

Linguistics 555 — Programming for Computational Linguists

Robert Felty

Fall 2008

Instructor: Robert Felty

Class meeting times: Monday and Wednesday, 4:00-5:15 p.m., LH 030

Office Hours: Thursday, 10:00-11:00 a.m., Psychology 186, 812-855-4893

Overview

This course is designed to give linguists a practical foundation in programming, which will allow them to efficiently take advantage of existing tools, as well as create their own tools for a variety of linguistic tasks, including searching corpora and databases, compiling statistics from databases, analyzing experimental data, and preparing stimuli for experiments.

The course will focus primarily on the Python programming language, but will also cover some of the commonly used unix utilities which are handy for linguists.

Goals

By the end of the course, you should be able to:

- Feel comfortable using a unix/linux/mac command line
- Understand key programming concepts such as conditionals, iteration, recursion, functions, and objects
- Be able to write your own programs which can help you to answer linguistic questions and solve everyday problems

Text

Most of the course will use the textbook *Beginning Python: From Novice to Professional*, by Magnus Lie Hetland. All readings are from this book unless otherwise noted.

Grading

The focus of this course is on practical applications. The grading also reflects this, in that the majority of the grade will come from homework assignments.

The homework assignments will mostly be small programming problems of the sort that linguists frequently deal with. All homework assignments will be due on Mondays before the start of the class, and should be submitted electronically. Homework should include all source code, with meaningful comments, and input and output where appropriate.

Since we will want to discuss homework solutions in class while it is still fresh in our minds, **late homework will not be accepted.**

class participation	10%
homework assignments (11 total — one can be dropped)	60%
final presentation	10%
final paper/project	20%

Final presentation / project

The culmination of the course will be a final project of your choosing. For this project, you should choose some task or problem relevant to your research interests, and write a program which solves this problem. You will be asked to give a short presentation outlining the problem and your solution to it, and finally will turn in a working program with thorough documentation.

Depending on the scope of the problem, it may be suitable to only solve a particular subset of the possible scenarios. Please begin to think about possible projects as soon as possible and discuss them with me.

Examples of possible projects:

- Develop a custom GUI application to control psycholinguistic experiments
- Write a program which performs a complex search of a linguistic corpus or database and computes some sort of statistics about it
- Use the natural language toolkit to parse a grammar
- Write a program which translates a corpus from one transcription/tagging system into another
- Write a program which selects stimuli from a database for a psycholinguistic experiment based on a number of different criteria

Calendar (tentative)

date	topic	reading	assignment due
Wed Sep 3	How programming will make your life easier. Intro to unix	none	none
Mon Sep 8	Common unix utilities		hwk #0
Wed Sep 10	More Unix utilities; Globs		
Mon Sep 15	Regular Expressions	Ch. 10, pp. 235–245	
Wed Sep 17	Intro to python	Ch. 1	hwk #1
Mon Sep 22	Lists and Tuples;	Ch. 2	hwk #2
Wed Sep 24	Strings	Ch. 3	
Mon Sep 29	Dictionaries (Hashes)	Ch. 4	hwk #3
Wed Oct 1	Conditionals and Loops	Ch. 5	
Mon Oct 6	File Input and Output	Ch. 11	hwk #4
Wed Oct 8	NO CLASS - I am at a conference		

date	topic	reading	assignment due
Mon Oct 13	More conditionals and Loops Using linguistic corpora and databases		hmwk #5
Wed Oct 15	Word frequency and co-occurrence Functions and procedures	Ch. 6	
Mon Oct 20	More on Functions		hmwk #6
Wed Oct 22	Object-oriented programming	Ch. 7	
Mon Oct 27	More on Object-oriented programming		hmwk #7
Wed Oct 29	Handling errors and exceptions	Ch. 8	
Mon Nov 3	More on errors and exceptions N-grams, Markov analysis		hmwk #8
Wed Nov 5	More on object-oriented programming Constructors, methods, and iterators	Ch. 9	Project Proposal
Mon Nov 10	Version control, edit distance, and more object-oriented programming		hmwk #9
Wed Nov 12	modules & command line arguments	Ch. 10	
Mon Nov 17	More on modules		hmwk #10
Wed Nov 19	Sharing programs (distutils)	Ch. 18	
Mon Nov 24	Verb finding, Zipf's Law	<i>Think Python</i> , pp. 125–134	hmwk #11
Wed Nov 26	NO CLASS – happy thanksgiving		
Mon Dec 1	Final presentations		
Wed Dec 3	Final presentations		
Wed Dev 17	FINAL PROJECT DUE		