Programming and Problem Solving in C++ (CS 333) Syllabus

Required Textbook:


(Along with CDs not required)

Additional Resources (optional):

Additional materials are listed for those interested in more advanced approaches to common problems; however, the textbook is the main informational source for this course. The instructor's lecture notes cover several topics not included in this textbook. These will answer specific questions regarding implementation and design patterns:

- **More Effective C++** by Scott Meyers (ISBN: 020163371X) – Addison-Wesley Professional
- **Effective STL** by Scott Meyers (ISBN: 0201749629) – Addison-Wesley Professional
- **Effective Modern C++** by Scott Meyers (ISBN: 1491903996) – O'Reilly Media
- **C++ Coding Standards** by Andrei Alexandrescu and Herb Sutter (ISBN: 0321113580) – Addison-Wesley Professional
- **Exceptional C++** by Herb Sutter (ISBN: 0201615622) – Addison-Wesley Professional

Co-Requisite:

CS 252 – Intro to UNIX for Programmers

**NOTE:** While CS 252 is a co-requisite, certain portions of CS 252 should be completed in preparation for certain CS 333 labs/assignments. Successful students complete CS 252 during or prior to the completion of Part I of the CS 333 (i.e., the first three weeks of the semester).

Communications:

This online course will not have regularly scheduled lectures. Blackboard announcements will list a limited number of attendance-optional network conferences, which will be recorded for the convenience of those unable to attend. Whereas this offers students scheduling flexibility, it also requires significant maturity and self-discipline to avoid falling behind on coursework. There are several options for course communications:

- **Blackboard Discussions** – since posts are frequently checked, students should make these forums their primary method for general course discussions, discussions regarding website problems, or general questions.
- **Scheduled Net Meetings** – we will cover course material and answer questions via periodic, recorded WebEx sessions.
- **Email** – send specific questions regarding assignments or grade questions directly to the instructor.
- **Office Hours** – office hours will be posted on the instructor’s website (http://www.cs.odu.edu/~jhowland/contact.html).

Course Description and Organization:

This course introduces computer programming and problem solving using the C++ language. Topics include C++ syntax and semantics, principles of design, and basic software engineering skills. This accelerated course covers the material and satisfies the requirements of both CS 150 and CS 250. Students find this course requires more attention than regular courses; therefore, students should already have programming knowledge in either C++ or another high level programming language.

This course is divided into three sections. The first section is roughly equivalent to CS 150, while the latter two sections are equivalent to CS 250. It emphasizes solving problems using the C++ programming language.

- **Part I** – an accelerated review of CS 150 with emphasis on the C++ language
- **Part II** – an introduction to advanced programming topics
- **Part III** – an introduction into design and testing with a continuation of advanced programming topics

Assignments and Grading:

Each part contains several topics. Each topic is addressed in the textbook readings, lecture notes, and a variety of activities, including:

- **Self-Assessments** – ungraded online quizzes
- **Labs** – ungraded activities to practice techniques used in assignments
- **Assignments** – graded activities (most involve programming)
- **UNIX Assignments** (refers back to the co-requisite) – students should have completed CS 252 curriculum up to and including the listed assignment, as they greatly benefit from the skills taught in that course
- **Exams** – administered after each part (first two mostly covering their respective parts with a cumulative final)
- **Project** – a semester term project will apply the techniques of design, implementation, testing, and debugging to a larger problem

<table>
<thead>
<tr>
<th>Activity</th>
<th>Allocation</th>
<th>Range</th>
<th>Grade</th>
</tr>
</thead>
<tbody>
<tr>
<td>Part I Exam</td>
<td>10%</td>
<td>84 – 87</td>
<td>B</td>
</tr>
<tr>
<td>Part II Exam</td>
<td>10%</td>
<td>80 – 84</td>
<td>B-</td>
</tr>
<tr>
<td>Part III Exam</td>
<td>15%</td>
<td>77 – 80</td>
<td>C+</td>
</tr>
<tr>
<td>Term Project</td>
<td>20%</td>
<td>74 – 77</td>
<td>C</td>
</tr>
<tr>
<td>Assignments</td>
<td>45%</td>
<td>70 – 74</td>
<td>C</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td><strong>100%</strong></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Range | Grade
--- | ---
≥ 94 | A
90 – 94 | A-
87 – 90 | B+
84 – 87 | B
80 – 84 | B-
77 – 80 | C+  
74 – 77 | C
70 – 74 | C
67 – 70 | D+
64 – 67 | D
60 – 64 | D-
< 60 | F
Auto-Grader and Testing:
Test-driven development (TDD) is a topic of particular importance in both academia and industry. Students should utilize black-box testing after covering the material in Part II of the course. Students should utilize white-box testing and unit testing after covering that material in Part III of the course.

