CS 215 – Introduction to Program Design, Abstraction, and Problem Solving

Fall 2012

This course will introduce the student to object-oriented design and problem solving. Subjects include data structures, dynamic data allocation, pointers, recursion, and debugging. The fundamental goal of the course is to introduce the student to the C++ programming language and to provide the student with a toolkit for solving problems with an object-oriented programming language.

Course web page: http://protocols.netlab.uky.edu/~davidb/cs215

Meeting Times and Places

<table>
<tr><th>Sec.</th><th>Teaching Assistant</th><th>Lecturer</th><th>Lecture Time/Place</th><th>Lab Time</th><th>Lab Place</th></tr>
<tr><td>001</td><td rowspan="2">Xiwei (Jeffrey) Wang</td><td rowspan="5">David Bingham Brown</td><td rowspan="5">TR 9:30-10:45 AM
CB 122</td><td>W 10:20-11:50 AM</td><td rowspan="6">RGAN 103</td></tr>
<tr><td>002</td><td>W 12:00-1:30 PM</td></tr>
<tr><td>003</td><td rowspan="2">Pengpeng (Mark) Lin</td><td>W 2:00-3:30 PM</td></tr>
<tr><td>004</td><td>W 4:00-5:30 PM</td></tr>
<tr><td>005</td><td>Xudong Liu</td><td>W 8:15-9:45 AM</td></tr>
<tr><td>401</td><td colspan="2">Frank Richardson</td><td>R 6:00-8:00 PM
FPAT 257</td><td>T 6:00-8:00 PM</td></tr>
</table>

Course Staff

Staff member	Office	Office Hours	Email
David Bingham Brown	228 Hardymon	TR 2-4 PM	david.b.brown@uky.edu
Frank Richardson	Ingels Hall	TR 4-6 PM	frank.richardson@uky.edu
Xiwei (Jeffrey) Wang	Multilab (Annex 203)	W 2-5 PM	j.wang@uky.edu
Pengpeng (Mark) Lin	Multilab (Annex 203)	TBA	pli222@uky.edu
Xudong Liu	Multilab (Annex 203)	TBA	xli243@uky.edu

Course Materials

While additional course materials (in the form of assignments, problem sets, and additional readings) will be provided online through the course web page, you are responsible for obtaining a copy of the course's textbook. Students in the daytime sections will also need to obtain a "clicker" for use in the course. The course web page, source control system, and email to students will all be used to disseminate required information for the course.

The **required text** for the course is *C++ For Everyone* (Second Edition), by Cay Horstmann (Wiley, ISBN 97-0-470-92713-7).

"Clickers" (again, required only for the daytime sections) are, formally called TurningPoint "ResponseCard" RF clickers and are available at the bookstore; you will need to ask at the counter for one if you need one. These devices are campus standard and required for a number of classes, so if you already have one it should work for this class as well as your other classes.

Grading, Assignment, and Submission Policies

Your final grade in the class will be determined using the following weights and assigned a letter grade based on this scale:

Programming Assignments	**30%**	**90-100%**	**A**
Weekly Assignments (Labs, quizzes, problem sets)	**30%**	**80-89%**	**B**
Attendance	**10%**	**70-79%**	**C**
Midterm Exam	**15%**	**60-69%**	**D**
Final Exam	**15%**	**<60%**	**E**

Programming Assignments

There will be four programming assignments spaced throughout the semester. Each programming assignment will be announced in class with instructions and grading notes provided through the class web site. Together, these four assignments will compromise 30% of your final grade.

Weekly Assignments

During most weeks, a problem set will be assigned on Tuesday*, available on the class web site, and will be due Friday night at midnight. A lab exercise will be assigned on the lab day, covered during the lab session, and due that night at midnight. Quizzes may as well be given during the lab session. Altogether, these assignments will compromise 30% of your final grade.

Attendance

Starting with the third week (on Tuesday, September 4th*), attendance will be taken in class through use of each student's "clicker". This will take the form of (typically) one or two questions asked in class; to get credit for attendance, you must only *answer* the questions, not answer them correctly. Attendance will compromise 10% of your final grade. Answering at least 80% of the attendance questions correctly will give you 5% extra credit to your final grade.

Midterm and Final Exams

The midterm (Thursday, October 11th*, normal class time and location) and final (Tuesday, December 11th*, normal class location, but 8:00 AM) will each be 15% of your final grade. Note that the time of the final exam is a function of the university's final exam schedule, and we have no influence over it. Make-up exams will not be provided without an excused absence.

