

Crafton Hills College Course Outline

1. **Discipline:** Computer Information Systems
2. **Department:** Business and Information Technology
3. **Course Title:** Introduction to C#.NET Programming
4. **Course I.D:** CIS 125
5. **Prerequisite(s):** None
Corequisite(s): None
Departmental Recommendation(s): None
6. **Semester Units:** 3
7. **Minimum Semester Hours:**
Lecture: 48 **Lab:** 0 **Clinic:** 0 **Field:** 0

8. **Need for the Course:**

The ability to design, code, and debug applications using an object-oriented language such as C# within the Microsoft .NET platform is in demand in the public and private sector. CIS 125 will provide students with the skills necessary to meet these demands. This course applies to the Associate in Science degree as well as to the certificate requirements for Computer Information Systems.

9. **Goals for the Course:**

CIS 125 enhances the offerings in Computer Information Systems, giving students the opportunity to learn an object-oriented by presenting an object oriented programming language. The addition of this course allows us to bring one of the most highly demanded courses to our program. This course is appropriate to the college's mission in that it is part of a comprehensive vocational education program leading to employment and preparation for transfer to a higher level of educational institution.

10. **Catalog Description:**

Introduction to object oriented programming using the C# programming language and the .NET platform environment. Includes program development, debugging, and implementation of software components including Windows desktop, console-based and web-based applications. Topics include the C# programming environment, program control structures, functions, classes, stream input/output, array processing, and event-driven, and web-based applications.

11. **Schedule Description:**

Introduction to object oriented programming using the C# programming language and the .NET platform environment. Includes program development, debugging, and implementation of software components including Windows desktop, console-based and web-based applications.

CIS 125

Page 1 of 4

Last Updated: 01/31/2005

Board Approved: 04/14/05

Semester Effective: Fall 2005

12. Entrance Skills:

- A. Requisite Skills:** None
- B. Recommended Skills:** None

13. Course Objectives:

Upon satisfactory completion of the course, students will be able to:

- A.** Interpret and analyze a problem and develop a program solution in C#.
- B.** Design, code, compile, debug, and run C# programs.
- C.** Design and develop a program that used standard input/output C# streams.
- D.** Explain and demonstrate proper structured modular program design.
- E.** Identify and define the standard C# and user defined data types.
- F.** Identify and define arrays.
- G.** Design, code, compile, debug and run an event based C# program in the Windows environment.

14. Representative Texts and Instructional Materials:

Dietel, H., Dietel, P (2004). *How to Program C#*. Boston, MA. Prentice Hall

Lippman, B. (2004). *C# Primer*. San Jose, CA. Wiley and Sons.

Doyle, B. (2004). *Visual C#.NET Programming: From Problem Analysis to Program Design*. Boston, MA. Thomson Press.

15. Course Content:

- A.** An Overview of C#
 - 1. C# programming environment
 - 2. Program design and preparation
 - 3. Input/Output streams
 - 4. Variables, expressions, and assignment
 - 5. Use of #define and #include
 - 6. Flow of control
 - 7. Functions
 - 8. Arrays and Strings
- B.** Lexical Elements, Operators and the C# System
 - 1. Preprocessor functions
 - 2. Working with a compiled language
 - 3. Libraries
 - 4. C# components
 - 5. C# syntax
 - 6. Keywords
 - 7. Identifiers
 - 8. Constants and variables
 - 9. Arithmetic, relational, and logical operators
- C.** Fundamental Data Types
 - 1. Standard C# data types
 - 2. User defined structures and classes
 - 3. Defining variables and constants of a particular data type
 - 4. Assigning values to variables of each data type

- D. Flow of Program Control
 1. Expression statements
 2. Compound statements
 3. If-Else statements
 4. While statement
 5. DO statement
 6. Break, continue, and exit statements
 7. Switch and case statements
 8. Conditional statements
- E. Functions
 1. Definitions
 2. Prototypes
 3. Parameters and arguments
 4. Scope rules
 5. Storage classes
- F. Arrays and Strings
 1. Declaring one-dimensional arrays
 2. Subscripting
 3. Using arrays for strings
 4. Working with multi-dimensional arrays
- G. Introduction to Classes
 1. Building classes
 2. Declaring objects of Classes
 3. Constructors
 4. Access
 5. Vectors
 6. Templates
- H. Windows and Event Based Programming
 1. Defining, creating and using GUIs (Graphical User Interface)
 2. Developing Windows forms and controls
 3. Defining, creating and using Delegates
 4. ListBox, ComboBox, Menu and CheckBox controls
- I. Errors and Handling Exceptions
 1. Defining
 2. Developing techniques
 3. Exception classes
 4. File streams, databases and ADO.NET
- J. Introduction to Web-Based Applications
 1. Defining
 2. