CE 108 Introduction to Computer Methods in Civil Eng. 2 Units

USC | SONNY ASTANI DEPARTMENT OF CIVIL AND ENVIRONMENTAL ENGINEERING

ABET Course Syllabus

Course Information, Textbook and Supplementary Materials

Course Description: Computer programming, organization of problems for computational solution, flow charts, programming; numerical methods; analysis and solution of civil engineering problems.

Required for: All Civil and Environmental Engineering undergraduate degree programs

Prerequisites: Sufficient mathematics background

Co-Requisite: None

Required Textbooks:

- *The C Programming Language*, Second Edition, by Brian W. Kernighan and Dennis M. Ritchie, Prentice Hall, ISBN 0-13-110362-8.
- *FORTRAN 77 for Engineers and Scientists with an Introduction to FORTRAN 90*, Fourth Edition,1996, by Larry Nyhoff and Sanford Leestma, Macmillen,2012, ISBN 0-02-388631-5.
- *Matlab for Engineering Applications*, by William J. Palm III, McGraw-Hill, ISBN 0-07-047330-7.
- *Mathematica: A Practical Approach,* by Nancy Blachman, Prentice-Hall, ISBN 0-13-563826-7.

Topics Covered	Learning Outcomes
▪ Introduction to USC computer systems, EMACS and PICO editors ▪ Procedural languages (C, Fortran) and basic Input / Output (I/O) statements ▪ Basic data types and algebraic expressions ▪ File Input / Output (I/O) and logical expressions ▪ Loops and repetitive execution ▪ Iterative algorithms in applied mathematics ▪ Arrays and pointers ▪ Linear Algebra applications ▪ Solution of matrix equations	1. Competency in the fundamentals of programming in MATLAB® ▪ Navigate folder and files ▪ Understand types of numbers (integer and floating point), character strings, logical (Booleans), etc. ▪ Understand the concept of a computer algorithm, flow charts ▪ Perform arithmetic and logical expressions using a computer program ▪ Perform basic input/output operations for reading and writing data to/from a file ▪ Understand the concepts of data structures and cell arrays ▪ Perform basic plotting and graphing functions. ▪ Use vectors and matrices to store and retrieve data ▪ Develop functions and sub-functions to perform some task ▪ Use conditional statements to handle optional parts of an algorithm ▪ Use loops to perform repeated tasks ▪ Use the built-in debugger to follow the flow of code and debug code ▪ Understand the concept of different workspaces for each function as well as base and global workspaces ▪ Understand how to use recursion to solve numerical problems ▪ Understand the limited precision of floating point math and how it impacts numerical solutions. ▪ Use inline functions and function handles to pass arbitrary (anonymous) functions to another function 2. Ability to design algorithms for solving problems in civil engineering. ▪ Develop a numerical algorithm from a set of given instructions or procedures ▪ Debug an existing numerical algorithm ▪ Write code in a style that is easily humanly readable

▪ Programming with functions ▪ Introduction to Object Oriented Programming (C++, Visual Basic)	3. Understand numerical algorithms: sorting, roots of equations, numerical integration, and solving linear equations ▪ Sort numbers ▪ Find the roots of an equation ▪ Perform numerical integration ▪ Solve a set of linear equations in multiple unknowns

CE 108 Introduction to Computer Methods in Civil Eng. 2 Units

USC | SONNY ASTANI DEPARTMENT OF CIVIL AND ENVIRONMENTAL ENGINEERING

Lecture and Lab Schedule			
Lecture		**Lab**	
Sessions per Week	Duration per Session	Sessions per Week	Duration per Session
1	1 hour	1	1 hour

Relation of Course Objectives to Program Outcomes

The Civil Engineering program is designed to teach beyond the technical content of the curriculum and prepare the students to utilize what they learn in a professional setting.

This course contributes to the program outcomes as outlined in the adjacent table.

Course Contribution to Program Outcomes (a-k)	Key ✓
k. An ability to use the techniques, skills, and modern engineering tools necessary for engineering practice.	

Course Coordinator: Dr. Erik Johnson
Professor of Civil Engineering

Revised: Fall 2014