San José State University
College of Engineering
Department of Mechanical Engineering

ME 30 Computer Applications, Fall 2021

Course and Contact Information

Instructor:	Dr. Salman Ahsan
Email:	Salman.Ahsan@sjsu.edu
TAs:	Ryan Shu, Suresh Goud Mula
Email:	Ryan.Shu@sjsu.edu, SureshGoud.Mula@sjsu.edu
Grader:	Bharadwaj Jagannath, Bharadwaj.Jagannath@sjsu.edu
Hardware Guru:	Eric Wertz
Email:	eric@edushields.com
Office Hours:	Friday 1700 (Tentative)
Class Days/Time:	Monday 0800 – 0850
Classroom:	ENG 189
Laboratory:	ENG 213 or ENG 407

Course Format
  Technology Requirements

As a technology intensive course, ME30 requires regular access to the Internet and access to a personal computer. Access to a computer is required and instruction will primarily target the Microsoft Windows Operating System, though some help may be provided for the Apple OS as well. Students with experience in Linux are welcome to utilize that platform, though even less technical support will be available.

Students without a personal computer, please contact the Mechanical Engineering department to see if the department or the university can arrange for a loaner.

  ME30 will utilize the following programs:
  - Anaconda (Python 3.x, Jupyter Lab)
  - Mu (CircuitPython)
  - Adobe Acrobat (Converting files to PDF for submission)

Each student will be required to obtain a Circuit Playground Express (CPX) board from AdaFruit: https://www.adafruit.com/product/3333

Faculty Web Page and Messaging

Course material, such as the syllabus, handouts, copies of the slides, etc. may be found on Canvas, http://sjsu.instructure.com.

Assignments shall be turned into Canvas. Course announcements will also be distributed from the Canvas system. You are required to check Canvas regularly to keep abreast of the course announcements and any changes to the schedule.

Course Description

Using a computer to solve engineering problems through programming and the use of engineering application procedures. Use of procedural and informational problem-solving methods and practices applied to software design, application, programming and testing. Lecture 1 hour/lab 3 hours. 2 units.

Course Goals and Learning Objectives

The goals of this course are to help you:
- o Understand how mechanical engineers can and do use computers to solve engineering problems
- o Learn how to solve engineering problems using computational methods
- o Get experience in developing algorithms for effectively solving problems using computers
- o Gain familiarity with well-known software libraries that are widely used by mechanical engineers to solve analytical and numerical problems
- o Prepare for subsequent courses and industry practice which involve computation to solve engineering problems
- o Your ability to write code to solve simple problems is the KEY performance indicator in this class

Course Learning Objectives

The student who successfully completes the course will be able to:

1. Code
- 1.1 Given a problem statement, be able to generate pseudocode that outlines the solution to the problem
- 1.2 From pseudocode write a Python program with correct syntax
- 1.3 Take a block of Python code with errors, find the errors, fix them and make it work. This is called debugging

2. Programming Methodology
- 2.1 Apply basic good programming practices – such as, short but descriptive variable names, developing an algorithm through the use of pseudocode, etc.
- 2.2 Write programs that are sufficiently documented so that colleagues can understand their operation

3. Applications of Programming Principles to Physical Computing
- 3.1 Given a problem statement, write Python code to interact with the physical world outside the computer that fulfils the problem specifications

Recommended Textbooks
We will not follow any textbook, but you'll find the following reference texts helpful.

How to Think Like a Computer Scientist: Learning with Python 3 Documentation Release *3rd Edition*, Peter Wentworth, Jeffrey Elkner, Allen B. Downey and Chris Meyers.
https://buildmedia.readthedocs.org/media/pdf/howtothink/latest/howtothink.pdf

Automate the Boring Stuff with Python *2nd Edition,* Al Sweigart:
https://automatetheboringstuff.com/

Both of these textbooks are available for free download on the websites listed above.

Additional References

http://www.learnpython.org/
https://www.tutorialspoint.com/python/index.htm
https://jakevdp.github.io/WhirlwindTourOfPython/
http://ibiblio.org/g2swap/byteofpython/read/index.html
https://en.wikibooks.org/wiki/Python_Programming
https://realpython.com/
https://www.python-course.eu/python3_course.php
https://www.datacamp.com/community/tutorials/tutorial-jupyter-notebook

Software
- Anaconda (https://www.anaconda.com/download/)
- Mu (https://codewith.mu/en/download)

Library Liaison

Our liaison to the University Library is Krista Anandakuttan krista.anandakuttan@sjsu.edu. Krista can help you make optimum use of information resources available to you through the University Library.

Assignments and Grading Policy

Assessment for the purposes of determining your course grade will consist of evaluating your performance on homework/lab assignments, a project, quizzes, a midterm exam, and a final examination. Quizzes will take place on a weekly basis. Homeworks, quizzes and exams will all be administered through Canvas, unless otherwise noted. Consult Canvas for homework due dates. All of your work must be submitted in softcopy form (via the Canvas course management system) by the due date.

The penalty for late homework submission will be calculated by the Canvas system as follows:
- Submitted before the due date/time: No penalty
- Up to 24 hours late: 20% penalty
- From 24 to 48 hours late: 40% penalty
- From 48 to 72 hours late: 60% penalty
- From 72 to 96 hours late: 80% penalty
- More than 96 hours late: No credit will be given!

