Lecture: Monday, Wednesday & Friday, 9:25-10:15, Nieuwland Science Hall 118

Labs (CSE 21212): You are required to attend the lab section you are registered for.
• Section 1: Monday 3:00-3:50 pm in Cushing 303
• Section 2: Monday 4:05-4:55 pm in Cushing 303
• Section 3: Tuesday 5:05-5:55 pm in Cushing 303
• Section 4: Wednesday 3:00-3:50 pm in Cushing 303
• Section 5: Wednesday 4:05-4:55 pm in Cushing 303

An Invitation

It is my goal for you to learn C++ and enjoy this course along the way! I would like to balance us having a positive experience with developing real, applied skills that will be highly useful to you in interviews, this summer, next fall in Data Structures and beyond. Remember the TAs and I are here at every step to help you with material outside of class, working on a large project as a member of a team, and to continue improve as a programmer through object oriented programming (OOP) concepts. With this in mind we will have both active learning experiences and in-class demos of final projects. Along the way I will try to attend each lab briefly to answer questions one-on-one if you have them.

Team Fund II:

• Dr. Scott Emrich
  o Office: 211B Cushing Hall; 631-0353 Cell: (574) 485-6837
  o Electronic contact: semrich@nd.edu; IM/Skype: semrichnd; Google: scott.emrich
  o Tentative office hours: See Doodle poll email at start of semester; by appointment
  o If my office door is open, you are welcome to come in and ask questions; office visits, however, are best arranged via email.

<table>
<thead>
<tr>
<th>TAs</th>
<th>Email</th>
<th>Office hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>Buchanan Bourdon</td>
<td><a href="mailto:bbourdon@nd.edu">bbourdon@nd.edu</a></td>
<td>See Piazza</td>
</tr>
<tr>
<td>Ryan Tick</td>
<td><a href="mailto:rtick@nd.edu">rtick@nd.edu</a></td>
<td>See Piazza</td>
</tr>
<tr>
<td>Ryan Moran</td>
<td><a href="mailto:rmoran@nd.edu">rmoran@nd.edu</a></td>
<td>See Piazza</td>
</tr>
</tbody>
</table>

Our course webpage: http://www3.nd.edu/~cse/2016sp/20212/

Our textbook: Deitel, C++ How to Program, Pearson/Prentice-Hall; 7th or later edition is fine

Course Outcomes: To successfully complete the course, students must be able to:
1. Specify, design, implement and test programs of moderate size in C++. Assessed in programming assignments, and the semester project.
2. Be able to design classes competently, including data hiding, operator overloading, polymorphism and exception handling features. Assessed in programming assignments.
3. Demonstrate understanding of, and proficiency in use of C++/object-oriented concepts including inheritance, templates and generics especially for implementing these simple data structures: vector, list, array and trees. Assessed via exam and programming assignments.
4. Demonstrate the ability to understand and use design patterns. Assessed in exams and on programming assignments.
5. Demonstrate understanding of, and the ability to use basic elements of the STL. Assessed in exams and in programming assignments.

Major Topics:

1. Review of C++ basics (1 hour)
2. Classes (3 hours)
3. Object-oriented programming (overloading, inheritance, polymorphism) (5 hours)
4. Templates/basic STL (5 hours)
5. Exception handling (1 hour)
6. Basic graphics/game design (3 hours)
7. Software Engineering: design patterns, UML, case studies (5 hours)
8. Data structures: vector, list, simple tree, dictionary (4-5 hours)
9. Associative containers: maps and sets (2 hours)
10. Special topics (5 hours)
11. In-class examples of object-oriented programming: OOP week! (4 hours)
12. Exam (1 hour)

Grading: Your letter grade will be computed from a course percentage that is a weighted sum of the fraction of points received in five groups of graded work plus participation:

40%: Labs (including pre-lab)
25%: Final Project
15%: midterm exam (Monday, February 22, in class)
15%: final exam (TBA)
5%: class participation

Late labs will not be accepted. However, we will accept one assignment for re-grading without penalty (amnesty) at the end of the semester. Dates will be announced on the course website.

Course percentage will be translated into letter grades as follows: A: 94% and up; A-: 91-94%; B+: 88-91%; B: 84-88%; B-: 81-84%; C+: 78-81%; C: 74-78%; C-: 71-74%; D: 64-70%; F: 0-64%. Requests for regrading of any item must be communicated, in writing (including e-mail), to Prof. Emrich (not a TA), within one week of the day the items were returned to you. Note that failure to participate in the final project results in a maximum grade of C (75%) and failure to complete labs results in a maximum grade of F (60%).

Absences from lab or lecture are only excused in accordance with University policy (see du Lac).

Project: A programming project will be due at the end of the semester. Groups must consist of two to four students (solo projects won’t be allowed). Two weeks of lab will be devoted to project support. The last set of projects will be demonstrated at an end-of-semester pizza party!
ADA statement: Any student who would need an accommodation based on a disability on file with the University can contact Prof. Emrich privately. Full accommodation will be provided.

Academic Code of Honor: Entering Notre Dame you were required to study the on-line edition of the Academic Code of Honor, to pass a quiz on it, and to sign a pledge to abide by it. The full Code and a Student Guide to the Academic code of Honor are available at: http://www.nd.edu/~hnrcode/docs/handbook.htm. Perhaps the most fundamental sentence is the beginning of section IV-B:

The pledge to uphold the Academic Code of Honor includes an understanding that a student’s submitted work, graded or ungraded – examinations, draft copies, papers, homework assignments, extra credit work, etc. - must be his or her own.

You are encouraged to study and discuss course material with classmates, but assessed material must derive from your head to your fingers to your text editor. In other words, no “copy and pasting” of code from group or other settings. Because any instance of academic dishonesty will be reported, please see Prof. Emrich if you are ever in doubt on what is allowed.

Attendance and Time Management

You will notice that much of Fundamentals II will be structured like Fundamentals I. In general, topics will be introduced on Monday before labs, followed up on Wed, and practical applications will be discussed on Friday once everyone has had lab.

No matter how you look at it programming is a time-intensive activity that is best done throughout the week (and not the night before it is due). To be fair to all, labs will be due 30 minutes before the next lab session. For example, if a student has lab on Wed, 2/5 that ends at 4:50pm, the assignment will be due 2/12 at 3:30pm (roughly one week after lab). As in the fall, no labs will be accepted late! Please submit partial work into your drop box as instructed in the lab submission guideline and we will provide partial credit along with constructive feedback.

Class participation points will be earned similar to Fundamentals I as follows:

0.5 points – Show up at your scheduled lab on time as TAs will take attendance (once per week)
1 point – Complete a substantial portion of the first part of lab, show it to a TA. (once per week.)
1 point – Stop by Prof. Emrich’s office hours during the semester. (max one point)
0.5 points – Participate in a classroom exercise during lecture. (max seven points.)

Your class participation grade will be calculated as (points earned / 20), so you can earn a few extra credit points by participating in lectures throughout the semester.

Because the instructor realizes that sometimes the most recent version may not have been submitted, concepts can be mastered later, or things just happen, the instructor will accept one lab for regrading without penalty (lab amnesty) at the end of the semester.

Coder Dollars / Bonuses:

For the past few years, the instructor piloted offering an optional problem for extra credit. These assignments had little effect on the final grades but provided another avenue to express creativity.
Rather than give explicit extra credit points, however, we will give out what we will call “Coder Dollars” that can be exchanged for extra credit points (min 2 per student who participates at least twice this semester; max of 200 for all students combined). The co-instructor will be our project manager, will meet with you as needed in lab, and will determine coder dollars for all contributions.