CS 122 – Lab 8 – Recursively Composing Text

In today’s lab we will write a program that generates some random text.

Please download the file Story1.java from the lab08 folder on the class Web site http://cs.furman.edu/~chealy/cs122/ and copy it to a new folder on your account or hard disk. As you work through the lab, you may want to review the earlier Lobster program that we did in class for inspiration.

Today, the text we will automatically generate is based on a nursery rhyme called “Old Mother Hubbard,” which starts like this:

Old Mother Hubbard went to the cupboard to fetch her poor dog a bone.
But when she came there the cupboard was bare, and so the poor dog had none.

We are going to randomly add lines to continue this story. Old Mother Hubbard is going to go out shopping. We need to have the computer decide where she will shop, and also decide what she will buy there.

Shopping venues:

- baker’s
- tavern
- grocer’s
- tailor’s
- cobbler’s
- chemist’s

Things that she can buy (anywhere):

- shoe
- hat
- cigar
- beer
- cake
- suit
- tire

As a result, here is a grammar that specifies the format of our story.

S → “Old Mother Hubbard went to the cupboard to fetch her poor dog a bone. But when she came there the cupboard was bare, and so the poor dog had none.” A

A → “” | BA

B → “She went to the” P “to buy him a” T

P → “baker’s” | “tavern” | “grocer’s” | “tailor’s” | “cobbler’s” | “chemist’s

T → “shoe” | “hat” | “cigar” | “beer” | “cake” | “suit” | “tire”

To make the story interesting, we want the old lady to go out shopping several times. Where in the grammar do we make this possible?

____

Note that different grammar rules for the same variable are separated by the '|' symbol, which basically means "or". For instance, we have 6 possible places to shop, and 7 things she could buy. The way the grammar is written, these are independent choices because the choices are specified in different variables. We are going to use random numbers to choose which rule gets chosen for a variable whenever there is a choice.

Part 1: Finish implementation of Story1.java

All you need to do is complete the methods called B( ), P( ) and T( ). Note that these methods should follow very closely to the rules given in the grammar above.

Compile your program, and run it several times. You should be able to observe that many different items are bought from the various venues. √

Part 2: Enhancing the story

Copy Story1.java into a new source file called Story2.java. This will be an enhanced version of the problem. This time, we want to tell what happens while Old Mother Hubbard is away from the house. In her absence, the dog engages in some mischief. Eventually, we want our story to look something like the following example output.

Old Mother Hubbard went to the cupboard
To fetch her poor dog a bone;
But when she came there the cupboard was bare,
And so the poor dog had none.

She went to the cobbler's to buy him a cigar
But when she came back, he was dancing with the guitar

She went to the tavern to buy him a shoe
But when she came back, he was burning the flue

She went to the cobbler's to buy him a cake
But when she came back, he was eating the rake

What is new in Story2.java is that we are adding a sentence each time the old lady goes somewhere. The new sentence has the form:

> But when she came back, he was <doing something> with the <name of victim>.

There is one interesting constraint about this sentence, and this leads us to an important observation. Notice the words cigar and guitar; shoe and flue; cake with rake. They rhyme. In other words, the item being bought by the old lady has to rhyme with the thing being abused by the dog. How can we make sure that these two words always rhyme? The answer can be found if we look back to the Lobster program, which did the same thing.

The key observation is that the choice of item to buy and dog's victim need to be selected at the same time. In other words, they must be specified in the same rule. The item to buy is being specified in rules for the variable T. So, we need to modify the grammar so that we also select the appropriate victim word that rhymes with what is being bought.

For our grammar, we also need to add some new variables and rules. Up until now, our grammar has rules for S, A, B, P and T. In addition to modifying the rules for T, we will add two new variables: M and D. The purpose of the M text is to specify all of the words between the two words that rhyme. The purpose of the D variable is to specify what the dog could be doing to its victim.

Here are the 7 choices for victim, and note that each one rhymes with an item to buy:

- flue
- cat
- guitar
- steer
- rake
- loot
- fire

And here are the 8 things the dog can do to its victim

- eating
- burning
- tearing up
- selling
- standing on
- dancing with
- burying
- chasing

Complete the new grammar below:

S → "Old Mother Hubbard went to the cupboard to fetch her poor dog a bone. But when she came there the cupboard was bare, and so the poor dog had none." A

A → "" | BA

B → "She went to the" P "to buy him a" T

P → "baker's" | "tavern" | "grocer's" | "tailor's" | "cobbler's" | "chemist's

T → ____

____

____

M → ____

D → ____

I encourage you to show me your grammar before you make significant changes to Story2.java. Once you are satisfied with the grammar, modify the code in Story2.java accordingly. Compile your program, and run it several times. Verify that the item being bought always rhymes with the dog's victim. √