University of Arkansas-Fort Smith
5210 Grand Avenue
P. O. Box 3649
Fort Smith, AR 72913-3649
479-788-7000

General Syllabus:

ITC 1374 Programming for Engineers

Credit Hours: 4   Lecture Hours: 4   Laboratory Hours: 0

Prerequisite: MATH 1403 College Algebra or higher MATH

Effective: 2018-2019

I.   Course Information:

    A. Catalog Description
        A course for students majoring in engineering. Topics include data
        representation, high-level languages, looping, functions, arrays, pointers, and
        an introduction to the Linux operating system shell.

II.   Student Learning Outcomes

    A. Subject Matter:
        Upon completion of this course, the student will be able to:

        1. Design solutions to problems using computer programs.
        2. Demonstrate the ability to create and debug computer programs written in
            the C Language.
        3. Describe the mechanics of running, debugging, and testing programs in
            the Linux environment.
        4. Use the Linux shell to edit, debug, and test programs.
        5. Explain the basic types of commands provided by the Linux shell.

    B. University Learning Outcomes:
        The Programming for Engineers course material enhances student abilities in
        the following areas:

        Analytical Skills
        Critical Thinking Skills: Students will identify a problem, break it down into
        its component parts and develop a solution in the C language.

III. Major Course Topics

A. Linux Shell Commands – command line control commands
B. Creating and compiling C programs in Linux –
C. Using Native Linux commands to Create and Compile C programs
D. Program Debugging
E. Determine compile and runtime errors using logic and critical thinking skills
F. Data Representation
G. How numerical values and Strings are structured in C
H. Control Structures
I. Designing programs with decision, iteration
J. Functions and Methods
K. Designing program organization and modularity in C
L. Arrays
M. Creating and working with consecutive like information using indexes.
N. Pointers
O. Using pointers to process variables
P. Using pointers to accomplish "pass by reference" with functions
Q. Using pointers to process arrays instead of indexes with both static and dynamic memory
R. String handling