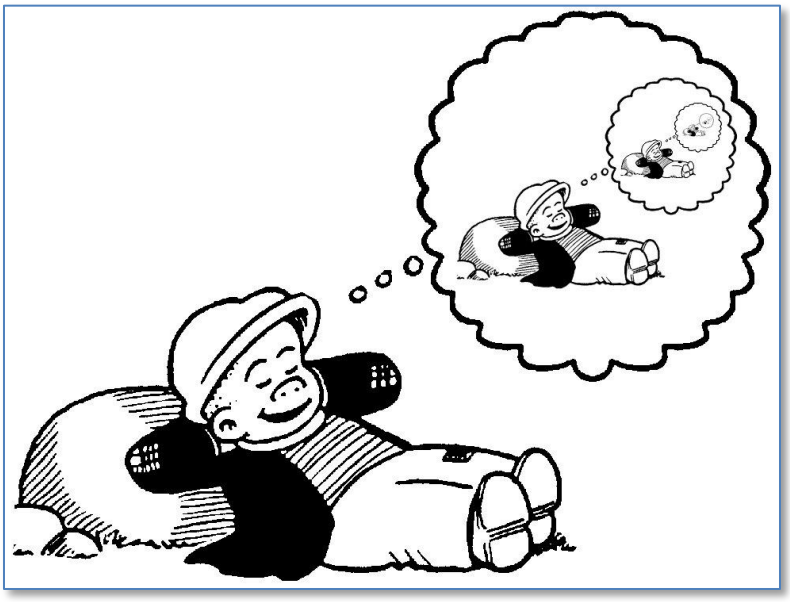
**Assignment – Recursive Descent Parser**
**50 points**
**See Canvas for due date**

---

Implement a class called SubstringGenerator that uses recursion to generate all substrings of a given String. For example, the substrings of the string "Sluggo" are the 22 strings:



```
Sluggo
Slugg
Slug
Slu
Sl
S
luggo
lugg
lug
lu
l
uggo
ugg
ug
u
ggo
gg
g
go
g
o
""
```

You are not allowed to use any loops to build the substrings -- at least in your final product. You may, however, find it productive to solve the problem using loops first, and then translate to recursive code.

---

Implement a class called SubstringTester that is used to interact with the user and allow for testing of the SubstringGenerator class. This class should

- Get a string from the user
- Generate the substrings of the string

- Display the substrings
- Repeat the above three things until the user chooses to quit

**Javadoc**

Additionally, this assignment requires that you 'Javadoc' your source code. That is, your source code must conform to the Javadoc documentation standards. See 'Course Documents' for Javadoc examples.

**To Turn In**

Turn in all source code in a single zip file. Name your zip file as usual. Submit to Blackboard.

**Note**

You should not have a great deal of code for this assignment -- think about it -- its recursion :-)