

GMU Spring 2021

CS 222-DL2 Computer Programming for Engineers

Instructor: Hal Greenwald (hgreenwa@gmu.edu)

1:30 pm - 2:45 pm	TR	ON LINE	Lecture
-------------------	----	---------	---------

Office Hours: by appointment

REQUIRED TEXTBOOK: Hanly and Koffman,
"Problem Solving and Program Design in C," 8th ed.

DESCRIPTION: CS 222 is a second course in computer programming, emphasizing programming concepts relevant to engineers. Topics include both higher level and lower level language aspects as well as elementary data structures. Students are expected to have a basic understanding of programming in some high-level language. CS 222 will be taught using C language, however students are not presumed to be familiar with C prior to this class. This course is intended as a terminal course in programming for engineers.

Prerequisite: C or better in CS 112.

Course outcomes:

1. Be able to implement, test, and debug a designed solution to a problem using the C programming language.
2. Demonstrate a good understanding of C language constructs such as pointers, dynamic memory management, and address arithmetic.
3. Demonstrate a good understanding of C libraries for input and output, and the interface between C programs and the UNIX operating system.
4. Demonstrate an ability to use UNIX tools for program development.

COURSE POLICIES

Class: Students are expected to attend lecture each week and are responsible for all material covered during lecture.

Programming Assignments: Programming assignments will be posted on Blackboard as they are assigned, and must be submitted on Blackboard by the assigned due date. If your program is incomplete, you may still submit for partial credit. **Your code must run without obvious errors (even if not all functionality is present).** Building your programs in a modular fashion, and debugging as you go, are crucial concepts in program development. Any programming assignment that is submitted but either does not compile or has major errors when it is run will receive no more than 50% credit.

Students are expected to work independently outside of class to familiarize themselves with the GMU computer systems, to read and review all assigned materials and to complete all homework and programming projects.

Reading: Students are responsible for reading and understanding all assigned material. Be aware that some material covered in class may not be found in your textbook. If you do not understand any of the covered material or reading assignments, there will be opportunities to ask questions in and after class.

Due Dates: Due dates for all programming projects will be posted on Blackboard. Homework assignments are due in the manner stated (either through Blackboard or in class.)

Late work: No late submissions are permitted. If your program isn't the way you'd like it to be when the deadline arrives, submit it anyway for partial credit.

GRADING POLICY

Your course grade will be an aggregate of the following items:

- Homework (40 points)
- Midterm Exam (30 points)
- Final Exam - cumulative (30 points)

Grades will be assigned as follows:

A: 90 - 100%
B: 80 - (90)%
C: 70 - (80)%
D: 60 - (70)%
F: 0 - (60)%

CLASS COMMUNICATIONS

CS 222 will be using the Blackboard system as well as Piazza for most class communications. You are responsible for any notifications or information posted on Blackboard and Piazza. Please check these platforms regularly for such notices. Individual communications with the professor may be done by email using your GMU email account.

When you email, please be sure to include your name, the class number and the topic in the subject header. (E.g.: Subject: Sam Jones / CS 222 / assignment 2)

PROGRAMMING POLICIES

(1) **No sharing or discussion of code for assignments.** Unless specifically stated otherwise, all assignments are individual projects, not group projects. Students are expected to do their own work, not to share programs with each other, nor copy programs from anyone else. Any discussion or sharing of code outside these guidelines constitutes an honor code violation. Suspected honor code violations are taken very seriously, and will be reported to the Honor Committee.

(See <https://oai.gmu.edu/mason-honor-code/>)

(2) **No incorporation of code from any source external to the course.** You may not incorporate code written by others. Of course, you may freely use any code provided as part of the project specifications, and you need not credit the source. Working something out together with the instructor usually will not require crediting the source.

(3) **Back up your program regularly.** You are expected to backup your program in separate files as you get different pieces working. Failure to do this may result in your getting a much lower grade on a program if last minute problems occur. (Accidentally deleting your program, having problems connecting, etc., will not be accepted as excuses.)

(4) **Keep an untouched copy of your final code submission.** It is important that you not touch your programs once you have made your final submission. If there are any submission problems, consideration for credit will only be given if it can be verified that the programs were not changed after being submitted.

