

# ENGR 138: COMPUTER PROGRAMMING AND NUMERICAL METHODS FOR ENGINEERS

## Citrus College Course Outline of Record

Heading	Value
Effective Term:	Fall 2023
Credits:	4
Total Contact Hours:	108
Lecture Hours :	54
Lab Hours:	54
Hours Arranged:	0
Outside of Class Hours:	108
Total Student Learning Hours:	216
Prerequisite:	MATH 190 or higher.
Strongly Recommended:	ENGL 101.
Transferable to CSU:	Yes
Transferable to UC:	Yes - Approved
Grading Method:	Standard Letter

## Catalog Course Description

Engineering computation using MATLAB and VBA. Topics include programming fundamentals, matrix computation, statistical analysis, data graphing, data extraction from spreadsheets and solving mathematical problems using numerical methods. 54 lecture hours, 54 lab hours.

## Course Objectives

- Identify, formulate and solve computational problems.
- Design algorithms and flowcharts to facilitate programming.
- Create, test and debug programs using procedural and object oriented approaches.
- Write programs requiring subroutine call outs and importing of data from files.
- Write MATLAB programs to analyze data and generate tables, charts and graphs.
- Write VBA programs to assist in extracting information from a spreadsheet to analyze and generate tables, charts and graphs.

## Major Course Content

1. Introduction to computer programming
  - a. Basic computer programming concepts
  - b. Programing languages
  - c. Matlab language attributes
2. Linear algebra and array concepts
  - a. Linear equation applications and solutions
  - b. Definitions and operations of arrays
3. Problem-solving methods, scripting, inputs and outputs
  - a. Relational and logical operators, IF, IF-ELSE statements
  - b. IF Ladders, flowcharting

- c. FOR loops, series approximation and indexing
  - d. Nested FOR loops, vectorization and 3D plots
  - e. WHILE loops, newton and bisection methods
4. Problem solving applications
  - a. Interpolation, least squares regression, linearization
  - b. Heat transfer and stress applications
  - c. Solving 1st order ordinary differential equations, exponential growth
  - d. Monte Carlo simulation
5. Program organization and debugging
  - a. Strings
  - b. Cell arrays and structures
  - c. Sorting and searching
6. Introduction to Microsoft Excel VBA and macros
  - a. Spreadsheet low-level file input/output
  - b. Object-oriented programming, graphical objects
  - c. Graphical user interfaces
  - d. Exporting data to MATLAB from a spreadsheet
  - e. Importing data from MATLAB

## Lab Content

1. Writing and correcting expressions in Matlab
  - a. Working with arrays
  - b. Developing databases
  - c. Sorting lists
  - d. Plotting with advanced techniques
2. Problem solving applications
  - a. Solving circuits and statics problem
  - b. Analyzing projectile motion
  - c. Calculating areas and volumes of solids
  - d. Analyzing solid object stress data
  - e. Analyzing heat transfer data
  - f. Solving functions of one variable
3. Introduction to Excel VBA
  - a. Data input and extraction form spreadsheet
  - b. Writing Macros to process data
  - c. Developing simple graphic user interfaces (GUI)
  - d. Transfer spreadsheet data to MATLAB, process data and report results on graph or chart

## Suggested Reading Other Than Required Textbook

Engineering journals or periodicals from professional engineering organizations such as ASME, SAE, IEEE, ASCE, etc.

## Examples of Required Writing Assignments

Each section will have questions to answer from the textbooks in short answer form. e.g. Describe how a program can be written to cause an infinite loop to occur.

## Examples of Outside Assignments

Section practice problems: - Given the matrix representation for the planar circuit, solve the matrix. Apply your technique to the given DC circuit. Now modify your solution to deal with AC circuits of variable frequency. - Enter the following list into an Excel spreadsheet. Write a program in VBA to divide the Velocity, Time and the 'X' and 'Y' distances into arrays on a separate sheet. Write a MATLAB program that will read the divided data and plot the data as a graph representing the projectile's position with respect to time and velocity.

## Instruction Type(s)

Lab, Lecture, Online Education Lab, Online Education Lecture