**CS323 Documentation**

About 2-3 pages

# 1. Problem Statement

This program will read the text file that has been inputted and process the text stream. Each lexeme in the text file is filtered to find the corresponding token. If the lexeme is an identifier it is entered on the symbol table and given a memory address. There is a function that checks if the identifier is already in the symbol table. The program will execute the instructions on the text file using a stack. There is also an assembly array that keeps track of all of the instructions that happen in the program.

**2. How to use your program**

My program is an executable file called Project 3. Once the program is running you can select from three different test files. Each test file tests different inputs on the program. To select a file type “test1.txt “. The program will execute printing out the grammar used in the file, the assembly array, and the symbol table. After the program prints you can choose another test file, or you can exit the program by entering “0”. When you open the file where the program is stored you can edit the different text files that will allow you test different situations.

# 3. Design of your program

My program stores the lexemes and tokes in a vector of vector pairs. The newest addition to the program will read this complete vector analyze what type of statement the line is and send the program into a state. Depending on the state of the program different actions will be taken but in the end the functions and operations are added to the stack. For the stack I am using an array of integers with size of 100. I am using a vector of tuples for the symbol table. The layout is string, integer, string. For the assembly array I am using an array of pairs set to the size of 1000.

# 4. Any Limitation

None.

# 5. Any shortcomings for each iteration

I could not implement the functions output, or input. The input code has to be in c++ syntax.

# Software Requirements Document Template

## 1. Introduction

### **1.1 Purpose**

The purpose of the program is to take an input as text document, parse the code, and output the grammar rules that were used.

## 2.Overall Description

### **2.1 Product Functions**

The program can take an input file as a text document. It will the parse though the document classifying each lexeme with their respective token. The program then stores the token and lexeme pair. After storing the program will then read the input and determine what grammar rules were used in each statement. The stored file is read by the newly implemented functions and execute the steps on the text document. You can choose from three different text document and you can also edit these text document to run your own code.

### **2.2 Operating Environment**

The operating Environment is a computer running the current version of windows.

### **2.3 Design and Implementation Constraints**

The design of my program made is so that input code in the text document and has to be in c++ syntax.

### **2.4 Assumptions and Dependencies**

 I assume that the program being input is of correct syntax and there are no errors. The program will not catch errors such as using a variable that you have not initialized yet.