Submission

All assignments (whether weekly problem sets, lab exercises, or programming assignments) must be submitted electronically via the class's source control system (or CS department portal for the evening section). Use of this system will be explained during the first meeting of the lab session on Wednesday, August 29th*. **Late assignments (of any type) will not be accepted without an excused absence.**

Grades for all assignments will typically be posted (and available for discussion with your TA) during the next lab session.

* Note that dates and schedules will vary somewhat for the evening section of CS 215

Academic Dishonesty

All assignments in class are individual work. All work submitted as part of the class must be your own. You may not share work (whether from weekly problem sets, lab exercises, or programming assignments). You may not "click" in for another student or have another student "click" in for your attendance grade. Academic offenses will be taken seriously, and the *minimum* penalty for academic dishonesty will be a score of zero assigned to the affected assignment and a reduction by one letter of the final grade in the course. Note that for this purpose, the entire attendance section of your final grade is considered one assignment.

Computer Facilities

Lab sessions will be held in the computer lab in RGAN 103. All lab exercises and programming assignments will be compatible with the virtual desktops available to students at http://apps.uky.edu. The course will be taught (and assignments designed with) the Visual Studio 2010 IDE; this IDE will be available both from the virtual desktop above and through the MSDNAA program through the university to students of computer science classes. The first lab session will be focused on familiarizing students with these resources.

In-Class Engagement

The use of electronic devices (other than the "clicker" required for class and the workstations used during lab sessions) is not permitted during class sessions. Students are expected to be engaged while in class and not providing distractions for other students present.

Accomodations

If you have a documented disability that requires accommodation, please contact the instructor (either David Brown for the daytime section or Frank Richardson for the evening session) as soon as possible at the beginning of the course. In order to receive accommodation, you must provide a letter from the Disability Resource Center on campus.

Educational Objectives

This class is designed to provide the student with a toolkit for solving problems with an object oriented programming language. This toolkit includes (but is not limited to!): the basics of the C++ language, data types, data structures, algorithm design, recursion, and debugging.

Proficiency Goals

- Object oriented approach to programming
- Dynamic memory management
- Recursive programming techniques
- Data structures, such as linked lists, stacks, queues, and trees

Familiarity Goals:

- Sorting techniques
- Searching techniques, such as binary search trees
- Algorithm run-time analysis (Big-O notation)
- User interfaces and event-driven programs

Course Evaluation Questions

These questions will appear on the final course evaluation. The goal of the course is to provide a directed and comprehensive learning experience that will enable you to offer the highest rating on each of the questions. In order for that to happen, you will need to work hard to do your part – attend lectures, study the material, do the readings from the course text, complete individual exercises, programs, and labs.

37. I have confidence in my ability to solve programming problems using classes.
38. I have confidence in my ability to solve programming problems using dynamic data and pointers.
39. I have confidence in my ability to solve programming problems using recursion.
40. I understand the basic data structures (linked lists, stacks, queues, trees) and can use them in programs.
41. I understand the principles of sorting and searching.

Schedule of Lectures

Note: This schedule is subject to change; the evening section will have different lecture dates.

Week	Date	Reading	Topic
1	R 8/23	N/A	Course Introduction
2	T 8/28	2,3,4	"Hello World", data types, what is a program?
	R 8/30		Decisions and loops
3	T 9/4	5	Debugging and scope
	R 9/6		Functions and parameters
4	T 9/11	6	Arrays and vectors
	R 9/13		Multi-dimensional arrays and vectors
5	T 9/18	7	Pointers
	R 9/20		Dynamic memory (and more pointers)
6	T 9/25	8	Streams
	R 9/27		More streams
7	T 10/2	9	Classes and object oriented programming
	R 10/4		Pointers to objects
8	T 10/9	N/A	Midterm Review
	R 10/11		**Midterm – 9:30 AM, CB 122**
9	T 10/16	10	Inheritance
	R 10/18		Polymorphism
10	T 10/23	TBD	Recursion
	R 10/25		More recursion
11	T 10/30	11	Complexity analysis
	R 11/1		Debugging techniques
12	**T 11/6**	**No classes – Election Day**	
	R 11/8	12	Sorting and searching
13	T 11/13	13	Lists, stacks, and queues
	R 11/15		Sets, maps, and priority queues
14	T 11/20	14	TBD
	R 11/22	**No classes – Thanksgiving Holiday**	
15	T 11/27	TBD	Event-driven programming
	R 11/29		Graphical user interfaces
16	T 12/4	N/A	Special Topics/Review
	R 12/6		Final Review
Final	**T 12/11**	**Final – 8:00 AM, CB 122**	