Lassen Community College Course Outline

CIS 81 Introduction to Python Programming 3.0 Units

I. Catalog Description

This course teaches Python programming for non-programmers, working professionals, and the curious. No previous programming experience is required. The course covers the fundamental concepts and models of application development including the program design, data types, programming, problem-solving, logic, and design techniques for event-driven programs. Provides hands-on experience with a modern development environment that is perfect for beginners. This course aligns with the PCEP and PCAP exams.

Prerequisites: None

Transfer Status: Non Transferrable
51 hours lecture, 102 hours: 153 total learning hours
Scheduled: Spring

II. Coding Information

Repeatability: Not Repeatable, Take 1 Time
Grading Option: Graded
Credit Type: Credit - Degree Applicable
TOP Code: 0707.1

III. Course Objectives

A. Course Student Learning Outcomes

Upon completion of this course the student will be able to:

1. Develop a working knowledge of how computers operate and how computer programs are executed.
2. Translate real-world issues into computer-solvable problems
3. Develop small programs that implement basic algorithmic designs

B. Course Objectives

Upon completion of this course the student will be able to:

1. Use primitive data types and data structures offered by the development environment.
2. Choose an appropriate data structure for modeling a simple problem.
3. Identify basic programming concepts.
4. Write simple applications that relate to a specific domain.
5. Design, implement, test, and debug a program that uses each of the following fundamental programming constructs: basic computation, simple I/O, standard conditional and iterative structures, and the definition of functions.
6. Test applications with sample data.
7. Apply core program control structure

IV. Course Content

A. Outline of Topics

1. Program design
2. Program development lifecycle
3. Requirements determinants and analysis
4. Modular design
5. Techniques for modeling program structures
6. Programming concepts
7. Variables
8. Literals
9. Types
10. Expressions
11. Procedures
12. Functions
13. Parameters
14. Operators and operations
15. Decision logic
16. Looping
17. Sub-procedures
18. Passing parameters
19. Coding
20. Unit testing
21. Control structures

V. Assignments

A. Appropriate Readings

Tech journals, blogs and online articles.

B. Writing Assignments

1. Write out, in detail, an algorithm for searching for a specific value in an array of integers, and implement the algorithm in a short program.
2. Using the sample program provided by the instructor, correct the formatting to follow the coding convention required in the class, and write documentation for all of the functions in the program.

C. Expected Outside Assignments

Students will be required to complete two hours of outside-of-class homework for each hour of lecture.

D. Specific Assignments that Demonstrate Critical Thinking

1. Design and fully document a set of test data that would be appropriate for the sample program provided by the instructor.
2. Design and implement a short program that opens a text file and searches for and counts the number of occurrences of a given string

VI. Methods of Evaluation

Exams
Quizzes
Projects

Lab Activities

VII. Methods of Delivery

Check those delivery methods for which, this course has been separately approved by the Curriculum/Academic Standards Committee.

☒**X** Traditional Classroom Delivery ☐ Web-enhance course

☐ Correspondence Delivery ☐ Hybrid Delivery X☒ Online Delivery

VIII. Representative Texts and Supplies

Cisco Network Academy Netacad learning management system. (www.netacad.com)

IX. Discipline/s Assignment

Computer Information Systems

X. Course Status

Current Status: Active
Original Approval Date: 10/18/2022
Course Originator:
Board Approval Date:11/8/2022
Chancellor's Office Approval Date:
Revised By:
Curriculum/Academic Standards Committee Revision Date: