

CS 111: INTRODUCTION TO PROGRAMMING CONCEPTS AND DESIGN

Citrus College Course Outline of Record

Heading	Value
Effective Term:	Fall 2023
Credits:	4
Total Contact Hours:	72
Lecture Hours :	72
Lab Hours:	0
Hours Arranged:	0
Outside of Class Hours:	144
Total Student Learning Hours:	216
Strongly Recommended:	ENGL 101.
Transferable to CSU:	Yes
Transferable to UC:	Yes - Approved
Grading Method:	Standard Letter, Pass/No Pass

Catalog Course Description

An introduction to the principles of computer programming and software development. Topics covered include the program development cycle, developing algorithms, data and control structures, structured programming, and object-oriented programming. Data types, expressions, control structures, functions, file and stream I/O, and structured and abstract data types are introduced in this course. Microsoft's Visual Studio's integrated development environment is used to help illustrate programming concepts common to modern high-level programming languages. Students must wait two years before retaking this course. 72 lecture hours.

Course Objectives

- explain the basic principles of good program design by summarizing the main steps involved in the program development cycle to understand how computer programs develop from conception to completion
- create programs that include repetition structures and selection structures as demonstrated through written exercises to design versatile programs
- create programs that include arrays, both one-dimensional and two-dimensional as demonstrated through written assignments to design versatile programs
- read independently to acquire information for discussion on the critical steps needed to be a software program regardless of the programming language used
- read independently to acquire information for discussion on the common features found in all high-level programming languages
- practice integrity by displaying ethical and honest behaviors in completing assignments while working as a member of a learning pair/group
- develop effective strategies to collaboratively work with others by participating in small groups to complete designated assignments in a timely, efficient manner

- display time management and student habits that reveal ability to complete assignments independently by meeting deadlines for submitting well-prepared work according to designated guidelines
- evaluate important ethical questions in the use of information technology (IT) and demonstrate effective decision-making strategies regarding IT ethics and issues by participating in directed discussions
- utilize the tools found in the interface to create, code, document, and test programs as demonstrated through written assignments to develop software applications
- utilize the tools found in the programming interface to create, code, document, and test programs as demonstrated through written assignments to develop software applications

Major Course Content

1. Software life-cycle including design, development, styles, documentation, testing and maintenance
2. Procedural versus objected oriented programming
 - a. Survey of current languages
3. Program design tools and programming environments
4. Documentation
5. Coding conventions
6. Data types, variables, expressions, sequential processing
7. Arrays
8. Declaring and allocating arrays
9. Multiple-subscripted arrays
10. Control structure
11. Selective structures: if and switch
12. Repetitive structures: loops
13. Algorithms including simple sorting and searching
14. File I/O
15. Files and streams
16. Sequential access files
17. Error handling
18. Passing parameters by value and by reference
19. Principles of testing and designing test data

Suggested Reading Other Than Required Textbook

www.htdp.org (i.e., How to design programs) www.code.org

Examples of Required Writing Assignments

Create a written program that a school teacher can use to keep track of his/her students' scores.

Examples of Outside Assignments

Create a GUI (i.e., graphical user interface) that allows the user to enter two numbers and then determines the minimum between the two simultaneously.

Instruction Type(s)

Lecture, Online Education Lecture