

# CS2604: Data Structures and File Processing

## Fall, 2001

- Class:** (1) TuTh 11:00–12:15pm in McBryde 126  
(2) TuTh 2:00–3:15pm in McBryde 126  
(3) TuTh 2:00–3:15pm in Randolph 216  
(4) TuTh 3:30–4:45pm in McBryde 126
- Instructors:** (4) Dr. D.S. McCrickard, McBryde 623, x6698  
Office Hours: TuTh 2:00–3:00, 5:00–6:00  
E-Mail: mccricks@cs.vt.edu
- (2) Dr. C. North, McBryde 619, x2458  
Office Hours: TuTh 3:30–5:00  
E-Mail: north@cs.vt.edu
- (1 and 3) Dr. C.A. Shaffer, McBryde 331, x4354  
Office Hours: TuTh 10:00–10:45  
E-Mail: shaffer@cs.vt.edu
- GTA:** Sirong Lin, Email: silin@csgrad.vt.edu  
Sandeep Prabhakar, Email: sprabhak@vt.edu  
Yang Rui, Email: yrui@vt.edu  
Allan A. Sioson, Email: asioson@vt.edu
- Prerequisites:** CS2704 and either Math2534 or Math 3034
- Textbook:** *A Practical Introduction to Data Structures and Algorithm Analysis – Second Edition*  
by Clifford A. Shaffer  
A copy of the overheads is available from  
A-1 Copies in University Mall
- Class Homepage:** <http://courses.cs.vt.edu/~cs2604>
- Grade Weighting:** 4 Projects: 45% total  
Midterm and final: 35%  
Pop Quizzes: 5%  
Homework: 15%

### **Honor Code:**

The Honor Code, and in particular, the document “DEPARTMENTAL POLICY ON KOOFERS, OLD PROGRAMS, CHEATING, AND COMPUTER USE,” URL

[http://www.cs.vt.edu/UG\\_handbook/koofers.html](http://www.cs.vt.edu/UG_handbook/koofers.html)

applies to this course and will be strictly enforced. Homework and exams **must** be done strictly on an individual basis. Design and coding of programming assignments **must** be done strictly on an individual basis. 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 the graduate teaching assistants, the instructor, or the free tutors provided by ACM or UPE. Any discussion of your program source code must be limited to these people.

Always give credit for work that is not entirely your own (e.g., parts of programs or homework answers borrowed from a book).

### **Prerequisites:**

The Computer Science Department rigorously enforces the prerequisite requirements for all courses. Additionally, for majors or minors in Computer Science the Department enforces the requirement that all prerequisite Computer Science courses be completed with a grade of C or better. Any student not meeting these requirements and not obtaining written permission from the course instructor to remain in the course, must withdraw from the course within the first week of classes. Any student who is subsequently found not to meet these requirements **will be subject to an honors violation report** on the basis of falsification of qualifications. Instructors are NOT bound to investigate the records of students to ascertain their prerequisite status; this is the student’s own responsibility.

In **all** cases, **the student is responsible** for knowing all prerequisite material.

### **Assignments and Grading Policy:**

This is in large part a programming course, and programming projects count for 45% of your grade. You are expected to produce programs which are both readable and correct. The CS Departmental Documentation Standards entitled “Elements of Programming Style” (URL [courses.cs.vt.edu/~cs2604/Standards/Standards.html](http://courses.cs.vt.edu/~cs2604/Standards/Standards.html)) will be enforced.

One purpose of a data structures course is to teach efficient algorithms and use of appropriate data structures. Another purpose of this course is to exercise your design abilities. It is not sufficient that a program generate the correct answer and be written with good documentation style. Projects will also be graded in part on design and organization quality, and in part on efficiency. You should certainly pay attention when the instructors discuss issues related to “good” and “poor” design choices for the projects. These issues directly affect your grade.

Solutions to homework assignments must be typeset either using a word processor or in plain ASCII text. No handwritten work (including scanned documents) will be accepted.

All programming assignments will have a stated due date, a stated early bonus date (generally 2-3 days preceding the due date) and a stated “drop dead” late submission date (generally 2-3 days after the due date). Working programming assignments handed in by the

specified time on the bonus date will receive a 10% bonus. Programming assignments turned in after the stated due date will typically be penalized 10% per day up to the specified late due date. Programming assignments will not be accepted after that time, unless an extension has been granted.

Homework assignments are due at the date and time specified. No late homework assignments will be accepted unless an extension has been granted.

All assignments will be submitted electronically. The acceptor program used to receive your assignments will provide the official timestamp used to determine whether a program is on time. Assignments will lose 1 point per minute late until reaching the credit level for the next due date. For example, if a program is worth 100 points, and is turned in 3 minutes after the early bonus due date, then it would receive  $100/10 - 3 = 7$  bonus points. If the program were turned in more minutes late than the amount of the early bonus, but prior to the regular due date, it would simply be counted as being turned in on time. A similar calculation applies to projects turned in a few minutes after the regular due date or the late date. **Be warned** – the “few minutes late” penalty is automatic, and there will be no exceptions or mitigating circumstances. **Don't push deadlines.**

Requests for extensions for homework or programs must be made at least 24 hours **in advance** of the due date (**not** the late submission date).

If any student needs special accommodations because of a disability, please contact the instructor during the first week of classes.

### **Equipment and Programming Language:**

All programming for this course will be done in C++. The GTAs will compile and test programs under one of the following two platforms, at the choice of the student: MS Visual C++ Version 6.0 under Windows NT, or Gnu G++ under FreeBSD. It is the responsibility of the student to submit a program that will successfully compile and execute on the specified platform. Computing facilities are available for use in the Departmental Computing Lab in McBryde 118. All Computer Science majors should have lab accounts for use on these machines—if not, please see the consultant on duty in McBryde 116 to apply for one. For non-majors, accounts to the McB 118 machines will be given on request.

Test data files will be provided via the CS2604 WWW site.

### **Class Listserv**

Students enrolled in this course will be added to the class listserv. The listserv is the official source for disseminating administrative information, changes to project specification or due dates, etc. There will also be a newsgroup to serve as the public on-line focus of discussion for the class. You should receive email from the listserv during the first week of class giving information how to contribute to the newsgroup. For more information on the class listserv and newsgroup, see the class web site.