Instructor: Ben Keller  
525 McBryde (231-9367)  
\texttt{keller@cs.vt.edu}

Office Hours: 8:30-10 MWF, and by appointment (subject to change)

GTAs: Matt Aguirre, Celia Wang.  
Session assignments, contact information and consulting hours will be posted on the web page.

Objectives: Object-oriented programming concepts are studied and basic skills in software design are developed. Sound practices for design, construction, testing, and debugging of object-oriented software systems are emphasized. Object-oriented features of the C++ programming language are examined. The primary principles and language features studied are: objects, classes, inheritance, and polymorphism.

Prerequisites: CS 2574 or ECpE 2574. Computer Science majors and minors \textit{must} have completed the prerequisite with a final grade of C or better.

Texts: \textit{UML and C++: A Practical Guide to Object-Oriented Development}, by R.C. Lee and W.M. Tepfenhart,  

Course notes will be posted on the course website as available.

Course Website: (http://courses.cs.vt.edu/~cs2704/) The course website will include copies of the course contract (this document), pertinent department policy statements, office hours, test dates, programming project specifications as available, and timely announcements. You are advised to consult the website on a regular basis.

The course website also will have links to other useful information, such as brief tutorial introductions to the MS Visual C++ editor and debugger, example programs, koofers, and the course notes.

Assignments: Your grade will be based on 3 midterm tests, a final exam, homework and in-class assignments, and several programming projects, weighted as follows:

<table>
<thead>
<tr>
<th>Item</th>
<th>Weight</th>
<th>Tentative Dates</th>
</tr>
</thead>
<tbody>
<tr>
<td>Project Testing and SE</td>
<td>40%</td>
<td>TBA</td>
</tr>
<tr>
<td>Homework &amp; activities</td>
<td>10%</td>
<td>TBA</td>
</tr>
<tr>
<td>3 Midterm Tests</td>
<td>30%</td>
<td>Weds, Sept 20, Oct 18, Nov 8</td>
</tr>
<tr>
<td>Final Exam</td>
<td>20%</td>
<td>Tuesday, Dec 12, 1:05-3:05 (Location TBA)</td>
</tr>
</tbody>
</table>

Programming Projects: The programming projects must be implemented in Standard C++, as described in the course notes. You may use any ANSI conformant compiler you like, however all programming assignments submitted are required to compile under the Microsoft Visual C++, version 6.0 on Windows NT. Programs will be tested under Windows NT. It is your responsibility to ensure that your programs execute correctly in the appropriate environment.

The MS Visual C++ 6.0 compiler is the only one supported for this course. That means that neither the instructors nor the GTAs for this course will answer questions about the use of any other compiler, including earlier versions of Visual C++. The Visual C++ compiler is installed on the Windows PCs in the McBryde 116/118 computer lab, and in several University computer labs around campus. If you are using another compiler it may be advisable to test each of your programming projects in the lab prior to submission. If you do not already have a lab account, you may apply for one online through the lab website (http://www.cslab.vt.edu/online.html).

All the programming projects will be submitted electronically, using the Curator System. See the Curator Project Guides Page (http://ei.cs.vt.edu/~eags/CuratorGuides.html) for details and software. Be sure to download and read the \textit{Student Guide to the Curator} — it contains the answers to most of the questions students
have about the Curator System. The Student Guide also contains information about how the Honor Code applies when using the Curator; be sure to read and follow the guidelines given there.

Each of your programming projects will also be graded for adherence to good software engineering principles, including documentation, design, conformance to the stated specification, and programming style. Each project specification will include explicit guidelines that you will be expected to follow, in addition to the general SE principles discussed in class.

Homework and Activities: There will be a number of homework assignments throughout the semester. Homework assignments will also be submitted electronically. We may also do other small tasks during the class period; if graded, these will contribute to your homework grade. These activities may include pop quizzes.

Tests: There will be three smaller tests during the semester. Your score on the final exam will replace you’re the average of your test scores, if it is an improvement. You must bring your Va Tech ID card to the tests and final exam!
Grading Policies: This course is largely devoted to the development of skills in object-oriented software development, as reflected in the relatively heavy weight given to the programming assignments. You will be expected to produce programs that are not only functionally correct, but also well structured, well documented and readable. The Computer Science Department Documentation Standards, described in *Elements of Programming Style*, will be enforced on all programming assignments. A copy of this document is available from the course website.

