GMU Spring 2024

CS 222: *Computer Programming for Engineers* (3 credits)

7:30 AM - 8:45 AM	Tue/Thu (TR)	PLANET 120

Instructor: Justin Wilson (jwilso86@gmu.edu); TA: To Be Updated

---

Office Hours/Support: after class, by appointment, or email using your GMU email
Textbook: "Problem Solving and Program Design in C," 8th edition Hanly & Koffman
Credits: parts of this document sourced from previous syllabi of Hal Greenwald and Jim Chen

---

Description: Introduction to C as a second programming language with emphasis on problems and language features relevant to engineers. Students are not presumed to be familiar with C prior to this class. Topics include basic data types, pointers, elementary data structures, file/output, bitwise operations, and Unix commands for compilation and debugging. Intended as terminal course for some in programming for engineers (Source URL). *Prerequisite*: C or better in CS 112

Course Outcomes:

1. Use the C programming language to implement, test, and debug designed solutions to problems based on the requirements given.
2. Demonstrate proficiency of C language constructs such as pointers, dynamic memory management, and address arithmetic.
3. Display a good understanding of C libraries for input and output, and the interface between C programs and the UNIX operating system.
4. Gain experience developing and maintaining programs using UNIX tools.

Course Policies:

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

HW/Programming Assignments: assignments will be distributed in class and/or posted on Blackboard as they are assigned, and must be submitted as stated (in class or on Blackboard) by the assigned due date. If your work is incomplete, still submit for partial credit. **Your code must run without obvious errors (even if not all requirements are met). Code that does not compile/run will receive no more than 50% credit.** Remember top-down design:

- Write each step of the initial algorithm as a comment in the main function body
- Code each algorithm step in-line or as a function call, one at a time
- Save, compile, run, and test (SCRAT) for each added statement
- You should **not** write your program as a story from beginning to the end, and then compile, debug, and run! Debugging is much easier if you code and test each step as you go.

Work Independently: Students are expected to work independently outside of class to become familiar with GMU computer systems, to read and review all assigned materials, and to complete all homework, programming assignments, and exams. See Honor Code.

Reading: Students are responsible for reading and understanding all assigned material. Use resources such as office hours and/or Blackboard for help. Also, note that some material covered in class may not be found in your textbook. If you do not understand covered material, assignments, readings, or anything at all during the course, don't hesitate to ask.

Due Dates: Assignments are due in the manner stated (either through Blackboard or in class).

Late Work: No late submissions are permitted. Submit what you have before the deadline to be eligible to receive any partial credit.

Grading Policy: This course is graded on the Undergraduate Regular scale, defined as a letter (column 1) and + or – (column 2) in the table below based on a weighted total rounded to the nearest integer.

A: 90 – 100%	90% <= A- < 93% <= A < 97% <= A+
B: 80 – (90)%	80% <= B- < 83% <= B < 87% <= B+
C: 70 – (80)%	70% <= C- < 73% <= C < 77% <= C+
D: 60 – (70)%	D
F: < 60%	F

Your course grade will be a weighted total based on:

- Homework/Programming Assignments (50%)
- Midterm Exam (25%)
- Final Exam – Cumulative (25%)

The midterm and final exams will be held in class.

Class Communication:

In addition to in-class lecture, CS 222 will use Blackboard for class communications. Please check this platform regularly. Individual communications with the professor or teaching assistant (TA) may also be sent by email using your GMU email account.

Programming Policies:

(1) **No copying or sharing 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 Honor Code guidelines constitutes an honor code violation. Suspected honor code violations are taken seriously, and will be reported to the Honor Committee.

Honor Code: "*To promote a stronger sense of mutual responsibility, respect, trust, and fairness among all members of the George Mason University community and with the desire for greater academic and personal achievement, we, the student members of the university community, have set forth this honor code*: **Student members of the George Mason University community pledge not to cheat, plagiarize, steal, or lie in matters related to academic work.**"

(2) **No incorporation of code from any source external to the course**. You may <u>not</u> incorporate code written by others nor code generated from any external source (e.g. generative AI tools, ChatGPT, etc.). Note that these tools can be used to better understand course materials and concepts. Therefore, please credit these and other sources as documentation and references when used to enhance understanding. <u>However, they are not to be used to generate code or solutions to homework/programming assignments</u>.

(3) **Back up your program regularly**. You are expected to backup your program in separate files or a version control system as you get different parts of functionality working. Not only is this good practice, but failure to do this may result in a lower grade if you experience last-minute problems (e.g. accidentally removing a file, connection issues). Remember, late work is not permitted. Submit what you have before the deadline to be eligible to receive any partial credit.

(4) **Keep an untouched copy of your final code submission**. It is important that you keep an unmodified copy of your final code submission. If there are any submission issues, consideration for credit can only be given if the program is verified that it was not changed after the deadline.

