CSci 493.73 Programming Graphical User Interfaces
Essentials: Course Communications, Content, and Structure

Communications

Class Meetings:	Monday, Thursday 9:45 A.M. - 11:00 A.M., HW207
Office:	HN 1090J
Office Hours:	Monday 11:30 A.M. - 12:30 P.M., Friday 12:45 - 1:45 P.M.
Email:	stewart.weiss@hunter.cuny.edu
Telephone:	(212) 772-5469 or (212) 772-5213 (department office)

Resources

Textbooks: Andrew Krause, *Foundations of GTK+ Development, Apress Publishers. 2007.* ISBN-13: 978-1590597934

Computing Facilities: Registered students will be given user accounts on the UNIX hosts in the 1000G lab of the Computer Science Department, located on the tenth floor of Hunter North. This lab is open 24 hours a day, 7 days a week and access to it is limited to students enrolled in selected courses. In addition, students will be able to use a secure remote login service such as *ssh* to access these accounts.

Website: All course materials, including lecture notes, slides, assignments, syllabi, and other resources, including this document, are posted on my website, at http://www.compsci.hunter.cuny.edu/~sweiss/course_materials/csci493.73/csci493.73_spr10.php

Prerequisites

Enrolled students must have successfully completed CSci 235 and CSci 340.

Learning Goals

Material in this course supports the following departmental learning goals: 1d: (having in-depth knowledge of at least two areas of specialization: e.g. databases, networking, computer theory, graphics, etc.) by learning how the interfaces to major operating systems work; 2c (being able to apply principles of design and analysis in creating substantial programs and have experience working in teams on projects of moderately realistic scope.) by requiring that a large practical project be completed with feedback from the rest of the class; 3a: (ability to communicate ideas effectively) by requiring homework that is graded in part on clarity and proper use of the English language, and also by requiring an oral presentation.

Course Content

The advertised title of this course is a mistake. The correct title is Programming Graphical User Interfaces. This course is not about programming for the Windows® operating system., but it about how to create programs that use windows and graphical user interface elements. You will learn about a free, public-domain, cross-platform library of graphical user interface tools that can be used with UNIX and Windows operating systems alike.

The course will cover a bit of UNIX, software development in a command-line interface, including the use of make-files and libraries, concepts about window-based programs, events, signals, signal-handlers and callback functions, and GTK. You should be comfortable working in a UNIX environment if you plan to take this class. Elementary UNIX skills will not be covered.

Expectations, Tests, Assignments, and Grading

There will be several assigned programming projects as well as readings. You are expected to do all of the specified reading, complete all projects *on time*, and work independently. This is a seminar, not a regular lecture. As such, you are expected to be self-motivated and self-disciplined. The final grade in the class will be based entirely on the grades on the projects. The final project will require a 15 minute oral presentation to the class.

Course Materials, the Web, and Blackboard

As noted above, all lecture notes will be posted on my website, which, unlike *Blackboard,* does not require privilege to access. I rely on *Blackboard* only for communicating to all students, for posting grades and for the use of the *Discussion Board*, which is enabled so that students can have a free exchange of ideas. Therefore, you should check *Blackboard* before each class in case there are announcements.

I require that you use the following protocol if you have a question:

1. Check whether the question you want to ask has been asked and answered in the *Discussion Board*.
2. If it has been answered, you are finished. If not, post the question in the *Discussion Board* and ...
3. Send an email message to me asking me to look at the question on the *Discussion Board* and answer it there.
4. I will answer the question and send you an email message when I have answered it, so that you do not have to "poll" it waiting for an answer.

If you do not post your question, I will ignore it. I do this to save time for all of us.

Programming and System Access

Every student is given an account on the Computer Science Department's UNIX network, and has two different methods of accessing that account.

The first is to use the 1000G lab, which has workstations that run Red Hat Enterprise Linux 5, one version of UNIX. This lab is open “24/7” and has 24 workstations. The advantage of this is that you will be sitting in front of the monitor of the Linux host and will not be subject to potential disconnections that can take place when working remotely, nor the slowness of the network. The disadvantage is that it requires you to be in school.

The other choice is to work ***remotely***. The Computer Science Department has a UNIX host, ***eniac.geo.hunter.cuny.edu***, available to students who have access to the lab. You will be able to access this host from any computer that has ***ssh*** client software. If you download the ***ssh*** client software to your home machine, you will be able to login from home. Once you login to eniac, you should immediately ssh from ***eniac*** to one of the many hosts in the 1000G lab, which have names of the form ***cslab1***, ***cslab2***, ..., ***cslab25***. These machines are underutilized and will probably respond faster than eniac, which is primarily a gateway.

There are several versions of ***ssh***. ***OpenSSH*** is an open source version developed for the OpenBSD project. It is available for many operating systems. The OpenSSH home page is at

http://www.openssh.com.

Alternatives for Windows are at http://www.openssh.com/windows.html and those for Mac are at http://www.openssh.com/macos.html.

PuTTY ssh is another free version for *Windows* operating systems, available at http://www.chiark.greenend.org.uk/~sgtatham/putty/download.html. My preference for Windows is the original SSH client, which is no longer supported. It can be downloaded from my website at http://www.compsci.hunter.cuny.edu/~sweiss/resources.php#Applications.

About C and C++ in This Course

Most of the programming examples are written in C. Some students have a knee-jerk reaction when they hear this, saying, "but I don't know C." This is not true. The C++ language contains most of the C language. If you know C++, you know a great deal of C. There are minor differences that arise in the syntax of declarations (such as struct and function declarations), but the real problems are that C relies heavily on pointers since it does not have reference variables, and most students never learn how to use the C standard I/O library. Most students learn stream I/O and never bother to learn the seemingly archaic functions of the C standard I/O library. In general, you ought to know C, since its use is widespread. While you are free to write C++ code, you will be required to read C code.

Academic Honesty Policy

Hunter College regards acts of academic dishonesty (e.g., plagiarism, cheating on examinations, obtaining unfair advantage, and falsification of records and official documents) as serious offenses against the values of intellectual honesty. The college is committed to enforcing the CUNY Policy on Academic Integrity and will pursue cases of academic dishonesty according to the Hunter College Academic Integrity Procedures. In this class, I will enforce the University's Policy on Academic Integrity and bring any violations that I discover to the attention of the Dean of Students' Office.