Programming Languages
## Class Website

http://www.cs.utexas.edu/users/cannata/cs345/

<table>
<thead>
<tr>
<th>Dr. Philip Cannata</th>
<th>Fall 2011</th>
<th>Unique Number: #52413</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Room Number - GEA 105</strong></td>
<td><strong>Map</strong></td>
<td><strong>MW 4:5:30</strong></td>
</tr>
</tbody>
</table>

**TAs -->**

- Martin Georgiev  
  mgeorgiev@utexas.edu  
- Ebrahim Rajabzadeh  
  ebrahim@cs.utexas.edu

**Office Hours**

- Martin Georgiev  
  Monday - 10 am - 11 am  
  Friday - 1 pm - 2 pm  
  The location is: PAI 5.33.
- Ebrahim Rajabzadeh  
  Tuesdays and Thursdays  
  2:00-3:00 @ ENS 1  
  (ENS Building Room #1 which is the basement computer lab)

**Administration and Policies**

- Check your grades within Blackboard
- This is for Dr. Cannata: Setting up black board. Go to clips and then authorize assistants.

**Piazza**

**Calendar**

/* for Dennis Ritchie */
#include<stdio.h>
main() { printf("Goodbye World"); }

John McCarthy - cdr in peace
ON PROOF AND PROGRESS IN MATHEMATICS
WILLIAM P. THURSTON

Most of the audience at an average colloquium talk gets little of value from it. Perhaps they are lost within the first 5 minutes, yet sit silently through the remaining 55 minutes. Or perhaps they quickly lose interest because the speaker plunges into technical details without presenting any reason to investigate them. At the end of the talk, the few mathematicians who are close to the field of the speaker ask a question or two to avoid embarrassment. This pattern is similar to what often holds in classrooms, where we go through the motions of saying for the record what we think the students “ought” to learn, while the students are trying to grapple with the more fundamental issues of learning our language and guessing at our mental models. Professors compensate by giving homework and tests that are much easier than the material “covered” in the course, and then grading the homework and tests on a scale that requires little understanding. We assume that the problem is with the students rather than with communication: that the students either just don’t have what it takes, or else just don’t care. ...

In reaction to my experience with foliations and in response to social pressures, I concentrated most of my attention on developing and presenting the infrastructure in what I wrote and in what I talked to people about. I explained the details to the few people who were “up” for it. I wrote some papers giving the substantive parts of the proof of the geometrization theorem for Haken manifolds—for these papers, I got almost no feedback. Similarly, few people actually worked through the harder and deeper sections of my notes until much later. The result has been that now quite a number of mathematicians have what was dramatically lacking in the beginning: a working understanding of the concepts and the infrastructure that are natural for this subject. …

What mathematicians most wanted and needed from me was to learn my ways of thinking, and not in fact to learn my proof of the geometrization conjecture for Haken manifolds.
# CS345 Programming Language Concepts

