

ENGR 101: Introduction to Computers and Programming

Syllabus - Fall 2015

Course Overview

Engineering 101 introduces first-year students to the concept of an algorithm: a well-defined set of instructions that achieve a particular goal. Constructing an algorithm for a given purpose is a fundamental form of engineering design task, and developing computer programs is part of almost every modern engineering project. Students in ENGR 101 learn how to conceptualize algorithms for solving engineering problems and express them in the programming languages MATLAB and C++.

Instructors [please note e-mail policy below]

Section 100: Jeff Ringenberg (jringenb@umich.edu)
M/W, 9:00am - 10:30am in Stamps Auditorium

Section 200: Laura Alford (lslavice@umich.edu)
M/W, 1:30pm - 3:00pm in Stamps Auditorium

Section 300: Rick Niciejewski (niciejew@umich.edu)
Tu/Th, 12:00pm - 1:30pm in Chrysler (Chesebrough) Auditorium

Graduate Student Instructors

Eleanore An	anjwe@umich.edu
Michael Benson	mlbenson@umich.edu
Lisa Bergeron	berglisa@umich.edu
Kevin Collao	kcollao@umich.edu
Kelsey Hockstad	kelshock@umich.edu
Sarah Paris	sarparis@umich.edu
Joseph Scherping	jscherp@umich.edu
Sarah Spitzer	saraylet@umich.edu
Patrick Wilson	pdubslax@umich.edu

Instructional Aides

Joe Korany	joeyak@umich.edu
Sophia (Sonya) Kotov	kotova@umich.edu

Margo Leibold
Sara Rusignuolo
Tyler Seruga
Amit Shah

margolei@umich.edu
srus@umich.edu
tseruga@umich.edu
amitshah@umich.edu

For Office Hours, please refer to the table of office hours on Canvas.

Textbooks

MATLAB : A Practical Introduction to Programming and Problem Solving, 3rd Edition
Stormy Attaway, 2013
ISBN: 0124058760
Available (free for UM students) online at:
<http://www.sciencedirect.com.proxy.lib.umich.edu/science/book/9780124058767>

[Bielejew C++ Book](#) (also posted under Canvas Resources)

Website

Most course materials are made available on the course website, and are considered required reading. A detailed schedule, including lecture topics, assignment due dates and exam dates, is also available on the course site.

Forums and Email

We will be using Piazza to host a course forum. You are required to read this regularly; it is one of the main venues we will use for important course announcements and project clarifications. In addition, it will be a significant source of help and hints on the projects.

We do not answer technical questions via email. In order to save everyone time, we want all students to have the benefit of seeing each question and its answer, so please use the forum.

We ask that you do not post your own solutions, project code, test cases, or output to the forum. Also, please search the forum before posting to avoid questions that have already been asked and answered.

Programming Environment

For this course, you must have a CAEN account. Everyone enrolled in the College of Engineering should have one by default. If you do not, speak with your GSI immediately as it takes a little while to get it set up. We will grade all your programs in the CAEN Linux environment and they must be submitted, compiled (where applicable), and run correctly in this environment.

For programs written in C++, you are free to develop on any platform you like, but you may use only ANSI/ISO standard C++, and are responsible for any differences between your preferred platform and the grading platform. We **highly** recommend that you extensively test your program in the CAEN environment prior to submission. More details regarding this will be covered in lab.

Successfully Completing the Projects

We have found through many years of teaching experience that the most common reason for poor project performance is not starting early enough. Plan to do some work on the project every day and try to have it finished a few days ahead of the due date, because many unexpected problems arise during testing. In addition, the computing sites can become very crowded, making it difficult to get a computer to use, so plan for these things to happen.

The second most common reason for not doing well on the projects is not asking for help when you need it. We offer help in office hours and on the class forums. When you come to office hours, please be ready to provide access to your code, preferably electronic. Another good way to get help is to post a question to the course forum. Remember, if you find that you are stuck on a piece of your project for an undue amount of time, please see us!

One major goal of this course is for you to learn to test and debug your programs independently. We will not debug them for you. Instead, we will help you try to figure out how to test and debug your program yourself. We will also ask you to demonstrate what testing and debugging techniques you have already tried, and what the results were, before providing any advice.

Finally, always make multiple backup copies of your work! If you somehow lose your work, it is your responsibility.

Project Submission Policies

All projects must be turned in directly to your engr101 AFS course space unless otherwise announced. Each project will include detailed instructions as to how the project should be named and where it should be filed. The details of how to do this will be presented in your lab section. All projects must be submitted by 9pm on the due date, unless otherwise noted. Projects that are filed incorrectly or named improperly will be treated as not having been submitted. Shortly after the due time your submission will be rendered unalterable. Attempting to modify your submission after the due time is cheating and is a violation of the Honor Code. Such file alterations are quite easy to detect.

Academic Integrity

You may complete programming projects either alone or with a partner. All programming projects in this course are to be done by you or your partnership.

You may give or receive help on any of the concepts covered in lecture, discussion, or the textbook, and

on the specifics of C/C++ syntax. You are allowed to consult with other students in the class to help you understand the project specification (the definition of the problem).

You may not collaborate in any way with people outside your partnership when constructing your solution or to collaborate to the extent that your programs are identifiably similar; your partnership working alone must generate the solution to a programming project. If you have any questions as to what constitutes unacceptable collaboration, please talk to the instructor right away.

