

CS 195 Introduction to Computer Science

Fall 1998

Instructor: Kennan Shelton

Office: 318 Ohlendorf

Office Phone: 843-3721

E-Mail: shelton@mathcs.rhodes.edu

Office Hours: M 9-10, Tu 1-3

W 9-10 & 1-2, Th 4-5, F 9-10

Texts: *C programming, a Modern Approach*

By King

An Invitation to Computer Science

By Schneider and Gersting

Course Description: It is very hard to deny that computers do not affect our lives in many varied ways. From class registration to fuel injection systems in our cars to the video games we play, computers have revolutionized our way of life since their development 50 years ago. Today, knowing how to work with computers in some form is necessary in many occupations and knowing how to program a computer has become a “hot” skill in the job market. But how do computers actually work, and how can we control them? What uses do they have and what limitations? And most importantly, to what extent are they the answer to society’s ills, and what problems have they created? These are the broad questions we will attempt to answer.

There are two large goals to CS 195:

- ◆ To develop an understanding and appreciation of what constitutes “computer science” including
 - the construction and control of computers (computer architecture, compilers, programming languages and operating systems),
 - the theoretical aspects of computers (computability, what computers can and cannot do),
 - the social effects of computers (ethical and legal issues);
- ◆ to acquire a solid introduction to computer programming, specifically using the C programming language.

We will spend about half of the time discussing issues from Schneider and Gersting. The other half of the time will be focused on learning to program in C, using King as a primary source. A day-by-day class schedule is provided—it’s quite packed, so some adjustment will likely be necessary as the semester progresses.

Prerequisites: There are no formal prerequisites for this course. While I will not be covering such information as working with Windows95 or running software such as word processors or spreadsheets, I do assume that you are familiar with the basic operation of computers (e.g., how to use the keyboard and the mouse).

Homework and Labs: There is a lab associated with this course, worth one hour of credit. You must sign up for either the Monday or Thursday sessions (both run from 1 to 4 in the afternoon). The labs will examine further the programming elements of the course, providing you with sample programs as well as assignments for you to complete. There will be a fairly large number of programming assignments so that you have plenty of practice—some programs will be very short, others may require that you complete them outside of lab. **All lab assignments are due by the start of the next lab (or one week, whichever is less).** I will give you particulars on how to turn in lab work at the first lab.

I will sometimes assign homework problems from the texts, most often from Schneider and Gersting. I will inform you of the due dates at the time of assignment.

I encourage you to work together on homework, preferably in groups of no more than 3; you often learn the most while working with others. Be careful that you use other students effectively—simply copying their solutions will not guarantee understanding. **All work submitted to me must be your own—by putting your name on it, you are claiming it as your own original work.**

Projects: There will be 3 programming projects on topics I will assign. The first two projects are to be completed on your own—only the books, notes and myself are to be consulted. The final project will be a team project—you will work together in teams of 3 or 4 to complete the assignment, then (time permitting) make a “sales pitch” to the rest of the class. I will grade your projects on the correctness of the program, as well as how well it is written and documented.

Mid-term and Final Exam: The mid-term is to be taken in-class and is scheduled for Thursday, October 8. The final exam is scheduled for **Tuesday, December 15 at 1:00pm.** The final exam will focus primarily on material from the latter half of the course.

Grading and Expectations: All lab assignments, homework questions and test problems will be scored out of 4 possible points. In grading I will not only look for a correct, working program (or solution) but also good programming style, including proper documentation. For written answers, I will also examine the clarity, coherence and conciseness of your solution. It is possible to get the right answer or a working program and not earn full credit. Here’s the general meaning of each score:

- 4 The answer is correct; the solution is complete, well written and easy to read and follow; program works properly, is well documented, and displays good programming style; no complaints; woo hoo!
- 3 The answer is incorrect, but the mistake was minor; or, the solution/program is correct but could be improved; the program works, but lacks proper documentation

- 2 A reasonable attempt was made to solve the problem but with significant mistakes that prevented a complete or correct solution; program doesn't work but could be easily fixed
- 1 A decent attempt was made to solve the problem but nothing tangible was produced; program shows some useful elements, but not enough to produce and easily fixable program
- 0 No work submitted, or no real progress toward a solution; gross mistakes

The projects will be graded on the usual A, B, C, D, F scale.

