

Lab A: Reading Material

Lab A introduces basic programming in C for Unix. The reading material covers basic C programming, using the `make(1)` utility to compile and run programs, and the `Unix man(1)` facility — the traditional on-line Unix manual on command-line utilities, system calls, and C libraries.

Programming in C

There are many good books on the C programming language, including

- [C: A Reference Manual](#)
- The C Programming Language

and others. For the lab assignment, the students should read an introductory section of any available book on C programming; in particular, we recommend chapter: [Beginning C](#) from the freely available wikibook [C Programming](#). We also recommend to read about [ASCII Table and Description](#).

Using make and writing makefiles

There are several different versions of `make(1)` available, but the modern *de facto* standard is GNU Make. GNU Make is a free software included with Linux, and its manual is available online. Students should read and understand the [introductory section](#) of the manual, sections 2, 2.1, 2.2, 2.3, 4.1. The description of makefile syntax and make capabilities provided in the introduction is sufficient for the System Programming Laboratory course. You can read more and see an example in "Makefile examples" file.

Unix man(1)

`man` (short for Manual) is the traditional Unix browser for manual pages. Unix programmers and users consult `man` to learn about properties of unix commands, arguments and return values of library functions, configuration file formats etc. Two `man` pages serve as introduction to `man`: "`man intro`", which describes organization of contents of the Unix manual, and "`man man`", which explains how to use the `man` command. Please read and understand the `man` pages and try to use `man`. We will use it frequently during the lab sessions.

As part of your preparation for labs 1 and A, you should also read about the C standard library functions: `strncmp(3)`, `fprintf(3)`, `fgetc(3)`, `fputc(3)`, `fopen(3)`, `feof(3)`, `fclose(3)`.