Lab 2: Lexical Analysis

[Update 2018-9-25 21:54 PM]

We have added an extra testcase named as **testcases/test52.tig** and the corresponding output named as **refs/test52.out** to test escape sequences. Please make sure you have these two files in your lab environment, if you don't have them, you should redownload the lab environment in the below **Environment** section. And if you have submitted your lab, you should also redownload the new lab environment, and submit your lab again once you have passed all the testcases.

Description

Use Lex to implement a lexical analyzer for the Tiger language. Appendix A describes, among other things, the lexical tokens of Tiger.

This chapter has left out some of the specifics of how the lexical analyzer should be initialized and how it should communicate with the rest of the compiler. You can learn this from the Lex manual, but the "skeleton" files in the lab directory will also help get you started.

Along with the tiger.lex file you should turn in documentation for the following point:

• how you handle comments;

• how you handle strings;

• error handling;

• end-of-file handling;

• other interesting features of your lexer.

Notice: Before you start this lab, you should carefully read the chapter 2 of the textbook, and you may need to get some help from the lex manual and the *Tiger Language Reference Manual* (Appendix A). As usual, if you have any question about this lab, feel free to contact Haoyu Li, who is the teaching assistant responsible for lab 2.

Environment

You can download the lab environment here, and then decompress it to your current directory by the following command.

```
shell% tar -zxvf lab2.tar.gz
```

Moreover, if your system lacks Lex and Yacc, you can install them with the following command (on Ubuntu).

```
shell% sudo apt-get install flex bison
```

Grade Test

The lab environment contains a grading script named as **gradeMe.sh**, you can use it to evaluate your code, and that's how we grade your code, too. If you pass all the tests, the script will print a successful hint, otherwise, it will output some error messages. You can execute the script with the following commands.

```
shell% ./gradeMe.sh
shell% ...
shell% [^_^]: Pass #If you pass all the tests, you will see these messages.
shell% TOTAL SCORE: 100
```

Handin

The deadline of this lab is on Friday 12:00 AT NOON, Oct 12, 2018, and no delay is allowed!

Afte you have passed the grade test, you need first package your code and rename the file lab2_xxx.tar.gz to lab2_[your student ID].tar.gz. For example, if your student ID is 516037900000, then the file name should be lab2_516037900000.tar.gz, and no any other letters are included. You can use the following commands to finish this step.

```
shell% make handin #This command will do the packaging for you, generating a file named as lab2_xxx.tar.gz
shell% mv lab2_xxx.tar.gz lab2_516037900000.tar.gz
```

In the end, you need to submit the renamed tar.gz file to ftp://haoyu.li:public@public.sjtu.edu.cn/upload/lab2 before the deadline.

Go to Top // *Compilers* Home Page
Questions or comments regarding *Compilers* course? Send e-mail to the course Staffs or TAs.
Last updated: Mon Sep. 17 2018