Working in a partnership is optional. If you work in a partnership, ***you and your partner must still submit your projects individually***. Write both of your usernames in the comments at the top of every file to avoid a potential referral to the Honor Council. You cannot change partners in the middle of one project, unless your partner drops the course. You may change partners only after a project is completed and submitted. However, you are free to work individually as much as you like or collaborate as much as you like, as long as it is with your partner.

You are expected to exercise reasonable precautions in protecting your own work. Do not let other students borrow your account or computer, do not leave your program in publicly accessible directory, and take care when discarding printouts.

We report suspected violations to the Engineering Honor Council. To identify violations, we use both manual inspection and automated software to compare present solutions with each other and with prior solutions. The Honor Council determines whether a violation of academic standards has occurred, as well as any sanctions. Read the Honor Code for detailed definitions of cheating, plagiarism, and other forms of academic misconduct.

All exam work must be carried out completely on your own. You will be asked to sign the Honor Code on all exams.

Guidelines for Partnerships

The following guidelines apply to those who choose to work in a partnership:

DOs

- Do READ THESE GUIDELINES CAREFULLY before programming with another student. You must follow these guidelines, or risk being investigated for an Honor Code Violation.
- Do choose a partner from the current semester of this course.
- Do try to find a partner in the same lab section in which you are enrolled.
- Do put both your username and the username of your partner in the comments at the top of all code files. This is important to avoid referral to the honor council.
- Do submit your own copy of the project (i.e. each person in a partnership must submit his/her own copy).

DON'Ts

- Do not program with someone without understanding these guidelines.
- Do not partner on an project with someone who has already solved the problem. Students who do this will not learn as much as those who partner with someone at a similar skill level.
- Do not share code with anyone other than your partner or a staff member.

- Do not split the work in half. It is important that both partners work on all parts of the program. Splitting the work may harm your or your partner's understanding of that part of the solution.
- Do not partner with anyone who is not currently enrolled in the course.

Grading Scheme

Your grade in this class will consist of your performance on programming projects, exams, and your lab section participation. It is important to recognize that at the University of Michigan, a “C” grade represents adequate performance in the class, “B” means very good performance, and “A” represents high accomplishment in mastering the material. Grades will not be curved: your grade depends solely on your own work, regardless of the performance of your peers. Our goal in teaching this class is to provide each student with all the resources necessary to master the material and earn an A.

Late submissions will not be accepted. We do not generally offer extensions. For example, we do not offer extensions due to crowded computing sites, internet access problems, accidental erasure or loss of files, or outside conflicting commitments.

We will consider extension requests made in person and at least 2 weeks in advance, for example, for religious holidays. Additionally, we will consider requests for documented, unanticipated medical or personal emergencies. If you can't see the instructor in advance due to the emergency, then see him/her as soon as you possibly can. In all cases, we require to written proof of the emergency.

The following tables show how each component is weighted with respect to your final grade, and how numerical grades correspond to letter grades:

Item	Percentage of final grade
Labs	15 %
In-class exercises	15 %
Projects	30 %
Exams	40 %

Numerical Grade (%)	Letter Grade
$97 \leq \% \leq 100$	A+
$92 \leq \% < 97$	A
$89 \leq \% < 92$	A-
$86 \leq \% < 89$	B+
$82 \leq \% < 86$	B
$79 \leq \% < 82$	B-
$76 \leq \% < 79$	C+
$72 \leq \% < 76$	C
$69 \leq \% < 72$	C-
$66 \leq \% < 69$	D+
$62 \leq \% < 66$	D
$60 \leq \% < 62$	D-
$\% < 60$	E

Regrade Policy

In general, regrade requests must be submitted within one week after the project or exam score is returned to you, unless otherwise noted. A more detailed description of how regrade requests work is posted in a separate document on Canvas entitled “Formal Regrade Request Policies”.

Exams

If you have an unresolvable scheduling conflict for an exam, you may be able to take the exam at another time. However, you must have a sufficient reason and will need to request instructor approval at least one week in advance. Instructions on how to request an alternate exam will be announced on Canvas.

Exam 1:

Covering Lectures 1–12

Time: See our Course Schedule on Canvas

Location: See our Announcements on Canvas later in the semester

Exam 2:

Mainly covering Lectures 13–25

Time: See our Course Schedule on Canvas

Location: See our Announcements on Canvas later in the semester

If you have a **MAJOR** academic conflict with one of the two exams, please fill out the following form as soon as possible:

<https://goo.gl/K83PBq>

Accommodations for Students with Disabilities

If you think you need an accommodation for a disability, please let your instructor know **at the beginning of the semester or at least 2 weeks in advance**. Some aspects of this course may be modified to facilitate your participation and progress. As soon as you make us aware of your needs, we can work with the Services for Students with Disabilities (SSD) office to help us determine appropriate academic accommodations. SSD (734-763-3000; <http://ssd.umich.edu>) typically recommends accommodations through a Verified Individualized Services and Accommodations (VISA) form. Any information you provide is private and confidential and will be treated as such.

To request accommodations for the midterm and final exams, please do the following:

1. Fill out this form: <https://goo.gl/49D1qy>
2. Scan and email a copy of your SSD form to Michael Benson (mlbenson@umich.edu) . Hard

copies **will not** be accepted. Scanners can be found in all University libraries as well as most CAEN labs on north campus.