ENEE 140 Syllabus

Course information

ENEE 140 provides an introduction to computer programming. The course focuses on leaning important programming principles, and the C programming language is used to illustrate these principles. Topics include an introduction to the programming workflow (coding, compiling, debugging, testing), data types and type conversions, arithmetic operations, control flow, functions, variable scope, vector data types (arrays and strings), input/output.

The material will be taught through reading assignments, in-class discussions, lectures and additional discussions on a message board. ENEE 140 is a hands-on course, where students learn by writing many computer programs outside the classroom.

No prior knowledge of programming is required for this class.

Prerequisites

- Permission from the Electrical & Computer Engineering department.
- Restriction: Must be in the Electrical Engineering program.

Learning outcomes

- Learning how to read C code
- Learning how to use the debugger to understand how programs behave
- Learning top-down problem solving and how to break down the functionality of a program into smaller modules, implemented as C functions
- Understanding the difference between ints and floats, and the result of type conversions
- Learning how to avoid reading/writing beyond array and string bounds (including when using C library functions)
- Understanding loop invariants and exit conditions, and how to implement them with various kinds of loops in C

Logistics

Class Web site: `http://ter.ps/enee140` (includes links to Elms and Piazza)

Instructor: Tudor Dumitraș, `tdumitra@umiacs.umd.edu`
Instructor's Office: AVW 3425
UTFs:

Chris DeFrancisci	Section 0101	`chrisdefrancisci@gmail.com`
Juan Luis Alonso Cruz	Section 0102	`jlalonso123@gmail.com`
Eric Huang	Section 0103	`ricci.p.huang@gmail.com`

Lectures: Tu 3:30 – 4:45 pm in PLS 1140
Labs: W/F 9 am – 12 pm in AVW 1442
Office Hours

Name	Location	Time
Tudor	Tu 2–3 pm	AVW 3425
Chris	Tu 5–6 pm	AVW 1442
Juan	W 2:30–3:30 pm	AVW 1442
Eric	Th 1–2 pm	AVW 1442

Please forgive the UTFs if they are a little late or have to leave a little early, as they might have classes before or after the office hours.

Textbooks

Required:

- **B. Kernighan and D. Ritchie, The C Programming Language, 2nd edition, Prentice Hall 1988.**
  The best book about C, and a great reference for programming in general, but a bit challenging for students who are new to programming. Also known as K&R.

- **S. Summit, C Programming Notes.** (`http://www.eskimo.com/~scs/cclass/krnotes/top.html`)
  Excellent notes to supplement K&R with more detailed explanations.

Recommended:

- **B. Kernighan and R. Pike, The Practice of Programming.**
  A great book on programming principles, with practical advice and examples in several programming languages (including C).

Grading

Your final grade for the course will be based on the following weights:

10%	**Quizzes**	∼**10** short quizzes on the reading assignments, due on Monday before class
20%	**Homeworks**	**10–13** weekly recitation assignments, due every Friday.
35%	**Projects**	**3** programming assignments. Tentative due dates: March 22, April 12, May 10.
10%	**Midterm Exam**	Tuesday, March 29 in class.
25%	**Final Exam**	Wednesday, May 18, 10:30 am – 12:30 pm, in class
5%	**Bonus Points**	(e.g. for answering your classmates' questions on Piazza)

The dates are subject to change. Check the class web page for updates.

Letter grades will be assigned using the following algorithm:

- The +/− system will be used in this course (A+,A,A-,B+, B, . . .).

- There will not be a curve of any kind in any of the above categories.

- The letter grades will be assigned based on the sum of the above categories (100% total) using the following method:

  - Rank the sum of each student from high to low

  - Find the difference between each two adjacent sums (highest–second highest, second highest–third highest, . . .)

  - Make a cut (in letter grade) when the difference is large

- There is no quota for As and there is no guarantee for any kind of grade distribution. Everyone starts the semester with an A and it is yours to lose. You will receive a good grade if you allocate enough time each week to study and to complete all the assignments (including the reading assignments).

- Some statistics from the past semesters:

  - A couple of students get A+ each semester.

  - Typically at least 20% of the students get As, and more than 2/3 get B- or better.

  - A couple of students claim to work very hard but they do not complete their assignments or their submitted programs do not work. These students receive D or F—hopefully we will not have any such cases this semester.

  - Normally, 90% or higher will get at least A-, and 85% or lower will get a at most B+.

Late submissions, re-grading and makeup exams

- **Late homeworks and quizzes will not be accepted**.

