"
```

Notes

- For words with less than three characters, the function must return the same word (no traction appliable).
- Remember, into the left part characters are counted from 1 to the end, and, in reverse order until 1 is reached, into the right.