CS 2605

Computer Science 2605
Data Structures and OO Development I

Canyon de Chelly, Navajo Nation

Instructor and TAs

William D McQuain

<table>
<thead>
<tr>
<th>Email</th>
<th><a href="mailto:wmcquain@cs.vt.edu">wmcquain@cs.vt.edu</a></th>
</tr>
</thead>
<tbody>
<tr>
<td>Office</td>
<td>631 McBryde Hall</td>
</tr>
<tr>
<td>Office Hours</td>
<td>2:00 – 3:00 MTWRF, and by appointment</td>
</tr>
</tbody>
</table>

Morgan Pitkin

<table>
<thead>
<tr>
<th>Email</th>
<th>TBA</th>
</tr>
</thead>
<tbody>
<tr>
<td>Office</td>
<td>118 McBryde Hall</td>
</tr>
<tr>
<td>Office Hours</td>
<td>TBA</td>
</tr>
</tbody>
</table>
## Prerequisites

**CS 1706 or ECE 2574**

CS Majors and Minors must have completed CS 1706 with a grade of C or higher (C- is not acceptable). CpE Majors must have completed ECE 2574 with a grade of C- or higher.

**Math 2534**

There is no grade requirement for Math 2534, other than a passing grade.

There will be absolutely NO exceptions to these requirements.

**Note**

Lying about whether you meet the prerequisites is an Honor Code offense. Students who do so may be charged at the sole discretion of the Department of Computer Science.

## Texts and Course Website

### Required:

*Data Structures and Algorithms in C++, 3rd Edition*

Adam Drozdek, Brooks/Cole, ©2005

ISBN 0-534-49182-0

*C++ for Java Programmers*

Mark Allen Weiss, Prentice-Hall, ©2004


### Course website:

courses.cs.vt.edu/~cs2605/summer12006
Other Useful References

C++: How to Program, 5th Ed., by Deitel & Deitel, Prentice-Hall, ©2005

CS 1044 Course Notes, Summer 2006 Edition, N D Barnette & W D McQuain
©2003 at: courses.cs.vt.edu/~cs1044

CS 1704 Course Notes, Fall 2003 Edition, N D Barnette & W D McQuain
©2003 at: courses.cs.vt.edu/~cs1704

Thinking in C++ by Bruce Eckel:
www.bruceeeckel.com

The C++ Programming Language, 3rd Ed. by Bjarne Stroustrup, Addison-Wesley, ©2005

Assignments and Weights

<table>
<thead>
<tr>
<th>Item</th>
<th>Weight</th>
<th>Tentative Dates</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pop Quizzes</td>
<td>8%</td>
<td>Frequent</td>
</tr>
<tr>
<td>Homework and Online Quizzes</td>
<td>20%</td>
<td>Varied, posted on course website</td>
</tr>
<tr>
<td>Programming Assignments</td>
<td>35%</td>
<td>Varied, posted on course website</td>
</tr>
<tr>
<td>Midterm Test</td>
<td>12%</td>
<td>Friday, June 9</td>
</tr>
<tr>
<td>Final Exam</td>
<td>25%</td>
<td>10:30 – 12:30 pm, Saturday July 1</td>
</tr>
</tbody>
</table>

Pop quizzes short, typically one question, taken from the assigned reading, or from the lectures, or from current programming assignment; no makeups, some will be dropped

Homework smallish programs, design problems, etc.

Online quizzes multiple-choice questions, usually relating to the syntax, semantics and use of the C++ language; late homework will only be accepted in extremely unusual circumstances

Programs gradually grow in complexity; grading will be based primarily upon whether your version solves the specified problem; internal documentation and coding style may be used as a factor as well; late submissions will carry a per-diem penalty
Grading Policies

Grade Scale
The usual 10-point scale will apply (subject to any curve). A final average of 90% will guarantee an A-, 80% will guarantee a B-, and so forth.

Curve
A grade curve may or may not be employed in this course. The application of a curve is dependent upon class performance on tests and homework. The decision to utilize a curve rests entirely with the course instructor.