(5) **Programs must be developed using UNIX.** Students may develop programs using any computer system they have available, however it must run on the UNIX operating system. Submitted projects must be compiled using the GNU Compiler Collection (gcc) and display properly with the vi editor. Your documentation should clearly state the hardware/software used for development. Once makefiles are introduced, a makefile should be included with each assignment submission. No extensions will be given due to compiler incompatibilities.

TENTATIVE CLASS SCHEDULE

The following represents the essential schedule of topics that will be covered during lectures in the weeks indicated below. Reading assignments from the textbook or from slides/handouts and homework/programming assignments may be adjusted. These adjustments will be clearly stated under Blackboard Announcements.

Note: Syllabus may be subject to modification.

Date:	Topics include:	Reading Assignment
week 1	Introduction, Syllabus, Overview of C HW # 1 assigned	lecture notes & ch 1,2,3
week 2	Overview of C <i>continued</i> , vi editor, compilation platforms, Top-down Design with Functions, Selection Structures, Repetition and Loops, Logical Operators, Modular Programming HW # 2 assigned	lecture notes & ch 3,4,5
week 3	Pre/Post Increment Operators, POSIX, User-defined functions, Function Prototypes, Arrays, Pointers	lecture notes & ch 6
week 4	User-defined functions, Arrays, Pointers <i>continued</i> , C structures, scope rules HW # 3 assigned	lecture notes & ch 7
week 5	Character Arrays and Strings, Text File Processing, Reference and Dereference Operators	lecture notes & ch 8
week 6	The ASCII Character Set, Structures, Unions, Unix Time, Type Casting, Debugging Techniques, Text File Processing	lecture notes & ch 10, 11
week 7	Recursion, Text and Binary File Processing Midterm Exam on Thursday, March 18 HW # 4 assigned	lecture notes & ch 9,10,11
week 8	Scope Rules for C Variables, Array of Structures, Case Study	lecture notes
week 9	Command Line Arguments, Programming in the Large, Makefiles, Compilation of multiple files	lecture notes & ch 12
week 10	The GNU Project Debugger, File I/O <i>continued</i> , Command Line Arguments <i>continued</i> , Pointers to Structures	lecture notes
week 11	Dynamic Memory Allocation, examples and analysis HW # 5 assigned	lecture notes & ch 13
week 12	Linked Lists, Binary Files <i>continued</i>	lecture notes & ch 13
week 13	Linked Lists <i>continued</i> , Static variables Examples and analysis	lecture notes
week 14	Linked Lists <i>concluded</i> , case study	lecture notes
May. 4th	Final Exam (online, cumulative)	

OTHER POLICIES

University Requirements

- Academic Honesty: <https://oai.gmu.edu/mason-honor-code/>
- Disability Statement: <http://ods.gmu.edu>.

ACADEMIC INTEGRITY

GMU is an Honor Code university; please see the University Catalog for a full description of the code and the honor committee process. The principle of academic integrity is taken very seriously and violations are treated gravely. What does academic integrity mean in this course? Essentially this: when you are responsible for a task, you will perform that task. When you rely on someone else's work in an aspect of the performance of that task, you will give full credit in the proper, accepted form. Another aspect of academic integrity is the free play of ideas. Vigorous discussion and debate are encouraged in this course, with the firm expectation that all aspects of the class will be conducted with civility and respect for differing ideas, perspectives, and traditions. When in doubt (of any kind) please ask for guidance and clarification.

GMU EMAIL ACCOUNTS

Students must activate their GMU email accounts to receive important University information, including messages related to this class. You should understand that your mailbox may be full, so check your email and make sure you have enough space daily.

OFFICE OF DISABILITY SERVICES

If you are a student with a disability and you need academic accommodations, please see me and contact the Office of Disability Services (ODS) at 993-2474. All academic accommodations must be arranged through the ODS. <http://ods.gmu.edu>

OTHER USEFUL CAMPUS RESOURCES:

WRITING CENTER: A114 Robinson Hall; (703) 993-1200;

<http://writingcenter.gmu.edu>

UNIVERSITY LIBRARIES Ask a Librarian

<http://library.gmu.edu/mudge/IM/IMRef.html>

COUNSELING AND PSYCHOLOGICAL SERVICES (CAPS):

(703) 993-2380;

<http://caps.gmu.edu>

UNIVERSITY POLICIES

The University Catalog, <http://catalog.gmu.edu>, is the central resource for university policies affecting student, faculty, and staff conduct in university affairs.