<table>
<thead>
<tr>
<th>Concept</th>
<th>Notes</th>
<th>Concept</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Token</td>
<td>HLL1 &amp; HLL2</td>
<td>Functions</td>
<td>HLL4</td>
</tr>
<tr>
<td>Terminal</td>
<td>HLL1 &amp; HLL2</td>
<td>Parameters and Arguments</td>
<td>HLL4</td>
</tr>
<tr>
<td>Non-terminal</td>
<td>HLL1 &amp; HLL2</td>
<td>Parameter Passing</td>
<td>HLL4</td>
</tr>
<tr>
<td>Lexical Analysis</td>
<td>HLL1 &amp; HLL2</td>
<td>Activation Records</td>
<td>HLL4</td>
</tr>
<tr>
<td>Regular Expression</td>
<td>HLL1 &amp; HLL2</td>
<td>Runtime Stack</td>
<td>HLL4</td>
</tr>
<tr>
<td>Syntactic Analysis</td>
<td>HLL1 &amp; HLL2</td>
<td>Side Effects</td>
<td>HLL4</td>
</tr>
<tr>
<td>Regular Grammar</td>
<td>HLL1 &amp; HLL2</td>
<td>First-Class Functions</td>
<td>HLL4 Chapters 5 &amp; 6</td>
</tr>
<tr>
<td>Context-free Grammar</td>
<td>HLL1 &amp; HLL2</td>
<td>Recursion</td>
<td>HLL4</td>
</tr>
<tr>
<td>BNF</td>
<td>HLL1 &amp; HLL2</td>
<td>Closure</td>
<td>HLL4</td>
</tr>
<tr>
<td>EBNF</td>
<td>HLL1 &amp; HLL2</td>
<td>Tail Recursion</td>
<td>HLL4</td>
</tr>
<tr>
<td>Parse Tree</td>
<td>HLL1 &amp; HLL2</td>
<td>Haskell</td>
<td>Haskell1 Chapters 7 &amp; 8</td>
</tr>
<tr>
<td>Associativity</td>
<td>HLL1 &amp; HLL2</td>
<td>Haskell Expression</td>
<td>Haskell1</td>
</tr>
<tr>
<td>Precedence</td>
<td>HLL1 &amp; HLL2</td>
<td>Haskell Functions</td>
<td>Haskell1</td>
</tr>
<tr>
<td>Ambiguous Grammar</td>
<td>HLL1 &amp; HLL2</td>
<td>Haskell Tuples</td>
<td>Haskell1</td>
</tr>
<tr>
<td>Abstract Syntax</td>
<td>HLL1 &amp; HLL2</td>
<td>Haskell Lists</td>
<td>Haskell1</td>
</tr>
<tr>
<td>Abstract Syntax Tree</td>
<td>HLL1 &amp; HLL2</td>
<td>Haskell Lists</td>
<td>Haskell1</td>
</tr>
<tr>
<td>Interpretation</td>
<td></td>
<td>Haskell Pattern Matching</td>
<td>Haskell1</td>
</tr>
<tr>
<td>Substitution</td>
<td>PLA1 2</td>
<td>List Comprehension</td>
<td>Haskell1</td>
</tr>
<tr>
<td>Symbol Table and Dictionaries</td>
<td>HLL3 &amp; PLA3</td>
<td>Lambda Expression</td>
<td>Haskell1</td>
</tr>
<tr>
<td>Scope</td>
<td>HLL3 &amp; PLA3</td>
<td>Lazy Evaluation</td>
<td>Haskell1 Chapters 7 &amp; 8</td>
</tr>
<tr>
<td>Nested and Dynamic Scope</td>
<td>HLL3 &amp; PLA3</td>
<td>Strong Type System</td>
<td>Haskell1</td>
</tr>
<tr>
<td>Local and Non-Local Variables</td>
<td>HLL3 &amp; PLA3</td>
<td>Propositional Logic</td>
<td>Haskell2</td>
</tr>
<tr>
<td>Static and Dynamic Scoping</td>
<td>HLL3 &amp; PLA3</td>
<td>Haskell and SQL</td>
<td>Haskell2</td>
</tr>
<tr>
<td>Resolving References</td>
<td>HLL3 &amp; PLA3</td>
<td>Syllogisms</td>
<td>Chapters 1 &amp; 2 Syllogism and Midter Review</td>
</tr>
<tr>
<td>Overloading</td>
<td>HLL3 &amp; PLA3</td>
<td>Proof by Contradiction</td>
<td>Syllogism and Midter Review</td>
</tr>
<tr>
<td>Variables</td>
<td>HLL3 &amp; PLA3</td>
<td>Godel’s Incompleteness Theorem</td>
<td>Prolog 1 &amp; 2</td>
</tr>
<tr>
<td>Static and Dynamic Binding</td>
<td>HLL3 &amp; PLA3</td>
<td>Deductive Logic</td>
<td>Prolog 1 &amp; 2</td>
</tr>
<tr>
<td>Types</td>
<td></td>
<td>Prolog</td>
<td>Prolog 1 &amp; 2</td>
</tr>
<tr>
<td>Type System</td>
<td>PLA1 Types &amp; HLL1</td>
<td>Prolog Facts and Rules</td>
<td>Prolog 1 &amp; 2</td>
</tr>
<tr>
<td>Static and Dynamic Typing</td>
<td>PLA1 Types</td>
<td>Prolog Lists</td>
<td>Prolog 1 &amp; 2</td>
</tr>
<tr>
<td>Implicit Type Conversion</td>
<td>PLA1 Types</td>
<td>Proof by Contradiction</td>
<td>Prolog 1 &amp; 2</td>
</tr>
<tr>
<td>Type Judgments</td>
<td>PLA1 Types</td>
<td>Unification</td>
<td>Prolog 1 &amp; 2</td>
</tr>
<tr>
<td>Type Judgment Tree</td>
<td>PLA1 Types</td>
<td>Resolutions</td>
<td>Prolog 1 &amp; 2</td>
</tr>
<tr>
<td>Type Soundness</td>
<td>PLA1 Types</td>
<td>Backtracking</td>
<td>Prolog 1 &amp; 2</td>
</tr>
<tr>
<td>Type Safety</td>
<td>PLA1 Types</td>
<td>Constraint Satisfaction Programming</td>
<td>Prolog 1 &amp; 2</td>
</tr>
<tr>
<td>Type Inferencing</td>
<td>PLA1 Types</td>
<td>Prolog and SQL</td>
<td>Prolog 1 &amp; 2</td>
</tr>
<tr>
<td>Relations</td>
<td>PLA1 1</td>
<td>Semantic Web Programming</td>
<td>Semantic Web &amp; Semantic DBs and OO PLs</td>
</tr>
<tr>
<td>Primitive Recursive Functions</td>
<td>Genesis</td>
<td>Compressions</td>
<td>Prolog 1 Page 9</td>
</tr>
<tr>
<td>Lambda Calculus</td>
<td>PLA1 1 Chapter 22</td>
<td>Compressions</td>
<td>Prolog 1 Page 9</td>
</tr>
<tr>
<td>Combinator Logic</td>
<td>Genesis</td>
<td>Kolmogorov Complexity</td>
<td></td>
</tr>
<tr>
<td>Lisp</td>
<td>PLA1</td>
<td>Semantic Web Programming</td>
<td></td>
</tr>
</tbody>
</table>

