Fall 2023 ME 021: Engineering Computing

Instructor: Ayush Pandey (meet with me 1-on-1 using this URL)

Email: ayushpandey@ucmerced.edu Office: SE2 381

Teaching Assistants:

1. Hassan Jafari Mosleh (Lab 02L, Lab 03L, Lab 04L)

   Email: hjafarimosleh@ucmerced.edu

2. Amir Kouravand (Lab 05L, Lab 06L, Lab 07L)

   Email: akouravand@ucmerced.edu

3. Hansong Lee (Lab 08L)

   Email: hlee287@ucmerced.edu

4. Alex Frias (Project mentor and course reader)

   Email: afrias5@ucmerced.edu

🏛️Class Discussions: Mon from 1.30pm to 3.20pm at COB2 110

📓Ayush's office hour:

- Mon 3.30pm to 4.30pm in SE2 lobby
- Tue 12pm to 1pm on Zoom
- Tue 3:00pm to 4:00pm in SE2 381
- Thu 1:00pm to 2:00pm in SE2 381 or Zoom.

👩Alex's office hour: Fri 12pm - 2pm. Location SE2 lobby.

💻Labs:

- 02L on Wed 7.30am to 10.20am at Kolligian 208 (Hassan)
- 03L on Fri 7.30am to 10.20am at SSM 154 (Hassan)
- 04L on Wed 7.30pm to 10.20pm at Kolligian 208 (Hassan)
- 05L on Tue 10.30am to 1.20pm at SSM 154 (Amir)
- 06L on Thu 10.30am to 1.20pm at SSM 154 (Amir)

- 07L on Fri 7.30pm to 10.20pm at Kolligian 208 (Amir)
- 08L on Wed 4.30pm to 7.20pm at SSM 154 (Hansong)

🙋🖥️Alex's project mentoring sessions:

- 02L on Wed 9.20am to 10.20am at Kolligian 208
- 03L on Fri 9.20am to 10.20am at SSM 154
- 04L on Wed 8.30pm to 9.30pm at Kolligian 208
- 05L on Tue 11.50pm to 12.50pm at SSM 154
- 06L on Thu 11.50pm to 12.50pm at SSM 154
- 07L on Fri 8.20pm to 9.20pm at Kolligian 208
- 08L on Wed 5.35pm to 6.35pm at SSM 154

Too long; didn't read summary of the syllabus:

1. You will learn Python and MATLAB programming in this course and will be able to develop functioning and meaningful programs by the end of the course.
2. CatCourses is your best friend for all deadlines and announcements. Bring your laptop to the class to participate in the learning together with everyone.
3. Contact your lab TA for extension requests and feel free to ask for help in labs and office hours! We are here to support your learning and ensure that you can excel in Python and MATLAB programming.

Course prerequisites:

All UC Merced students are welcome! There are no course pre-requisites to take this class and excel in it. It is expected that all students are comfortable with middle school algebra and in using computers. This is intentionally defined in a subjective manner to welcome students from all backgrounds and experiences. This course serves as a training platform for all students interested in learning about computing and computer programming. This course will actively attempt to create a welcoming climate for everyone without relying on prior computer programming experience. If needed, feel free to discuss your preparation with the instructor or the course staff.

Course learning goals and outcomes:

By the end of this course, you will be able to demonstrate computer programming skills with Python and MATLAB. You will be able to create custom programs, investigate errors in existing programs independently, and create fully functioning applications.

Course overview:

This course is your first introduction to "computing" in engineering, "MATLAB", and "Python" — the computer programming languages. The topics in this course are some of the most fundamental things that you will ever learn! Computing and programming literacy is increasingly becoming one of the most desirable skills, no matter what kind of education or job you are involved with in the future.

The philosophy of the course is grounded in the computing for all mission (https://www.csforall.org/, https://www.nsf.gov/news/special_reports/csed/csforall.jsp). The course staff will strive to foster learning opportunities such that every student can achieve an A in this course! The assessments will be designed in a way that this is possible and there will be no grading on a curve. The course staff will try their best to support your learning throughout the semester to lower the barriers in learning computer programming.

Catalog description:

Introduction to Matlab and a line programming language including FORTRAN, PYTHON, and C++. Concepts of formatted input/output, data types, variables, arrays, strings, variable scopes, logic statements, loops, functions and subroutines, and data graphing. Computing examples are drawn from mechanical engineering topics including linear algebraic equations and root search.

Course policy and expectations:

1. **Lectures** will be designed as class discussions where you will be able to learn most of the key concepts required to complete the assignments successfully. So, please bring a device that you can use to write and run code (a laptop).

2. **Lab attendance** is not mandatory but you are encouraged to use the lab time to complete the assignments of the week with the TA's help.

3. Try to make the best of the **office hours**, we are here to answer your questions in the office hours.

4. If you ever need an **extension**, contact your lab TA. All legitimate extension requests will be granted for lab assignments. The TA can decide on the length of extension based on the individual circumstances. Late work is still eligible for a 50% partial credit (no matter how late). You can also use extra credit activities to make up for lost points.