Students should address Auto-Grader test difficulties after the first assignment submission. Programs usually fail tests because students do not account for an edge case. Do not email the instructor, “What test did I fail?”! One of this course’s objectives is to get students to think about how they should test their problems’ solutions before implementation. Determining which test cases failed is part of this course’s problem-solving aspect.

The Auto-Grader will provide a “grade”; however, the instructor manually grades all final submissions to adjust for things the Auto-Grader cannot account for. If students receive partial credit on all test cases from the Auto-Grader, that typically means they added something to the output (e.g., extra punctuation). This will not be counted against you and will be addressed prior to grade postings.

Course Policies:
Assignments
- Turned in through the CS submission system (not Blackboard)
- Graded by the Auto-Grader (results sent to students’ CS email)
- Unless explicitly otherwise, students may submit a total of three (3) times per assignment (instructor takes the best submission)
- Students may not submit after viewing the sample solution

Due Dates and Late Submissions
You may schedule your own time; however, you cannot defer all coursework until the last month of the semester. Due to the nature of this course, a few hard deadlines exist:
- The dates of the three (3) exams
- Each part’s assignments are due before that part’s exam
- The semester project is submitted in phases (the instructor provides feedback as students progress to the next phase)

Late submissions will only be accepted up to one (1) week after the due date and will incur a 20% penalty. There are no late submissions for any phase of the semester project. There are no make-up exams. Only university exceptions will be allowed (i.e., unusual situations or unforeseen circumstances outside the student’s control). For conflicts with foreseeable deadlines, prior arrangements must be made.

Suggestions for Success:
Students are highly encouraged to adopt successful behaviors:
- Write test cases before attempting to solve the problem. Determine how you will test your solution before you write your solution. Test-Driven Development (TDD) is an important skill to develop.
- Pay close attention to all necessary scenarios when designing your test cases. Failed test cases usually mean an unaccounted for edge case and should be addressed after the first assignment submission.
- Be deliberately mindful in the changes made for the 2nd and 3rd submissions. Do not simply resubmit hoping for a better grade.
- Be systematic (i.e., (1) review the assignment, (2) review your testing procedures, (3) reevaluate your test cases)

Academic Honesty:
Everything turned in for grading in this course must be your own work. The instructor reserves the right to question a student orally and/or in writing and use his evaluation of the student’s understanding of the assignment and of the submitted solution as evidence of cheating. Violations of the Honor Code will be reported to the Office of Student Conduct and Academic Integrity. This includes students who contribute to violations by sharing their solutions.

This policy is not intended to prevent students from providing legitimate assistance to one another. Students are encouraged to collaborate to aid in the learning process. General assistance on the subject matter of the course is acceptable and encouraged. Specific discussions of solutions to any graded work are forbidden.

Honor Pledge:
I pledge to support the honor system of Old Dominion University. I will refrain from any form of academic dishonesty or deception, such as cheating or plagiarism. I am aware that as a member of the academic community, it is my responsibility to turn in all suspected violators of the honor system. I will report to Honor Council hearings if I am summoned.

By attending Old Dominion University, you have accepted the responsibility to abide by this code. This is an institutional policy, approved by the Board of Visitors.

Educational Accessibility:
Old Dominion University is committed to ensuring equal access to all qualified students with disabilities in accordance with the Americans with Disabilities Act (ADA). The Office of Educational Accessibility (OEA) is the campus office that works with students who have disabilities to provide and/or arrange reasonable accommodations.
- If you experience a disability which will impact your ability to access any aspect of the course, present me with an accommodation letter from OEA so that we can work together to ensure that appropriate accommodations are available to you.
- If you feel that you will experience barriers to your ability to learn and/or complete examinations in the course but do not have an accommodation letter, consider scheduling an appointment with OEA to determine if academic accommodations are necessary.

The Office of Educational Accessibility is located at 1021 Student Success Center, and their phone number is (757)683-4655. Additional information is available at the OEA website (http://www.odu.edu/educationalaccessibility/).