Dr. Philip Cannata
You may need two brains

<table>
<thead>
<tr>
<th>Date</th>
<th>Topic</th>
<th>Chapter Reading</th>
<th>PPT Notes</th>
<th>PDF Notes</th>
<th>Homework</th>
</tr>
</thead>
<tbody>
<tr>
<td>Wednesday, August 24, 2011</td>
<td>Introduction and Tools</td>
<td>Purple are Project suggestions.</td>
<td>Intro &amp; Tools</td>
<td>Intro &amp; Tools</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Syntax, Grammar, Parsing and Interpretation</td>
<td></td>
<td>HLL 1</td>
<td>HLL 1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>javacc tutorial</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>javacc FAQ</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>JavaWorld javacc</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>javacc demo1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>javacc demo2</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Unix tutorial</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>sed one liners</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>javacc demo3</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Monday, August 29, 2011</td>
<td>Modeling Languages and Interpreting Arithmetic</td>
<td>PLAI Book: Chapter 22, 1, and 2</td>
<td>PLAI 1</td>
<td>PLAI 1</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Lisp with javacc - JDBLispv2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MetaMath pdf</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Getting started downloads</td>
<td>JDBLispv2 Doc</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>JDBLisp Email</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Wednesday, August 31, 2011</td>
<td>Chomsky, Lexical Analysis and Parsing, H--</td>
<td></td>
<td>HLL 2</td>
<td>HLL 2</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Hmm Grammar and Abstract Syntax</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Monday, September 03, 2011</td>
<td>Substitution</td>
<td>PLAI Book: Chapter 3</td>
<td>PLAI 2</td>
<td>PLAI 2</td>
<td></td>
</tr>
</tbody>
</table>

