

FINAL PROJECT

LING 388

The final project will consist of a flex scanner and bison parser, similar to those we have done in class. The scanner will be worth 10 points, and the parser will also be worth 10 points for a total of 20 points. Up to 10 extra credit points will be given for a python command handler described below. Please email me your scripts no later than noon on Friday December 18th — I would prefer that you email me sooner if you can manage.

The main idea of this project is to use some of the tools we have covered during the semester to create a program that accepts input strings such as ‘computer wiki robot’, ‘computer google linguistics’, or ‘computer map Kenya’. The program will then parse these strings and convert them into commands that the computer can understand, such as ‘open <http://en.wikipedia.org/wiki/robot>’.

1. FLEX SCANNER

10 pts. Write a scanner that can recognize the following:

- A keyword that calls the system, such as ‘computer’
- At least 5 commands, such as ‘wiki’, ‘google’, ‘map’, ‘translate’, ‘restaurant’, ‘etc.’
- Arguments for the commands – this can include recognizing anything as a valid argument. Arguments can be shared by separate commands

Your scanner should be formatted so that it works with the bison parser.

2. BISON PARSER

10 pts. Write a parser that accepts commands that contain a CALL followed by a COMMAND and ARGUMENTS, such as ‘computer wiki robot’. The parser should execute some action code that is appropriate for the command that was given. For example, if the command is ‘wiki’, the action code should do something with wikipedia, such as print a command that has the relevant wikipedia URL for the given argument. So if the input is ‘computer wiki robot’, the parser should print something similar to ‘open <http://en.wikipedia.org/wiki/robot>’.

Please include instructions for how to compile the parser and scanner together. This could be in the form of a Makefile or shell script. Please include a brief description of what your program does, i.e. explain what commands it understands and how to call the program.

3. EXTRA CREDIT

Up to 10 pts. For extra credit, write a Python script that listens to the output of the bison parser and then executes the commands in the output string. For example, if the bison parser prints ‘open <http://en.wikipedia.org/wiki/robot>’, your

Date: Due December 18, 2009.

script should go and open the URL in a browser. This can be accomplished by piping the output of the parser to the python script:

```
$ ./parser | python CommandHandler.py
```

To make this work, you will need to read up on the Python sys module, especially the sys.argv function, and the os module, especially the os.system function:

<http://docs.python.org/library/sys.html>.

<http://docs.python.org/library/os.html>