CSCI 1120 (Low-Level Computing), Spring 2011

Syllabus

1 Course description

Our traditional first course for computer science majors, CSCI 1320 (Principles of Algorithm Design I), is designed to not only teach basic programming and problem solving but also to expose students to certain concepts of computing closely related to the machine itself. Our alternative first course, CSCI 1311 (Introduction to Programming Logic) also teaches basic programming and problem solving, but in a way that does not provide the same exposure to low-level concepts. CSCI 1120 is intended to give students who start with CSCI 1311 the same exposure to low-level concepts provided by CSCI 1320.

Course goals

- Basic knowledge of the C programming language and Linux/UNIX command-line development tools.
- Basic understanding of machine arithmetic.

Course topics

- The Linux/UNIX command-line environment and tools relevant to program development.
- Basics of C programming, with a focus on how it differs from programming in higher-level languages such as Java.
- Basics of computer arithmetic.
- More advanced topics as time permits (e.g., multithreaded programming with OpenMP, full-screen text-based programming with the `ncurses` library, etc.).

2 Basic information

Class meeting times and location

- M 1:30pm – 2:20pm, HAS 228

Prerequisites

- CSCI 1311 or consent of instructor

Instructor contact information

- Dr. Berna Massingill
- Office: HAS 201L
- Office phone: (210) 999-8138
- E-mail: `bmassing@cs.trinity.edu`

Office hours

Scheduled office hours for this semester are as follows:

- Monday 12:30pm – 1:30pm
- Tuesday noon – 12:30pm, 2pm – 3:30pm
- Wednesday noon – 4pm
- Thursday noon – 12:30pm, 2pm – 3:30pm

These times are subject to change; a current schedule will be available on my Web page.

If I'm not in my office, I should be somewhere in the building (perhaps in one of the labs helping another student), and there will often be a note on my door saying where to find me.

Some office hours will be held in one of the classrooms/labs (times to be announced soon). These are "open lab" times, during which I'll be in one of the department's labs, prepared to answer questions. The intent is that students can use these times to work on assignments with someone available to help with any questions or problems.

In addition to scheduled office hours, you're welcome to drop by and see if I'm in my office and free to talk, or you can make an appointment by calling me or sending me e-mail.

E-mail is almost always a good way to reach me; I normally check it fairly often and reply promptly.

3 Course materials

Textbook

There is no required textbook for this course. The course Web page will have links to some useful on-line reading, which you will be expected to read/skim/consult. I recommend that you also obtain a copy of one of the books on C listed in "Other references" below.

Web page

Most course-related information (this syllabus, homework and reading assignments, etc.) will be made available via the Web. The course Web page is a starting point for Web-accessible course material; you can find it linked from my home page (`http://www.cs.trinity.edu/~bmassing`) or directly at `http://www.cs.trinity.edu/~bmassing/Classes/CS1120_2011spring/HTML`.

Other references

There are many books on the C language and UNIX, far too many to list. Here are a few that may be of interest.

- Behrouz A. Forouzan and Richard F. Gilberg. *Computer Science: A Structured Programming Approach Using C.* Thomson Course Technology, third edition, 2005. A book often used for CSCI 1320.
- Samuel P. Harbison and Guy L. Steele. *C: A Reference Manual.* Prentice Hall, fifth edition, 2002. A good reference manual.
- Brian W. Kernighan and Dennis M. Ritchie. *The C Programming Language.* Prentice Hall PTR, second edition, 1988. The classic reference on the language.
- K. N. King. *C Programming: A Modern Approach.* W. W. Norton & Company, second edition, 2008. A widely-recommended introduction to programming using C.
- Jon Lasser. *Think UNIX.* QUE, 2000. An interesting and short introductory book on UNIX.

- Peter Prinz and Tony Crawford. *C in a Nutshell.* O'Reilly, 2006. A good though compact reference manual also including material on some related tools.

4 Course requirements

Grading

Grades in this course will be determined by the results of several homework assignmentsand class participation, weighted as follows.

Component	**Maximum points**
Homework	about 150
Class participation	20

Numeric grades will be calculated as a simple percentage, by dividing total points earned on the above components by total points possible. These numeric grades will then be converted to letter grades based on a curve, but in no case will the resulting letter grades be worse than students would receive based on the following scheme.

Numeric grade	**Letter grade**
90 – 100	A-/A
80 – 89	B-/B/B+
70 – 79	C-/C/C+
60 – 69	D/D+
0 – 59	F

Homework assignments

Homework (in the form of programming assignments) is a crucial part of this course; much of what you learn will likely be learned in the course of completing the programming assignments. Detailed requirements will be provided as part of each assignment; due dates will be announced via the course Web page. You are strongly encouraged to use the department's network of Linux machines, but unless otherwise specified for individual assignments, you may use any other system that provides a suitable environment.

Attendance

Regular class attendance is strongly encouraged; class participation grades will be based largely on attendance.

E-mail

Course-related announcements will sometimes be made by sending e-mail to the Trinity e-mail addresses of all registered students. Students are strongly encouraged to read mail sent to their Trinity addresses frequently.

Late and missed work

Unless otherwise stated for a particular assignment, homework will be accepted up to one class period late, *but no more*, at a penalty of 10 percent off per working day. This penalty may be waived or additional time allowed *at the instructor's discretion* in cases of illness or conflict with a university-sponsored activity or religious holiday.

If you have unusual circumstances (as we all sometimes do), please discuss these with me as far in advance as possible.

Academic integrity at Trinity

All students are covered by the Trinity University Honor Code, which prohibits dishonesty in academic work.

The Code asserts that the academic community is based on honesty and trust, and defines specific violations as well as the procedure to determine if a violation has occurred. The Code also covers the process of hearings for alleged violations and the various sanctions applied for specific violations. The Code also provides for an appeal process.

The Code is implemented by the Academic Honor Council. Under the Code, a faculty member will (or a student may) report an alleged violation to the Academic Honor Council. It is the task of the Council to collect the pertinent evidence, adjudicate, and assign a sanction within certain guidelines if a violation has been verified.

Students who are under the Honor Code are required to pledge all written work that is submitted for a grade: "On my honor, I have neither given nor received any unauthorized assistance on this work" and their signature. The pledge may be abbreviated "pledged" with a signature.

The specifics of the Honor Code, its underlying philosophy, and the norms for sanctioning can all be found on the Academic Honor Council website, accessed through the Trinity Homepage, or directly here[1].

Collaboration and academic integrity in this course

Unless otherwise specified, all work submitted for a grade (homework assignments) must represent the student's own individual effort. Unless otherwise stated, all submitted work will be considered pledged work.

Discussion of homework assignments among students is encouraged, but not to the point where detailed answers are being written collectively. Graded papers and sample solutions from previous years are off limits. Answers that are identical beyond coincidence (either to another student's work or to a sample solution from a previous year) will be considered to be in violation of the Honor Code, and *will result in appropriate action.* You are responsible for the security of your work, both electronic and hard copy.

[1] `http://www.trinity.edu/departments/academic_affairs/honor_code/`