Course Outline

COURSE: CSIS 43  DIVISION: 50  ALSO LISTED AS:

TERM EFFECTIVE: Fall 2018  CURRICULUM APPROVAL DATE: 05/14/2018

SHORT TITLE: C PROGRAMMING
LONG TITLE: C Programming

<table>
<thead>
<tr>
<th>Units</th>
<th>Number of Weeks</th>
<th>Lecture</th>
<th>Lab</th>
<th>Other</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>18</td>
<td>3</td>
<td>3</td>
<td>0</td>
<td>6</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Lecture</td>
<td></td>
<td></td>
<td>54</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Lab</td>
<td></td>
<td></td>
<td>54</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Other</td>
<td></td>
<td></td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>108</td>
</tr>
</tbody>
</table>

COURSE DESCRIPTION:

This course introduces computer programming using the C programming language. Topics include variable and constant declarations, arithmetic operations, selection, input/output operations, repetition, functions and recursion, arrays, pointers, and other related topics. This course has the option of a letter grade or pass/no pass. ADVISORY: CSIS 10 BASIC Programming, CSIS 42 Python Programming, or other programming experience.

PREREQUISITES:

COREQUISITES:

CREDIT STATUS: D - Credit - Degree Applicable

GRADING MODES

L - Standard Letter Grade
P - Pass/No Pass

REPEATABILITY: N - Course may not be repeated

SCHEDULE TYPES:

02 - Lecture and/or discussion
03 - Lecture/Laboratory
04 - Laboratory/Studio/Activity
047 - Laboratory - LEH 0.7
05 - Hybrid
72 - Dist. Ed Internet Delayed

5/22/2018
STUDENT LEARNING OUTCOMES:
1. Create algorithms to solve simple programming problems.
   Measure of assessment: Homework, projects, lab exercises
   Year assessed, or planned year of assessment: 2018
   Semester: Spring
2. Describe and employ strategies that are useful in debugging.
   Measure of assessment: Homework, projects, lab exercises
   Year assessed, or planned year of assessment: 2018
   Semester: Spring
3. Design, implement, test and debug short programs.
   Measure of assessment: Homework, projects, lab exercises, exam
   Year assessed, or planned year of assessment: 2018
   Semester: Spring

CONTENT, STUDENT PERFORMANCE OBJECTIVES, OUT-OF-CLASS ASSIGNMENTS
Curriculum Approval Date: 05/14/2018

TOPICS AND SCOPE:
8 Hours: Topic:
   Introduction to programming concepts.
   History of C language and UNIX.
   Parts of a program
   Procedural and object-oriented programming
   The programming environment:
   Compiling and executing a first program
Student Performance Objective:
   Modify, compile, and execute a sample program.
Lab and Homework:
   Practice modifying, compiling and executing more simple programs to become familiar with the user
   interface and the programming environment.
8 Hours: Topic:
   Simple variables: types, declaring, initializing.
   Types: integers, floats, and chars
   Declaring and using constants
   Types and use of comments.
   Arithmetic operators
Student Performance Objective:
   Students distinguish between different types of simple variables and use them appropriately.
   Students use simple arithmetic operators to perform calculations with variables.
   Students use comments to document their programs.
Lab and Homework:
   Design, modify, compile and execute short programs that incorporate these concepts.
8 Hours: Topic:
   Numeric expressions: integers, reals, characters, strings, and
   booleans.
   Arithmetic: operations, operator precedence, use of parentheses.
Operators: binary vs. unary
Typecasts and combined assignment operators

Lab and Homework:
Design, modify, compile and execute short programs that incorporate these concepts.

Student Performance Objective:
Students appropriately use all the
simple numeric variable types, including integers and reals.
Students describe differences between characters and character
strings, and utilize them as appropriate.

8 Hours: Topic:
Making decisions: if/then, if/then/else
Relational operators, precedence of operators
Nested if statements
Comparing numbers
Logical operators, precedence of operators

Lab and Homework:
Design, modify, compile and execute short programs that incorporate these concepts.

Student Performance Objective:
Students describe and use several types of decision statements.
Students can analyze and incorporate relational and logical operators into decision statements.
Students can write simple programs that use nested if statements.

9 Hours: Topic:
Simple for loops
Nested for loops
Infinite loops and loop errors

Lab and Homework:
Design, modify, compile and execute short programs that incorporate these concepts.

Student Performance Objective:
Students recognize when loops are necessary to repeat statements.
Students can recognize infinite loops and other loop errors.

8 Hours: Topic:
While loops, do/while loops, continue/next statements

Lab and Homework:
Design, modify, compile and execute short programs that incorporate these concepts.