Dr. Philip Cannata
High Level Languages

Virtual Machines and Interpreters

1. Source Program
2. Lexical Analyzer
3. Syntactic Analyzer
4. Type Checker
5. Abstract Syntax
6. Interpreter
7. Input
8. Computer
9. Output
Relation-based (Functional) Languages

\[ \lambda f. (\lambda x. (f (x x))) (\lambda x. (f (x x))) \]
When one considers how hard it is to write a computer program even approaching the intellectual scope of a good mathematical paper, and how much greater time and effort have to be put into it to make it “almost formally correct, it is preposterous to claim that mathematics as we practice it is anywhere near formally correct. …

Godel’s incompleteness theorem implies that there can be no formal system that is consistent, yet powerful enough to serve as a basis for all of the mathematics that we do. …

Mathematics as we practice it is much more formally complete and precise than other sciences, but it is much less formally complete and precise for its content than computer programs. The difference has to do not just with the amount of effort: the kind of effort is qualitatively different. In large computer programs, a tremendous proportion of effort must be spent on myriad compatibility issues: making sure that all definitions are consistent, developing “good” data structures that have useful but not cumbersome generality, deciding on the “right” generality for functions, etc. The proportion of energy spent on the working part of a large program, as distinguished from the bookkeeping part, is surprisingly small. Because of compatibility issues that almost inevitably escalate out of hand because the “right” definitions change as generality and functionality are added, computer programs usually need to be rewritten frequently, often from scratch.
The tools you will need to get started

javac
javacc

cygwin (if you’re going to be using Windows)

Ant

jython

See next pages for downloading and installation

jDeveloper (see instruction doc on class calendar)

SQLDeveloper (see instruction doc on class calendar)
Installing javac on Windows


- Change your path environment variable to include the path to javac
## Installing javacc on Windows

<table>
<thead>
<tr>
<th>Date</th>
<th>Topic</th>
<th>Chapter Reading</th>
<th>PPT Notes</th>
<th>PDF Notes</th>
<th>Homework</th>
<th>Tests</th>
<th>Files for Assignments and Solutions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Wednesday, January 19, 2011</td>
<td>Introduction and Tools</td>
<td>Intro &amp; Tools</td>
<td>Intro &amp; Tools</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Monday, January 24, 2011</td>
<td>Modeling Languages and Interpreting Arithmetic</td>
<td>PLAI Book: Chapter 22</td>
<td>Lisp</td>
<td>Lisp</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Monday, January 31, 2011</td>
<td>Substitution</td>
<td>PLAI Book: Chapter 3</td>
<td>PLAI 2</td>
<td>PLAI 2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Wednesday, February 02, 2011</td>
<td>Python and Lambda Calculus</td>
<td>Abth Book: Chapter 1, Section 2.7 and Chapter 3</td>
<td>Hll 3</td>
<td>Hll 3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Monday, February 07, 2011</td>
<td>Abth and Javacc Formatters (Transformers)</td>
<td>Abth and Javacc Formatters (Transformers)</td>
<td>PLAI 3</td>
<td>PLAI 3</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

---

Dr. Philip Cannata
Installing javacc on Windows

http://java.net/projects/javacc/
Installing javacc on Windows

• Download javacc from http://java.net/projects/javacc/

• I put the zip file in C:\Program Files\Java and extracted it to there. So, javacc is in
  C:\Program Files\Java\javacc-5.0\javacc-5.0\bin

• Change your path environment variable to include C:\Program Files\Java\javacc-5.0\javacc-5.0\bin or whatever your path is. Do the following to do this:
  path environment variables are in the System Properties -> Advanced Tab -> Environment Variables?
  Add the javacc path to the front of the existing “path” environment variable followed by a semi-colon?

• Edit the javacc program as follows:

```bash
#!/bin/sh
JAR="`dirname "$0"`/lib/javacc.jar"

case "`uname`" in
  CYGWIN*) JAR="`cygpath --windows -- "$JAR"`" ;;
esac

java -classpath "$JAR" javacc "$@"
```

Put double quotes around $0
Greetings,

We've installed javacc, the java compiler compiler. This is a parser generator for java. See https://javacc.dev.java.net/ for details.

This will be made available on all departmental machines after our nightly cron job runs this evening. Javacc will be in the default $PATH, so you can access it simply by running 'javacc' from a terminal.

