CS 410 Project 2 Report

Umut Oskay

Context Free Grammar to Chomsky Normal Form Conversion

Abstract

In this assignment we will convert the given context free grammar to Chomsky normal form. By only using main class. It will be a console application. Results will be shown in console.

Main Class

In main class here is the steps for converting Context Free Grammar to Chomsky Normal Form:

- First, file is read by the FileReader class.
- nonTerminalsList, rules HashMap and start state, terminalList is created.
- I put my code on try catch in order to prevent any errors.
- Since normal reading doesn't have good functionalities, I read the file with bufferedReader.
- Now the story begins with while loop in order to read all of the text file until its null.
- We check the keywords "NON-TERMINAL, TERMINAL, RULES, START".
- For NON-TERMINAL keyword if line starts with NON-TERMINAL, we will read the inputs and fill nonTerminalsList until it sees TERMINAL keyword.
- For TERMINAL it does exactly the same with above but terminalList is filled in here.
- For RULES keyword, we need to do something different here. First we need to read until start and in each iteration we need to split the information into two parts by using split function. We are holding the data like key value pair and we will be mapping the rules corresponding each other.
- For START keyword, line is simply read by buffered reader.
- You can see the stack trace if you have an error in try catch mechanism.

File is read and required lists have been filled until here. Now, program is ready to convert context free grammar to Chomsky Normal Form.

- First, we created newNonTerminal string we will be using this string all over the operation.
- By looping around nonTerminalList and ruleList we need to update newNonTerminal and we should add it to the list.
- **In runtime, first new S will be created, after epsilons will be removed, null productions will be handled, unit productions and handled after that proper variable changes will be done.**
- Also, newRuleList is created because previous rules will not be valid anymore.
- There is the critical part happens. We need to check if length is longer than 2 so that we can select the ones that disobey the Chomsky normal form rule. Which is A->a or B -> CD.
- We change the rule list according to the length of right hand side.
- Finally, new rulelist will be filled and previous rules will be removed from the console.

Logic is implemented. Result should be seen on console. Here is how I show them in console:

- By iterating on each list, lists are printed in order to see the changes in rules.