Write **ONE** solution in your chosen language (choose from the notes below) that does the following

1. Accepts an input string called **N**
    a. Must still run with an empty string
    b. Must still run with Numeric Input
    c. Must still run with Non-Numeric input (including letters and symbols)
    d. Must be able to deal with mixed input (Numeric and non numeric)
    e. Input IS **CASE SENSITIVE (a does not equal A)**
2. Determine based on the input string
    a. Determine how many repeating unique substrings exist within the original string. Key word is repeating.
3. Output the answer, either to command window or in a popup.
    a. The answer must contain the count of unique substrings that have repeated themselves
    b. The answer must display all of the unique substrings found for any given input

EXAMPLES		
Input	"123123"	"abcabc"
Output	6 substrings - 1,12,123,2,23,3	6 substrings - a,ab,abc,b,bc,c

Note: This program must be written in either Java, Javascript or Python – any issues with this feel free to write back to us.

USE ONE OF THE TEMPLATES ON THE FOLLOWING PAGES WHEN MAKING YOUR CODE!

JAVASCRIPT TEMPLATE

```javascript
var N = "123123";

// *-------- MODIFY THE BELOW CODE TO SOLVE THE PROBLEM ----------*
// *-------- DO NOT CHANGE THE METHOD NAME - YOU MAY CHANGE PARAMETERS ---------*

function NABTest() {

}



NABTest();


// *----- YOUR CODE GOES ABOVE HERE ------*
```

PYTHON TEMPLATE

```python
N = "123123";

# *-------- MODIFY THE BELOW CODE TO SOLVE THE PROBLEM ----------*
# *-------- DO NOT CHANGE THE METHOD NAME - YOU MAY CHANGE PARAMETERS ---------*

def NABTest():



NABTest()


# *----- YOUR CODE GOES ABOVE HERE ------*
```

JAVA TEMPLATE

```java
// *-------- MODIFY THE BELOW CODE TO SOLVE THE PROBLEM ----------*
// *-------- DO NOT CHANGE THE METHOD NAME - YOU MAY CHANGE PARAMETERS ---------*

class java_YOURNAME {

  public static void main(String[] args) {
    NABTest();
  }

  String N = "123123";

  public static void NABTest() {

  }
// *----- YOUR CODE GOES ABOVE HERE ------*
}
```