However, if you need to use the javacc.jar file, you will need to add its location to your $CLASSPATH environmental variable. I have created a set of shell scripts to ease this process.

If you use bash, run the following:
source /lusr/share/lib/java/javacc/env.d/append.sh

If you run csh or tcsh, run this instead:
source /lusr/share/lib/java/javacc/env.d/append.csh

Please send mail to gripe if you have any questions.
-jason
Installing Cygwin from http://cygwin.com/

Cygwin

Get that Linux feeling - on Windows!

This is the home of the Cygwin project

What...

...is it?
Cygwin is:

- a collection of tools which provide a Linux look and feel environment for Windows.

- a DLL (cygwin1.dll) which acts as a Linux API layer providing substantial Linux API functionality.

...isn’t it?
Cygwin is not:

- a way to run native Linux apps on Windows. You must rebuild your application from source if you want it to run on Windows.

- a way to magically make native Windows apps aware of UNIX® functionality like signals, ptrs, etc. Again, you need to build your apps from source if you want to take advantage of Cygwin functionality.

The Cygwin DLL currently works with all recent, commercially released x86 32 bit and 64 bit versions of Windows, with the exception of Windows CE.

For more information see the FAQ.

Current Cygwin DLL version

The most recent version of the Cygwin DLL is 1.7.7.1. Install it by running setup.exe.

Use setup.exe to perform a fresh install or to update an existing installation.

Note that individual packages in the distribution are updated separately from the DLL, so the Cygwin DLL version is not useful as a general Cygwin release number.
Download Cygwin Install File (Optional)

Cygwin
Get that Linux feeling - on Windows!

Installing and Updating Cygwin

Run setup.exe any time you want to update or install a Cygwin package. The signature for setup.exe can be used to verify the validity of this binary using this public key.

When installing packages for the first time, setup.exe does not install every package. Only the minimal base packages from the Cygwin distribution are installed by default. Clicking on categories and packages in the setup.exe package installation screen will provide you with the ability to control what is installed or updated. Clicking on the "Default" field next to the "All" category will provide you with the opportunity to install every Cygwin package. Be advised that this will download and install hundreds of megabytes to your computer. The best plan is probably to click on individual categories and install either entire categories or packages from the categories themselves.

The latest net releases of the Cygwin DLL are numbered l.n.x, where "n" is currently "7" (e.g., 1.7.5). The l.n.x version numbering refers only to the Cygwin DLL. Individual packages like bash, gcc, less, etc. are released independently of the DLL. The setup.exe utility tracks the versions of all installed components and provides the mechanism for installing or updating everything available from this site for Cygwin.

Once you've installed your desired subset of the Cygwin distribution, setup.exe will remember what you selected so re-running the program will update your system with any new package releases.

Q: Is there a command-line installer?

A: Yes and no. The setup.exe program understands command-line arguments which allow you to control its behavior and choose individual packages to install. While this provides some functionality similar to such tools as apt-get or yum it is not as full-featured as those packages.

The basic reason for not having a more full-featured package manager is that such a program would need full access to all of Cygwin's POSIX functionality. That is,
Download Cygwin Install File (Optional)

1. **File Download - Security Warning**
   - Do you want to run or save this file?
   - Name: setup.exe
   - Type: Application.exe
   - From: cygwin.com
   - Options: Run, Save, Cancel

2. **Internet Explorer - Security Warning**
   - The publisher could not be verified. Are you sure you want to run this software?
   - Name: setup.exe
   - Publisher: Unknown Publisher
   - Options: Run, Don’t Run

3. **Cygwin Setup - Choose Installation Type**
   - Choose a download source:
     - Install from Internet (downloaded files will be kept for future re-use)
     - Download Without Installing
     - Install from Local Directory

4. **Cygwin Setup - Choose Installation Type**
   - Options: < Back, Next >, Cancel
Download Cygwin Install File (Optional)

1. Select Local Package Directory
   - Enter `C:\cygwin_download`

2. Select Your Internet Connection
   - Direct Connection

3. Choose Download Site(s)
   - Choose `http://cygwin.mirrors.pair.com`

4. Setup Alert
   - This is the first time you've installed Cygwin 1.7.x. Please be advised that this is a major release. If you have not done so already, please check out the documentation at http://cygwin.com/ to see how the upgrade could potentially affect any existing Cygwin installation.
   - OK

Dr. Philip Cannata
Download Cygwin Install File (Optional)

Don’t take the Default. See next page.
Download Cygwin Install File (Optional)

Click here until you have All Install
Download Cygwin Install File (Optional)
Installing Cygwin

Cygwin
Get that Linux feeling - on Windows!