Cheating
Cheating on the programming assignments, quizzes and tests will not be tolerated. Student submissions for the programming assignments will be subjected to automated analysis to detect suspicious similarities. We are very good at this.

I reserve the right to demand that a student explain the logic and/or language of any programming assignment. Inability to do so will lead to charges.

Statute of Limitations
Questions about the grading of an individual assignment must be raised within one week of the graded results being made available to the class.

Projects

S/E and Documentation Evaluation
All programming projects will 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 or refer to explicit guidelines that you will be expected to follow. In particular, you will always be expected to follow the guidelines on the Programming Standards page of the course website.

If the TAs evaluate an auto-graded project, they will grade your submission to the Curator that received the highest score. In the event of a tie for highest, they will grade the earliest one. Note that if you make an incomplete submission (e.g., omitting required documentation) and that receives a perfect score, then the TAs will evaluate that incomplete submission.

There will be no exceptions to this policy.
Development System

Test Environments
- All programming assignments submitted are required to compile under g++ 3.4, as installed in the McBryde 118 lab.
- Unless specified otherwise, programs will only be tested under that environment.
- It is the YOUR responsibility to ensure that YOUR programs execute correctly in the appropriate environment; programs that do not will receive substantial deductions.

Students developing on other systems, or using another C++ compiler are warned that there are many pitfalls and that they will be given NO compensation for those pitfalls.

Compliance with the ISO C++ Standard varies widely among older compilers, especially g++ prior to version 3.2 and Visual C++ prior to .NET 2003.

Your programs WILL be tested with the environment listed above. If it fails to compile, or exhibits incorrect behavior, we don’t care that it may compile elsewhere, or appear to run correctly elsewhere.

Late Work

Due dates
Each programming project and homework assignment will have a due date and time and will include instructions for submission.

Homework
Usually, no late submissions will be allowed for homework assignments.

Projects
Except in the very rare case that an extension is granted, late submissions will incur a penalty per diem late penalty that will be included in the project specification. This is typically 10%.

Extensions
Any request for an extension must be made, preferably by email, at least 24 hours prior to the due date.
Late submissions will not be given any credit if submitted after graded assignments or solutions have been released.

Statute of Limitations
Any questions about the grading of an assignment must be raised with your instructor within two weeks after the graded assignment has been made available to you.
### Sources of Help for This Course

#### General Issues
- CS 2605 classmates
- CS 2605 Forum online at forum.cs.vt.edu
- CS 2605 TA
- CS 2605 Instructor

#### C++ Language Help
- CS 2605 Forum
- texts from earlier courses
- alt.comp.lang.learn.c-c++

#### Lecture Instruction
Lectures will consist of presentations, applications, problems and solutions interspersed with classroom discussion.

### Damage Control

#### Backups

Students are responsible for making backup copies of all their work in this (and all) courses.

Loss of work due to hard drive failure is **NOT** an acceptable excuse. Backup copies of files on the same hard drive are not backup copies. Backup copies of files on second hard drives are also risky. Backup copies should be maintained on two separate distinct storage mediums, (e.g., hard drives and Zip disks).

Backup copies should be maintained until after the end of the term and students have received their course grade. (The Army lives by triplicate for a reason.)

Remember: Computer systems are mechanical devices.
- Systems fail.
- Drives die.
- Bad sectors appear.
- Network connections break.
- Plan for it. It is inevitable!
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 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 reprint 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 homework assignments and your program source code with the teaching assistants assigned to CS 2604, the instructor, or the free tutors provided by UPE. The discussion of your program source code must be limited to these people. Note that this specifically excludes discussions of your program source code with other students (even if they are not enrolled in CS 2605), or with tutors except for those named above. Privately hired tutors are not an exception to this requirement, nor are athletic or other tutors provided by the University.

Copies of all submitted work are retained indefinitely by the Department. Submitted programs are subjected to automated analysis for detection of cheating.

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.
- All submitted work is archived. All submitted programs will be subjected to automated cheat analysis.

Evidence indicating the violation of the policies stated above will be submitted 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.