5. **Academic honesty:** You are expected to write your own code and most importantly you must understand all parts of your submissions. Asking for help in understanding a concept, learning from discussions online, and asking ChatGPT for explanations are all OK — but, you must write your own code after you have understood the concepts. Remember that learning computer programming will enhance your future education and career, so copying code that you did not write will impede your learning and progress. With the online textbook platform that we have, it is easy for us to find out instances when you did not write your own code, so if such incidents occur you may receive a note from the course staff. The full university academic honesty policy PDF is posted on CatCourses. Please review that PDF.

6. **Class Conduct and Community:** Remember that contributions from each of us can help in building a respectful, courteous, and an intellectually stimulating class environment. Language or behavior that prevents any student to participate fully in class is not acceptable. It is important to remain open to each other's thinking and engage in rigorous, challenging discussion about issues of shared concern. This is distinct from participating in oppressive behaviors — racism, sexism, homophobia, transphobia, and ableism, that are designed to keep people out of conversations, not bring them in.

7. UC Merced is committed to providing an **equal opportunity environment** for all students and employees that remains free of all forms of discrimination, harassment, and exploitation. Discrimination and harassment based on race, color, religion, creed, sex, national origin, age, disability, veteran status, or sexual orientation is a violation of state and federal law and/or University policy and will not be tolerated. Retaliation against any person who complains about discrimination is also prohibited. If you witness or experience any form of harassment, please seek support and guidance. For more information, please visit https://ophd.ucmerced.edu/

8. **Emailing/Direct messaging on CatCourses:** If the syllabus does not answer your question, please reach out to the TA in the lab section or the office hour. If your question needs urgent attention, you can send an email to the TA or the instructor (make your best judgement on who might answer your question best and quickest). Please **mention the course number in the subject** of any email you send.

Assessment structure:

1. All in-class activities and quizzes are graded for participation – 5% of your total grade.
2. All reading assignments (knowledge checks) are graded for correct completion on zyBooks (you have unlimited attempts to do these). Two lowest scoring knowledge check assignments will be dropped from final grade computation. zyBooks will provide instant feedback on errors and mistakes in your code – 15%
3. All lab assignments require more independent effort than the reading assignments and are graded for correctness – 30%. Two lowest scoring lab assignments will be dropped from the final grade computation.
4. Your summative assessment consists of two independent projects. Each project is worth 25% of your final grade with a total of 10 project milestones.
5. Project grading structure: 30% from milestones (1 lowest scoring milestone will be dropped) + 40% code submission + 30% demonstration. (0 on project if it is not demonstrated).
6. You can use extra credit activities to earn a maximum of 5% toward your total grade.
7. Grading scheme according to final score (out of 105, because of 5 extra credit points):
    a. Score in [96, 105] is A+, [92, 96) is A, and [88, 92) is A-
    b. Score in [84, 88) is B+, [80, 84) is B, and [75, 80) is B-
    c. Score in [70, 75) is C+, [66, 70) is C, and [62, 66) is C-
    d. Score in [58, 62) is D+, [54, 58) is D, and [50, 54) is D-

Python project guidelines:

All students must work on an independent Python project. This project must demonstrate the following five key elements of Python that you learn in ME 021. An ideal project that grades 100% will

1. Control the program flow with branching and loops
2. Uses correct data structures for optimal computations
3. Uses functions for modular code
4. Loads data from files (real-world data is preferable) and writes outputs to files (if needed)
5. Documents the flow of logic with comments, docstrings, and user-friendly messages

Project ideas document: PDF

This is a document with potential Python project ideas. You may choose an idea from this document but it is not required that you choose from the list. You are free to choose your own idea for the project.

MATLAB project guidelines:

All students must work on an engineering computing project in MATLAB for their final project. Four options are provided for the project. Every student must **choose one** of the following projects:

1. Design and analysis of materials
2. Decoding images using matrix computations
3. Thermodynamic cost of heating and cooling
4. GUI design for engineering design and analysis

Each project has a computational, and a visualization component along with design questions to explore. This is the final course project and it must demonstrate the application of computing to an engineering problem. An ideal project that grades 100% will

1. Precise control of logic (using if-else/for/while/arrays/functions, as necessary)
2. Accurate computation of engineering metrics using mathematical equations.
3. Visualizations and interface design that is suitable for an engineering design project.
4. Exploration of at least one open-ended idea related to engineering design.
5. Articulating the engineering design and analysis in a technical report.

The course topics for each week are described in the table below. The column for supplemental reading will be continually updated. Other changes to the syllabus may be made but I will announce to the class if something major changes.

Date	Weeks	Topics
Aug 28	Week 1	Introduction to Computing, basics of Python programming, inputs, outputs, errors, variables, and expressions.
~~Sep 4, Labor day~~	~~Week 2~~	Common data types: strings, lists, integers Branching and if statements.
Sep 11	Week 3	Loops (while loop and for loop, nested loop, `break`, and `continue`)
Sep 18	Week 4	Data types and string operations
Sep 25	Week 5	Lists and dictionaries
Oct 2	Week 6	File handling and exceptions
Oct 9	Week 7	Functions and modules
Oct 16	Week 8	Project discussions
Oct 23	Week 9	Introduction to MATLAB
Oct 30	Week 10	Scripts and functions in MATLAB
Nov 6	Week 11	Branching
Nov 13	Week 12	Loops
Nov 20	Week 13	Arrays
Nov 27	Week 14	Data visualization and GUI
Dec 4	Week 15	Intro to data analysis and problem-solving with Python and MATLAB
Dec 11	Week 16	–