Installing and Updating Cygwin

Run \texttt{setup.exe} any time you want to update or install a Cygwin package. The \texttt{signature} for \texttt{setup.exe} can be used to verify the validity of this binary using \texttt{this public key}.

When installing packages for the first time, \texttt{setup.exe} \textit{does not install every package}. Only the minimal base packages from the Cygwin distribution are installed by default. Clicking on categories and packages in the \texttt{setup.exe} package installation screen will provide you with the ability to control what is installed or updated. Clicking on the "Default" field next to the "All" category will provide you with the opportunity to install every Cygwin package. Be advised that this will download and install hundreds of megabytes to your computer. The best plan is probably to click on individual categories and install either entire categories or packages from the categories themselves.

The latest net releases of the Cygwin DLL are numbered \texttt{l.n.x}, where "n" is currently "7" (e.g., 1.7.5). The \texttt{l.n.x} version numbering refers only to the Cygwin DLL. Individual packages like \texttt{bash}, \texttt{gcc}, \texttt{less}, etc. are released independently of the DLL. The \texttt{setup.exe} utility tracks the versions of all installed components and provides the mechanism for \textit{installing} or \textit{updating} everything available from this site for Cygwin.

Once you've installed your desired subset of the Cygwin distribution, \texttt{setup.exe} will remember what you selected so re-running the program will update your system with any new package releases.

\textbf{Q: Is there a command-line installer?}

\textbf{A:} Yes and no. The \texttt{setup.exe} program understands \texttt{command-line arguments} which allow you to control its behavior and choose individual packages to install. While this provides some functionality similar to such tools as \texttt{apt-get} or \texttt{yum}, it is not as full-featured as those packages.

The basic reason for not having a more full-featured package manager is that such a program would need full access to all of Cygwin's POSIX functionality. That is, running \texttt{setup.exe} in a host environment that does not have a fully functional POSIX environment is not possible.
Installing Cygwin

1. File Download - Security Warning
   - Do you want to run or save this file?
   - Name: setup.exe
   - Type: Application
   - From: cygwin.com
   - Options: Run, Save, Cancel

2. Internet Explorer - Security Warning
   - The publisher could not be verified. Are you sure you want to run this software?
   - Name: setup.exe
   - Publisher: Unknown Publisher
   - Options: Run, Don't Run

3. Cygwin Setup
   - Cygwin Net Release Setup Program
   - This setup program is used for the initial installation of the Cygwin environment as well as all subsequent updates. Make sure to remember where you saved it.
   - The pages that follow will guide you through the installation. Please note that Cygwin consists of a large number of packages spanning a wide variety of purposes. We only install a base set of packages by default. You can always run this program at any time in the future to add, remove, or upgrade packages as necessary.
   - Setup.exe version 2.738
   - Copyright 2000-2010
   - http://www.cygwin.com/

4. Cygwin Setup - Choose Installation Type
   - Choose A Download Source
   - Choose whether to install or download from the internet, or install from files in a local directory.
   - Options: Install from Internet (downloaded files will be kept for future re-use), Download Without Installing, Install from Local Directory

Dr. Philip Cannata
Installing Cygwin

1. Select Root Install Directory
   - Root Directory: E:\cygwin
   - Install For: All Users (RECOMMENDED)
     - Cygwin will be available to all users of the system.
   - Just Me
     - Cygwin will still be available to all users, but Desktop Icons, Cygwin Menu Entries, and important installer information are only available to the current user. Only select this if you lack Administrator privileges or if you have specific needs.