Student Performance Objective:
Students compare and contrast for loops, while loops and do while loops and describe the uses of each.
Students can explain the difference between continue and next statements to escape loops.

9 Hours: Topic:
Simple functions, defining functions, calling functions
Local and global variables, scope of variables
Storage classes
Returning values from functions

Lab and Homework:
Design, modify, compile and execute short programs that incorporate these concepts.

Student Performance Objective:
Students can describe the advantages of organizing programs through the use of functions.
Students can describe the difference between local and global variables and the uses of each. Students can compare different methods of passing values into functions and different ways of returning values from functions.

8 Hours: Topic:
- Simple arrays: One dimensional arrays
- Character string arrays
- Types and uses of subscripts
- Multiple dimensional arrays
- Processing arrays with loops and nested loops.
- Arrays and functions

Lab and Homework:
- Design, modify, compile and execute short programs that incorporate these concepts.

Student Performance Objective:
- Students write programs that process numeric and string arrays using loops, and use several different types of subscripts.

8 Hours: Topic:
- Pointer variables
- Address of variables, vs. value of variable
- Initializing pointers, comparing pointers.
- Using pointers as function arguments

Lab and Homework:
- Design, modify, compile and execute short programs that incorporate these concepts.

Student Performance Objective:
- Students can define pointers and describe their uses.

8 Hours: Topic:
- Structures
- Defining structures and initializing structures
- Using structure members
- Arrays of structures and structures as function arguments
- Limitations of structures
- Unions

Lab and Homework:
- Design, modify, compile and execute short programs that incorporate these concepts.

Student Performance Objective:
- Students can compare and contrast structures and unions and describe the uses and limitations of each.

8 Hours: Topic:
- Pointers and arrays
- More on pointers
- Pointers as function parameters

Lab and Homework:
- Design, modify, compile and execute short programs that incorporate these concepts.

Student Performance Objective:
- Students can describe the uses of pointers in arrays and as function parameters, and the advantages of doing so.

8 Hours: Topic:
- Structures, pointers, and functions
- Using structures and pointers with functions

Lab and Homework:
Design, modify, compile and execute short programs that incorporate these concepts.

Student Performance Objective:
Students can describe the uses of pointers with structures.

8 Hours: Topic:
File operations
Reading and writing different types of files

Lab and Homework:
Design, modify, compile and execute short programs that incorporate these concepts.

Student Performance Objective:
Students describe the uses of programs that read and write files.

2 Hours Final exam.

METHODS OF INSTRUCTION:
Lecture, guided discovery, demonstration.

OUT OF CLASS ASSIGNMENTS:
Required Outside Hours: 108
Assignment Description:
Read chapter in assigned text.
Do corresponding homework problems and programming assignments.

METHODS OF EVALUATION:
Writing assignments
Percent of total grade: 0.00 %
Writing assignments: 0% - 0% This is a degree-applicable course, but substantial writing assignments are NOT appropriate, because the course primarily: Involves skill demonstrations or problem solving
Problem-solving assignments
Percent of total grade: 30.00 %
Problem-solving demonstrations: 30% - 60% Homework problems Quizzes Exams
Skill demonstrations
Percent of total grade: 30.00 %
Skill demonstrations: 30% - 60% Class performance Performance exams
Objective examinations
Percent of total grade: 20.00 %
Objective examinations: 20% - 40% Multiple choice True/false Matching items Completion
Other methods of evaluation
Percent of total grade: 0.00 %
Other methods of evaluation: 0% - 0%

REPRESENTATIVE TEXTBOOKS:
Required Representative Textbooks

Reading Level of Text, Grade: Reading level of text, Grade: 12+ Verified by: Verified by:ev
Recommended Representative Textbooks
Reading Level of Text, Grade: 12+ Verified by: ev using MSWord
ARTICULATION and CERTIFICATE INFORMATION

Associate Degree:
CSU GE:
IGETC:
CSU TRANSFER:
  Transferable CSU, effective 200770
UC TRANSFER:
  Transferable UC, effective 200770

SUPPLEMENTAL DATA:
Basic Skills: N
Classification: Y
Noncredit Category: Y
Cooperative Education:
Program Status: 1 Program Applicable
Special Class Status: N
CAN:
CAN Sequence:
CSU Crosswalk Course Department: CSIS
CSU Crosswalk Course Number: 43
Prior to College Level: Y
Non Credit Enhanced Funding: N
Funding Agency Code: Y
In-Service: N
Occupational Course: C
Maximum Hours:
Minimum Hours:
Course Control Number: CCC000435832
Sports/Physical Education Course: N
Taxonomy of Program: 070100