I expect all of your work to be of high quality. This means that it should be neat, clean, clear and, in general, well written. Just because you are writing about computers and/or mathematics does not mean that you should forget how to communicate, and I will not give credit for work that I cannot read and make sense of.

Your semester grade will be determined by the homework and labs, the mid-term, the three projects and the final exam. The exact breakdown is:

Homework and Labs:	20%
Mid-term:	25%
Project 1:	5%
Project 2:	10%
Project 3 (Team):	15%
Final Exam:	25%

Descriptions of letter grades according to the Rhodes College Catalogue, with my interpretations:

A (Excellent)	You have clearly demonstrated a significant mastery of all the material and shown deep understanding of the major topics.
B (Good)	You have displayed mastery of most of the material and understanding of the fundamental concepts involved—all students should strive to attain at least this grade
C (Satisfactory)	You have some idea of what is going on, but have developed a mastery of only a small part of the material (“Satisfactory” is never satisfactory)
D (Passing)	You appear to grasp a few major points but seem to understand only a little of the others
F (Failure)	You’ll have to work hard to get this grade (or not at all)—I don’t like giving it out

Note that just being a warm body in the classroom is *not* “Satisfactory”.

Missed Tests and Late Work Policy: I do not accept late homework or programming assignments. Late projects will be accepted, but at a reduction in grade (one letter grade per day, to a maximum of 2 days). If you must miss the mid-term for a *very good reason*, inform me as soon as you know about the conflict and be prepared to provide documentation. Very good reasons to miss the mid-term do *not* include non-academic, non-emergency travel. You may e-mail, leave a message on my voice mail, or send a carrier pigeon, just as long as I know of the difficulty as soon as you do. Any delay will jeopardize your chance to take a make-up or take the test early.

Attendance: I will not keep a roll, but I would very much like to see your smiling face in every class. Class participation is not an explicit part of your grade, but will be taken into account when assigning semester grades, especially borderline cases. Excessive absences may be reported to the Dean.

Honor Code: It is expected that each student in this class will conduct him or herself within the guidelines of the Rhodes Honor Code. All academic work should be done with the high level of honesty and integrity that the students and faculty at Rhodes demand. If you have any questions about what is or is not allowed, please ask me.

Class Schedule: We will try to stay with this schedule, but some adjustment will likely be necessary. You are expected to have read the material listed by the given day.

Thur. Aug. 27	Introduction
Tues. Sept. 1	Algorithms and Computer Science (SG Ch. 1-2)
Thur. Sept. 3	Simple programs in C (King Ch. 1-4)
Tues. Sept. 8	Numbers and Logic (SG Ch. 4)
Thur. Sept. 10	Selection Statements (King Ch. 5)
Tues. Sept. 15	Numbers and Logic, cont.
Thur. Sept. 17	Loops (King Ch. 6)—First project assignment
Tues. Sept. 22	Computer architecture (SG Ch. 5)
Thur. Sept. 24	Functions (King 9-10)
Tues. Sept. 29	History of Computers (SG Ch. 5)
Thur. Oct. 1	Recursion
Tues. Oct. 6	Review/Catch-up—First project due
Thur. Oct. 8	Mid-term exam
Tues. Oct. 13	Operating Systems (SG Ch. 6)
Thur. Oct. 15	Arrays (King Ch. 8)
Thur. Oct. 22	Pointers (King Ch. 11-12)—Second project assign.
Tues. Oct. 27	Compilers (SG Ch. 9)
Thur. Oct. 29	Advanced Pointers (King Ch. 17)
Tues. Nov. 3	Computability (SG Ch. 10)—Team project assign.
Thur. Nov. 5	Strings (King Ch. 13)
Tues. Nov. 10	Internet and networking (SG Ch. 11)
Thur. Nov. 12	Structures, Unions, & Enumerations (King Ch. 16)
Tues. Nov. 17	Social Issues (SG Ch. 13)—Second project due
Thur. Nov. 19	Structures, etc. cont.

Tues. Nov. 24

Social issues cont.

Tues. Dec. 1

Team Presentations *--Team projects due

Thur. Dec 3

Team Presentations *

Tues. Dec. 8

Review

*Presentations if time permits, otherwise we'll play catch-up