CSE 341 - Programming Languages - Autumn 2012

We're using SWI Prolog, version 6.0.2.

When you start Prolog, you should see an introductory greeting and then the Prolog command prompt:

```
Welcome to SWI-Prolog .....
?-
```

Running Prolog on attu

If you're planning on loading a file, it will simplify things to first navigate to the directory that contains your file, so that Prolog can find it easily.

In any case run Prolog from the command line using

```
swipl
```

Running Prolog on the CSE Undergrad Windows Machines

Run Prolog from the "Start" menu. It's here:
`All Programs / DEV TOOLS & LANGUAGES / SWI-Prolog`

If you want to load a file, you'll either need to specify an absolute file path, or change the current working directory to the appropriate place. Here's how to change the current working directory. If your files are in your home directory on the Z drive in the `cse341` subdirectory, type this at the Prolog command prompt:

```
working_directory(X, 'Z:\\cse341').
```

(The double backslash is needed because a single backslash indicates some special character, e.g. `\n` for newline. And in case you're wondering, X will be unified with the previous working directory.)

Double-clicking on a `.pl` Prolog source file is going to start perl, so that's not going to work.

Running Prolog on a Personal Machine

SWI Prolog is available for Windows, Mac, and Linux. Here's the download page: http://www.swi-prolog.org/Download.html.

Basic Prolog Commands

A few essential commands:

`halt.`
: exit Prolog (short form: control-d)

`consult(filename).`
: load the file named filename.pl (note the added .pl extension). If you need some more complex name (for example with a path), put it in single quotes, for example
`consult('/Users/schmertzkopf/squid.pl').`

`[filename].`
: shorthand for consult

`help(topic) or apropos(topic).`
: Brings up the relevant section of the manual in a separate window (so requires X for linux/Mac).

Editing Prolog Code

Prepare source files using any convenient editor. The standard file extension for Prolog files is `.pl`

If you want to use emacs, include this in your `.emacs` file in your home directory:

```
(setq auto-mode-alist
  (cons (cons "\\.pl" 'prolog-mode)
     auto-mode-alist))
```

(Without this, emacs is going to think you're writing in perl.)

There are other versions of emacs Prolog mode if you want to try one - see e.g. Using SWI-Prolog with GNU-Emacs. However, the default mode should be fine for 341.

Debugging

SWI Prolog provides two interfaces to the debugger: a text-based and a graphical interface:

- Overview of the debugger
- The graphical debugger

The graphical debugger doesn't seem to be installed with the version of Prolog on the Lab linux machines, unfortunately. It is on the Lab Windows machines, though.

For old-school debugging use the write goal. For example, here it is in the body of a rule:

```
notworkingyet(A) :-
   complicated(A),
   write('the result after calling complicated is: \n'), write(A),
   yetmorecomplicated(A).
```