Other Project Resources

A list of other resources that you might find helpful:

- **C++ and UNIX**:
- Thomas Anderson's Quick Introduction to C++ (https://lagunita.stanford.edu/assets/courseware/v1/ae19277a04d1d9436b10e56ca5893d8c/c4x/Engineering/Compilers/asset/c_.ps) may be useful if you know C but are not familiar with all of the C++ subset we use in the programming projects.
- A more thorough guide to using the version of Gnu C++ installed on our Linux machines is here (http://gcc.gnu.org/onlinedocs/gcc-4.1.2/gcc/).
- Quick guide to UNIX development tools (http://cslibrary.stanford.edu/107) (one of the many useful documents available in Nick Parlante's ever-growing CS library (http://cslibrary.stanford.edu/))

- **Help with the gdb debugger:**
- GNU's online gdb users guide (http://sourceware.org/gdb/current/onlinedocs/gdb/)
- Printable quick reference: gdbref.ps (https://lagunita.stanford.edu/assets/courseware/v1/639a0f0b33370bc28a6c8c6fc9abd9c7/c4x/Engineering/Compilers/asset/gdbref.ps)
- A GDB article that Julie Zelenski wrote a few years ago for a programming journal: GDB breakpoint tricks (http://web.stanford.edu/class/cs107/gdb_coredump1.pdf)

- **References on lex & yacc:**

`Lex` is the original lexical scanner developed by Lesk and Schmidt; Paxson's improved version is `flex`. Similarly, `yacc` is Johnson and Sethi's original parser; `bison` is the GNU-equivalent written by Corbett and Stallman. Both are designed to be upward-compatible with the original while adding extensions and improvements.

- Original documentation by the authors of the tools themselves. These papers are quite readable and serve as an excellent introduction for familiarizing yourself with the tools.
    - Lesk and Schmidt on lex (http://dinosaur.compilertools.net/lex/index.html)

    - Johnson on yacc (http://dinosaur.compilertools.net/#yacc)
- Man pages are available from command line, e.g., man lex. We've also put up browsable versions of the Solaris man pages for lex (http://www.stanford.edu/class/archive/cs/cs143/cs143.1112/materials/other/manlex.html), flex (http://www.stanford.edu/class/archive/cs/cs143/cs143.1112/materials/other/manflex.html), yacc (http://www.stanford.edu/class/archive/cs/cs143/cs143.1112/materials/other/manyacc.html), and bison (http://www.stanford.edu/class/archive/cs/cs143/cs143.1112/materials/other/manbison.html).
- GNU's online documentation (full manuals, long, but very complete)
    - flex (http://flex.sourceforge.net/manual/)
    - bison (http://www.gnu.org/software/bison/manual/bison.html)
- The lex & yacc page (http://www.combo.org/lex_yacc_page/) from Combo.org.
- An article (http://www2.linuxjournal.com/lj-issues/issue51/2227.html) from the Linux Journal singing the praises of lex & yacc.

- **References on JLex and Java_cup:**
- Manual for JLex (http://www.cs.princeton.edu/~appel/modern/java/JLex/current/manual.html), Java CUP (http://www2.cs.tum.edu/projects/cup/manual.html).
- JLex (http://www.cs.princeton.edu/~appel/modern/java/JLex/).
- Java CUP (http://www.stanford.edu/class/archive/cs/cs143/cs143.1112/javadoc/java_cup/).
- Cool Trees (http://www.stanford.edu/class/archive/cs/cs143/cs143.1112/javadoc/cool_ast/) (Start with class TreeNode).

- **References on MIPS & SPIM:**
- A PDF version of the SPIM Manual (http://www.stanford.edu/class/archive/cs/cs143/cs143.1112/materials/other/SPIM_Manual.pdf) (appendix from Hennessy & Patterson's architecture book)
- The SPIM home page (http://www.cs.wisc.edu/~larus/spim.html) (downloadable versions, more docs).

Just for fun:

- A list of funny error messages (http://www.netfunny.com/rhf/jokes/91q3/cerrors.html) from the old MPW C compiler.
- A translator for Latin to Perl (http://www.csse.monash.edu.au/~damian/papers/HTML/Perligata.html) (and you thought there was no practical use for what you learned in 143!)