- **Late project submissions will be accepted, subject to some non-trivial penalty**. Detailed submission information will be provided with the project assignment.

- If you **dispute your score on any exam/project/lab report**, you must contact Dr. Dumitraș (for the exam) or your UTF/GTA (for others) within one week from the date that the score is announced. After the one-week period, all scores will be considered final and no changes will be made.

- **Project re-grade after debugging**: if you receive a considerably low score on the project and believe that you have only made minor mistakes in your code, please contact Dr. Dumitraș as soon as possible. In most cases, you will be given a couple of days to debug your program and re-submit the project. Based on how much your code has been modified and the nature of the modification, we will re-grade your re-submitted project.
  Important: this policy aims to encourage you to debug your code and to correct the minor mistakes you may have made. Do not abuse this. If the code changes are major, your original score will stand.

- **Homeworks and quizzes will NOT be re-graded**.

- **There will NOT be any make-up midterm exam**. If you must miss the midterm exam and believe that it is an excusable absence according school policy, you must receive permission from Dr. Dumitraș at least 48 hours before the exam so that portion of the grade can go to the final exam. Otherwise, 0 (zero) will be counted as the score for the missed midterm exam. Read school's student attendance policy for more details: `http://faculty.umd.edu/teach/attend_student.html`.

Readings, quizzes and class participation

Because ENEE 140 has only 15 lectures, I won't be able to go over the whole material. Instead, I intend to spend most of the class time discussing the concepts that students find most challenging. This will require you to read **in advance** about the concepts discussed and to try to understand how to apply these concepts in practice.

Each week I will ask you to **read several chapters** from the textbook. These chapters cover topics we will discuss during the following lecture. I will also give you **weekly challenges**—short programs that allow you to practice the programming concepts from the reading assignments. You will not be able to complete the projects without mastering these concepts.

The weekly challenges are not graded. Instead, on Mondays before class you will have to complete **quizzes** that test that you have prepared for the class (read material, attempted to complete the weekly challenge, etc.) These quizzes will not test that you fully understand the material, just that you have done your due diligence in preparing. The quizzes will be administered online on Elms. I anticipate there will be about 10 such quizzes.

During each lecture, I will clarify concepts from the reading assignment and we will solve the weekly challenge together. The best way to assimilate the material is to work on the weekly challenges (even if they seem hard) and come prepared to ask questions at the next lecture.

Communication is key

Good programmers write code that can be understood not only by the computer, but also by other programmers. Therefore, they interact frequently with each other by asking and answering programming questions, in person or on sites like Stack Overflow (`http://stackoverflow.com/questions/tagged/c`. For ENEE 140, we have a **discussion forum on Piazza**, where students can ask questions and where instructors will post important announcements. Please participate in Piazza discussions, interact with your classmates, and contact me and the UTFs if you have questions or concerns.

If you email us, please include "ENEE 140" in the subject so that your emails do not get lost. I promise to reply to all your emails within 3–4 days. For general questions about the class material you are better off using Piazza, where the typical response time is about 45 min.

Academic honesty

The University Code of Academic Integrity, which can be found at `http://thestamp.umd.edu/gh/academics/academic_integrity`, prohibits students from committing the following acts of academic dishonesty: cheating, fabrication, facilitating academic dishonesty, and plagiarism. Academic dishonesty in this class includes copying the homework, project, or exam solutions of other students.

In ENEE 140, **discussing assignment problems and preparing for the exams together is encouraged** (for example, students will receive bonus points for answering general questions on the discussion forum). However, **you must complete all individual assignments by yourselves**. When posting on Piazza, you may include short code snippets to illustrate the problem you are asking about; do not post your entire program, as other students may copy it and this would constitute an instance of academic dishonesty.

Instances of academic dishonesty will be referred to Office of Student Conduct.

Religious holidays

You must **notify me within 2 weeks** from the beginning of class if you will miss an exam due to a religious observance. Requests received after two weeks will not be accommodated.

Students with disabilities

You must **notify me within 2 weeks** from the beginning of class if you require special accommodations due to disabilities. All such requests must be accompanied by documentation from DSS (`http://www.counseling.umd.edu/DSS/`. Requests received after this deadline will not be accommodated.

Students with learning difficulties

If you are experiencing difficulties in keeping up with the academic demands of this course, contact the Learning Assistance Service, 2201 Shoemaker Building, 301-314-7693. Their educational counselors can help with time management, reading, note-taking and exam preparation skills.