Backups: It is your responsibility to maintain an up-to-date backup copy of each programming project (that is in addition to the copy you submit). The hard drives of the lab machines are recloned periodically, so don't try to leave a backup there! Keep a spare copy of all the relevant files for each project on floppy disk in case your assignment is mislaid.

Late Work: The most common problem that students have with this course is actually getting the projects submitted in time. So, this policy is less restrictive than usual, but I will be firm in enforcing deadlines.

Each programming project will be graded by demonstration with the GTA for your section. Each project may be demonstrated no earlier than one week after the assignment has been made, and no later than five weeks after the assignment is made, or 1 December 2000, whichever is sooner. Demonstrations must be scheduled within the first week after the initial assignment date, or a 5% penalty will be applied. Each GTA will have a fixed number of demonstration periods in each week, and he or she is not required to grade your demonstration exactly when you want, or, if you don’t make an appointment in advance.

Bug fixes will be allowed during demonstrations; however, major changes (the GTA’s judgment, not yours) will incur a penalty of 10% each. A second demonstration may be allowed provided that it does not push your demonstration date past the deadline. A penalty of 20% will be applied for a second demonstration. Programs must be submitted electronically prior to 6 a.m. the day of your demonstration.

Realize that this policy makes you responsible for your schedule. You can expect assignments to be made three to four weeks apart, with the possibility that they will be posted earlier than the “initial assignment” date on the assignment. You are discouraged from beginning to program a project that requires techniques or concepts that haven’t been covered in class. In general, you need to be extremely careful with your time, especially if you need to rely on the departmental labs.

Note well: delays resulting from machine availability, lab schedules, hardware failures or your failure to maintain a backup of your work do not merit an extension.

No other coursework will be accepted late.

Statute of Limitations: Any questions or complaints regarding the grading of an assignment or test must be raised within two weeks after the score or the graded assignment is made available (not when you pick it up).

Absences: If a serious illness prevents you from taking any of the tests, send a friend with a note describing your condition or notify me before the day of the test. Also, to establish a valid excuse for an illness you must get a note from a physician or the University infirmary. Before missing a test for any reason, you must make every effort to discuss the problem with me before the day of the test. Excuses other than an illness must be reported to your Dean's office so that they can send me a written explanation of the absence. If you need to be away for an official University event, this must be cleared with me in advance. Without a valid excuse, no makeup tests or exam will be given.

Grade Scale: Final grades will be set according to the usual 10-point scale; i.e., 90% guarantees at least an A-, 80% at least a B-, etc. A curve may or may not be applied to the final averages; that decision will be based on the overall class performance. The decision to utilize a curve rests entirely with the course instructor.
Honor Code: An exhaustive list of Honor Code violations would be impossible to present here, but among other things, each of the following is a flagrant violation of the Virginia Tech Honor Code, and violations will be dealt with severely (Honor Court):

♦ Working with another student to derive a common program or solution to a problem. **There are no group programming projects in this course.**
♦ Discussing the details required to solve a programming assignment. You may not share solutions.
♦ Copying source code (programs) in whole or in part from someone else.
♦ Copying files from another student's disk even though they might be unprotected.
♦ Editing (computer generated) output to achieve apparently correct results.
♦ Taking another person's printout from a lab printer, remote rprint printer, trash can, etc.

It is acceptable to discuss with classmates a programming assignment in a general way, i.e., to discuss the **nature** of the assignment. In other words, you may discuss with your classmates what your program is required to accomplish but not how to achieve that goal using C++. In no way should the individual statements of a program or the steps leading to the solution of the problem be discussed with or shown to anyone except those people cited in the following statement.

Feel free to discuss the assignment and your program specifically with the instructor or graduate teaching assistant. The discussion of your individual program must be limited to these people.

If you have any question as to how the Honor Code applies to this class, remember that:

♦ Any work done in this class must be done on an individual basis.
♦ Credit will be given only for work done entirely on an individual basis.
♦ Do not make any assumptions as to who can provide help on a programming assignment.
♦ Evidence indicating the violation of the policy stated above will be turned in directly to the Honor Court.
♦ It is much easier to explain a poor grade to parents or a potential employer than to explain an Honor Court conviction.

In addition, the Honor Code statement included in the *Student Guide to the Curator* is in force for this class.

The Honor Code will be strictly enforced in this course. All assignments submitted shall be considered pledged graded work, unless otherwise noted. All aspects of your work will be covered by the Honor System. Honesty in your academic work will develop into professional integrity. The faculty and students of Virginia Tech will not tolerate any form of academic dishonesty.