2. Select Local Package Directory
   - Local Package Directory: E:\cygwin\download

3. Setup Alert
   - This is the first time you've installed Cygwin 1.7.x. Please be advised that this is a major release. If you have not done so already, please check out the documentation at http://cygwin.com/ to see how the upgrade could potentially affect any existing Cygwin installation.
   - If this is the first time you've installed Cygwin on this system then you can ignore this message.

Dr. Philip Cannata
Don’t take the Default. See next page.
Installing Cygwin

Click here until you have All Install
Installing Cygwin
Installing Cygwin

Many hours later
Installing Cygwin

See next page
$ cat .profile
export PATH="/cygdrive/c/ant-1.8.2/bin:$PATH"
export PATH="/cygdrive/c/javacc/javacc-5.0/javacc-5.0/bin:$PATH"
export PATH="/cygdrive/c/java/jdk1.6.0_23/bin:$PATH"
export PATH="/cygdrive/c/Documents and Settings/cannata/My Documents/pcannata/UT/Fall 2011 Projects/Builds/jython/extlibs:$PATH"
export JAVA_HOME='C:\java\jdk1.6.0_23'
export JAVACC_HOME='C:\javacc\javacc-5.0\javacc-5.0\bin\lib'

$ . .profile

$ echo $PATH | tr ":" "\n"
/cygdrive/c/Documents and Settings/cannata/My Documents/pcannata/UT/Fall 2011 Projects/Builds/jython/extlibs
/cygdrive/c/java/jdk1.6.0_23/bin
/cygdrive/c/javacc/javacc-5.0/javacc-5.0/bin
/cygdrive/c/ant-1.8.2/bin
/usr/local/bin
/usr/bin
/cygdrive/c/app/cannata/product/11.2.0/dbhome_1/bin
/cygdrive/c/WINDOWS/system32
/cygdrive/c/WINDOWS
/cygdrive/c/WINDOWS/System32/Wbem
/cygdrive/c/Program Files/Toshiba/Bluetooth Toshiba Stack/sys
/cygdrive/c/Program Files/TortoiseSVN/bin
/usr/lib/lapack

cannata@CANNATA-PC ~
*** Important Cygwin Setup ***

```
Nicolette_Cannata@pennata ~
$ which javac
/cydrive/F/java/jdk1.6.0_23/bin/javac

Nicolette_Cannata@pennata ~
$ which javacc
/cydrive/F/javacc/javacc-5.0/javacc-5.0/bin/javacc

Nicolette_Cannata@pennata ~
$
```
Download jython source

1. Download SVN Client (tortoisesv): http://tortoisesvn.net/downloads.html

2. URL of repository: https://jython.sourceforge.net/svnroot/jython/trunk/jython

3. Create some folder to hold the jython code.
Build jython

1. Download ant: http://ant.apache.org/bindownload.cgi

2. $ cd C:

   $ cd Myjython/jython/

   $ vi setEnv.sh

   $ cat setEnv.sh
   export PATH="/cygdrive/c/ant-1.8.2/bin:$PATH"
   export JAVA_HOME="/cygdrive/c/java/jdk1.6.0_23"

   $ . ./setEnv.sh

   $ ant
   Buildfile: C:\Myjython\jython\build.xml
   init:
   needed-check:
   clean-if-antlr-needed:
   init:
   clean-checkout-dir:
   clean:
   prepare-output:
      [mkdir] Created dir: C:\Myjython\jython\build\classes
      [mkdir] Created dir: C:\Myjython\jython\build\gensrc\org\python\antlr
      [mkdir] Created dir: C:\Myjython\jython\build\exposed
   antlr_gen:
3. $ Try it out

$ dist/bin/jython
Jython 2.6a0+ (trunk, Aug 2 2011, 19:41:57)
[Java HotSpot(TM) Client VM (Sun Microsystems Inc.)] on java1.6.0_23
Type "help", "copyright", "credits" or "license" for more information.

>>> 1+2
3
>>> (lambda x: x)(4)
4
>>> (lambda x: x+10)(4)
14
>>>