(5) **Programs must be compiled, tested, and run using UNIX**. Please be prepared to develop your programs in UNIX on Mason. In the future, according to the assignment requirements, you may be allowed to use other environments with which to develop your code. However, you still must be sure it displays properly with the vi editor and compiles properly using the GNU Compiler Collection (gcc) in UNIX on the Mason cluster. Therefore, if you decide to use a different development environment, you must be familiar with transferring your files to and from the Mason system. Also, your documentation should clearly state the hardware/software used for development. Once makefiles are introduced, a makefile should be included with each assignment submission. <u>No extensions will be given due to compiler incompatibilities</u>.

Tentative Class Schedule:

The following tentative class schedule represents the schedule of topics that will be covered in the weeks indicated below. Reading assignments from the textbook or from slides/handouts and homework/programming assignments may be adjusted; however, any adjustments will be clearly stated under Blackboard Announcements.

Assignments are given every Thursday. HW due in 1 week and programs due in 2 weeks (except Program #1). The assignments are either a) homework questions or b) programs to code. Your assignments will be graded and returned before the next lecture after the due date.

Note: this syllabus may be subject to modification.

Date: (Tue, Thu)	Topics include:	Reading Assignment
Week 1 (1/16, 1/18)	Introduction, Syllabus, Overview of C, vi editor • Thu (1/18): Program #1 assigned & 1 week due Thu (1/25)	Lecture notes & Ch. 1, 2, 3
Week 2 (1/23, 1/25)	Compilation platforms, Top-down Design with Functions, Selection Structures, Repetition and Loops, Logical Operators, Modular Programming • Thu (1/25): HW #1 assigned & due Thu (2/1)	Lecture notes & Ch. 3, 4, 5
Week 3 (1/30, 2/1)	Pre/Post Increment Operators, POSIX, User-defined functions, Function Prototypes, Arrays, Pointers, Git • Thu (2/1): Program #2 assigned & 2 weeks due Thu (2/15)	Lecture notes & Chapter 6
Week 4 (2/6, 2/8)	User-defined functions, Arrays, Pointers *continued*, C structures, scope rules, Big O notation • Program #2 mid check	Lecture notes & Chapter 7
Week 5 (2/13, 2/15)	Character Arrays and Strings, Text File Processing, Reference and Dereference Operators • Thu (2/15): HW #2 assigned & due Thu (2/22)	Lecture notes & Chapter 8
Week 6 (2/20, 2/22)	The ASCII Character Set, Structures, Unions, Unix Time, Type Casting, Debugging Techniques, Text File Processing	Lecture notes & Chapter 10
Week 7 (2/27, 2/29)	Scope Rules for C Variables, Binary File Processing • **Thu (2/29): Midterm Exam (in class)**	Lecture notes & Chapter 11
Week 8 (~~3/5~~, ~~3/7~~)	No classes Week 8. Spring Recess Mon. Mar 4 - Sun. Mar 10	
Week 9 (3/12, 3/14)	Array of Structures, Bitwise Operators	Lecture notes
Week 10 (3/19, 3/21)	Command Line Arguments, Programming in the Large, Makefiles, Compilation of multiple files, Pointers to Structures • Thu (3/21): Program #3 assigned & 2 wk due Thu (4/4)	Lecture notes & Chapter 12
Week 11 (3/26, 3/28)	The GNU Project Debugger, File I/O, Command Line Arguments *continued*, Docker, Recursion • Program #3 mid check	Lecture notes & Chapter 9
Week 12 (4/2, 4/4)	Dynamic Memory Allocation, Pointer Review, Examples and analysis, Static variables, Binary Files *continued* • Thu (4/4): HW #3 assigned & due Thu (4/11)	Lecture notes & Chapter 13
Week 13 (4/9, 4/11)	Linked Lists, Stack, Queue, Recursion *continued* • Thu (4/11): HW #4 assigned & due Thu (4/18)	Lecture notes & Ch. 9 *cont.*
Week 14 (4/16, 4/18)	Tree structures, Ordered Lists, Binary Search, Examples and analysis, Linked Lists *continued*, Binary Search Tree	Lecture notes & Ch. 13 *cont.*
Week 15 (4/23, 4/25)	Review and preparation for Final Exam	Final Exam Review
Week 16 (~~4/30~~, ~~5/2~~)	Last day of classes Mon. Apr 29 so No classes Week 16	
Tue, May 7 7:30-10:15 AM	• **Tue (5/7): Final Exam (in class, cumulative)**	Final Exam

Other Policies:

Course Flexibility Policies

Once per semester, a student can ask the professor for:

- A) "Life Happens Pass" and get a 24-hour extension on 1 assignment
- B) "One Revision Pass" and allow a student to revise 1 assignment within 2 days after it is graded to get up to a 15% increase on prior grade.

University Requirements

- Academic Honesty: https://oai.gmu.edu/full-honor-code-document/
- Disability Statement: https://ds.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. 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. Your email box may be full, so check your email regularly and make sure you have enough space.

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 703-993-2474. All academic accommodations must be arranged through the ODS, https://ds.gmu.edu/.

Other Useful Campus Resources:

Writing Center: A114 Robinson Hall; (703) 993-1200; http://writingcenter.gmu.edu

University Libraries: Ask a Librarian; https://library.gmu.edu/

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. Also, an addendum is provided by the University identifying and describing relevant university policies (link to online version).