Keep in mind that Canvas keeps track of the time when homework is uploaded. If the due time is 11:59PM, and your submission finishes uploading at 12:00AM, the Canvas system will mark it as late and you will automatically be docked 20% of your points, despite the fact that you were only seconds late. So, take into account unexpected delays such as slow internet speeds which is the most common cause for such delays.

There will be no make-up assignments or quizzes.

If you cannot take the midterm or the final at the announced times, contact the instructor at least two weeks BEFORE the scheduled examination date. If you have a valid reason, I will try to find a way to accommodate you.

IMPORTANT NOTE! Make it a point to turn in something for every assignment, whether you have completed it or not.

Waiving the Late Penalty

Under special circumstances, such as sickness or family emergencies, the late penalty can be waived once, provided the instructor is informed at least 24 hours before the homework due time. Any subsequent requests have to be accompanied by evidence such as a note from the doctor. No work, under any circumstances, will be accepted 96 hours after the original due date/time.

Weighting of Course Components

Exam 50% (25% per exam), Homework/labs 20%, Project 20% and Quizzes 10%.

Criteria for Assigning Letter Grades

The scores on your homework, laboratory projects, quizzes and final examination will be normalized, combined and totaled using the weighting scheme described above. A final letter grade will be determined using statistical tools (curving). According to the table below:

Student's Standing on the Cumulative Grade Distribution	Letter Grade
Well above 1 sigma from the mean - exceptional performance. In some cases, no one will qualify for *A plus*	A plus
Above 1 sigma from the mean	A
The occasional case at the boundary between *A* and *B*. Depending on which distribution the student is closer to will determine *A negative* or *B plus*	A negative / B plus
Between the mean and one sigma above the mean	B
The occasional case at the boundary between B & C. Depending on which distribution the student is closer to will determine *B negative* or *C plus*	B negative / C plus
Between the mean and one sigma below the mean	C
The occasional case at the boundary between *C* and *D*	C negative
More than 1 sigma below the mean	D
More than 1 sigma below the mean and not showing basic competence, often students with F do not turn in more than 50% of the course deliverables	F

Final Project

Details will be announced roughly half way through the semester.

University Policies Relevant to Students Taking This Class

https://www.sjsu.edu/curriculum/courses/syllabus-info.php

ME 30 Course Schedule

The schedule below is a reasonable estimate of what will take place in the course and when. Make sure you turn on Canvas notifications, so whenever I (or the TAs) post an announcement, you get an email alert.

Notes

1. Reading assignments complement the lecture and allow you to quickly navigate to the chapter(s) in the textbooks that were covered in class. Readings are not mandatory and should be treated as a resource, among all other online resources.

2. Following each lecture, I highly recommend that you *review* any notes you took in lecture along with the material I upload on Canvas. Read back through your notes, and fill in any gaps using the associated chapters in the textbooks or by doing a web search. There is a lot of Python material available online. Write down any questions you have in the margins of your notes. Be sure to come to office hours, or ask about your questions in class or the lab sections.

ME 30 Lecture and Homework Schedule

The tentative schedule below is an overview of what you will be learning each week, but **always defer to the assignment due dates posted in Canvas.** If there are changes, I'll make the updates in Canvas. This document is not always updated in real time.

Tentative Schedule

Week	Lecture	Lecture Mon ENG189	Homework Submission Due Date	Suggested Readings (chapters) Automate	Think	Lab	Sec 2 RS M 0900-1145 ENG213	Sec 3 SM M 1330-1615 ENG407	Sec 4 RS T 1330-1615 ENG407	Sec 5 SM W 1330-1615 ENG407
						NO LAB				26-Jan
1	Intro to computers	31-Jan	11-Feb	1	1	Anaconda installation	31-Jan	31-Jan	1-Feb	2-Feb
2	Modes, variables and types	7-Feb	18-Feb	1	2	Install Mu & modes, var, types	7-Feb	7-Feb	8-Feb	9-Feb
3	Operators and CPX board	14-Feb	25-Feb	1	2	Operators and CPX board	14-Feb	14-Feb	15-Feb	16-Feb
4	Conditionals	21-Feb	4-Mar	2	3	Conditionals	21-Feb	21-Feb	22-Feb	23-Feb
5	Strings	28-Feb	11-Mar	6	4	Strings	28-Feb	28-Feb	1-Mar	2-Mar
6	Iteration	7-Mar	18-Mar	2	3	Iteration	7-Mar	7-Mar	8-Mar	9-Mar
7	Functions	14-Mar	25-Mar	3	4	Functions	14-Mar	14-Mar	15-Mar	16-Mar
	EXAM 1	21-Mar	9-May			Project Introduction	21-Mar	21-Mar	22-Mar	23-Mar
	SPRING BREAK	28-Mar				NO LAB	28-Mar	28-Mar	29-Mar	30-Mar
8	Recursion	4-Apr	15-Apr		10	Recursion	4-Apr	4-Apr	5-Apr	6-Apr
9	Lists and Tuples	11-Apr	22-Apr	4	3	Lists and Tuples	11-Apr	11-Apr	12-Apr	13-Apr
10	Dictionaries	18-Apr	29-Apr	5	5	Dictionaries	18-Apr	18-Apr	19-Apr	20-Apr
11	Object Oriented Programming	25-Apr	6-May	9	7	OOP, Project approval	25-Apr	25-Apr	26-Apr	27-Apr
12	Files	2-May			11	Work on project	2-May	2-May	3-May	4-May
13	Error Handling	9-May		11	12	Project Presentations	9-May	9-May	10-May	11-May
	EXAM 2	16-May								