CSC 130: The World of Computer Science

Catalog Description:
A broad overview of computer science, with topics ranging from the basic structure of a computer to artificial intelligence. Students will use a high-level language to investigate and implement solutions to problems in a range of fields. Suitable for non-majors who want to learn more about computer science. 3(3-0) F,S

There are no prerequisites for this course
This course is a prerequisite for: CSC 131, Computational Thinking

Required Text: *Fundamentals of Python: First Programs*, by Kenneth Lambert, Course Technology, 2012, http://home.wlu.edu/~lambertk/python/cs1python/index.html.

Major Topics (including information for course sequence or transition)

1. Image, sound, and text data representations
2. Binary number representation and conversions
3. Algorithms and Python programming, specifically:
   - **a.** Variables and types—integer, float, string, boolean
   - **b.** Conditional statements—if, else, elif
   - **c.** Loops—for and while
   - **d.** Functions
   - **e.** Data structures: lists and dictionaries
4. Applications: image processing, data mining, etc
5. Computing for the public good

Student Outcomes Assessed in CSC 130

A. Students will attain an ability to apply knowledge of computing and mathematics appropriate to the discipline
E. Students will attain an understanding of professional, ethical, legal, security and social issues and responsibilities
I. Students will attain an ability to use current techniques, skills, and tools necessary for computing practice

CAC Characteristics Enabled But Not Assessed in CSC 130

B. Students will attain an ability to analyze a problem, and identify and define the computing requirements appropriate to its solution
C. Students will attain an ability to design, implement, and evaluate a computer-based system, process, component, or program to meet desired needs

Table 1. Student Outcomes assessed by CSC 130

CSC 130 Student Outcomes	CSC 130 Performance Indicators	CSC 130 Assessment Goals
CSC 130 contributes to SO A: Students will attain an ability to apply knowledge of computing and mathematics appropriate to the discipline	**PI 130-1a:** Convert a binary number to decimal **PI 130-1b:** Convert a hexadecimal number to binary	**PI 130-1a:** >= 80% correct **PI 130-ab:** >= 80% correct
	PI 130-2a: Determine the amount of memory needed to store an RGB image using *n* bits per color channel **PI 130-2b:** Determine which character code is appropriate for storing a given set of characters	**PI 130-2a:** >= 70% correct **PI 130-2b:** >= 80% correct
	PI 130-3: Determine the maximum number of comparisons needed to carry out a binary search over *n* items	**PI 130-3:** >= 80% correct
CSC 130 contributes to SO E: Students will attain an understanding of professional, ethical, legal, security and social issues and responsibilities	**PI 130-4:** Identify an application that serves the public good	**PI 130-4:** >= 90% correct
CSC 130 contributes to SO I: Students will attain an ability to use current techniques, skills, and tools necessary for computing practice	**PI 130-5:** Determine when to use a while loop rather than a for loop	**PI 130-5:** >= 80% correct

Table 2. CAC Characteristics enabled by CSC 130

CSC 130 CAC Characteristics	CSC 130 Characteristics Enablers
CSC 130 enables Characteristic B: Students will attain an ability to analyze a problem, and identify and define the computing requirements appropriate to its solution	**CE 130-1.** Students write programs with increasing levels of complexity: use a function with no arguments to print name; use a function to print its argument; use variables to calculate and print shooting percentages for three NBA teams; use a dictionary to calculate and print the shooting percentages for all NBA teams.
CSC 130 enables Characteristic C: Students will attain an ability to design, implement, and evaluate a computer-based system, process, component, or program to meet desired needs	**CE 130-2.** Students write programs designed by the instructor to meet given specifications: given the function calls and designs, write code to count and display the counts of each pronoun in a text file; given the function calls and designs, write code to count and display the total number of medals won by countries at summer Olympics between 1896 and 2008.