CS4542 Compiler Design

Semester Project Part 1 – WinZigC Parser

Important:
This project will be graded on an Ubuntu 18.04 LTS machine.

Project Description

You are required to implement a lexical analyzer and a parser (recursive descent) for the language WinZigC. Refer WinzigC_Lex.pdf for the lexical rules and WinZigC_Grammar.pdf for the grammar details.
You should not use 'lex'. 'yacc' or any such tool.
You are allowed to use only **C/C++ or java** for this project.

Regarding grading and submission:

1. **Submit your source code as a single tar file via Moodle. Name the .tar file as project1<index no>.tar. Replace <index no> with your index number.**

   Do not use nested directories. It should unpack into a single directory.

2. **Please check the test programs in winzig_test_programs. You should match your output with the output from the test programs.**

3. **Grading:**
   Grading will be based on the output of your program. So please make sure your program runs without any errors and produce the exact output as in the test programs.

4.
   a. **Executing the Program for C/C++:**

You should create an executable file named winzigc. Provide a makefile to build your project. Make sure your program can be run with the following sequence.

```
> tar xvf  <submission_file>.tar
>make
> ./winzigc –ast winzig_test_programs /winzig_01 > tree.01
> diff tree.01 winzig_test_programs /winzig01.tree
> ./winzigc –ast winzig_test_programs /winzig_02 > tree.02
```

```
    > diff tree.02 winzig_test_programs /winzig02.tree
 …
```

b. Executing the Program for java:

You should create winzigc.java as the main class. Provide a makefile to build you project. Make sure your program can be run with the following sequence.

```
    > tar xvf  <submission_file>.tar
    >make
    > java winzigc –ast winzig_test_programs /winzig_01 > tree.01
    > diff tree.01 winzig_test_programs /winzig01.tree
    > java winzigc –ast winzig_test_programs /winzig_02 > tree.02
    > diff tree.02 winzig_test_programs /winzig02.tree
```

.