

CSCE-638, Programming Assignment #3 CKY

Due: Wed, Nov. 6th, 2019 by 11:00pm

For this assignment, you should write a program that implements the classic CKY algorithm for syntactic parsing. No starter code is provided this time since we expect you can handle file IO etc by now. As we discussed in class, the CKY algorithm is a dynamic algorithm to efficiently find the most likely syntactic parse tree for a sentence. As input, your program will need two files: (1) a file of grammar rules with probabilities, and (2) a file of sentences to process. Your program should get the names of the 2 input files from the command line. The command-line arguments should be in the following order:

```
CKY <grammar_file> <sentences_file>
```

```
CKY grammar_rules.txt sents.txt
```

1. The grammar_rule.txt File

This file will contain grammar rules with probabilities in the Chomsky normal form.

2. The Sentences File

This file will contain a set of sentences that you should run your program on. Each sentence will be on a separate line. There will not be any punctuation marks in this file. A sample file might look like this:

```
This is a sentence  
And here is another sentence
```

OUTPUT FORMATTING

The output produced by your program should consist of 2 items for each sentence that is processed:

1. The sentence being processed.
2. All final and intermediate probabilities generated during the parsing process. Note that probabilities should be printed in the order of outside-to-inside, left-to-right, as shown in the example output file.

IMPORTANT: When you print this information, please format it exactly like the given example outputs. Since you'll be printing a lot of numbers, this is crucial to ensure that we know exactly what your numbers and results correspond to.

GRADING CRITERIA

Your program will be graded based on both given dev cases (50%) and new test cases (50%)! **So please test your program thoroughly to evaluate the generality and correctness of your code!** Even if your program works perfectly on the examples that we give you, that does not guarantee that it will work perfectly on additional test cases.

ELECTRONIC SUBMISSION INSTRUCTIONS **(a.k.a. “What to turn in and how to turn in”)**

You need to submit 2 things:

1. The source code files for your program. Be sure to include all files that we will need to compile and run your program!
2. A report file that includes the following information:
 - how to compile and run your code
 - results and analysis
 - any known bugs, problems, or limitations of your program

REMINDER: your program **must** compile and run. We will not grade programs that cannot run.

How to turn in:

1. please include everything in a single .zip package and submit it on ecampus.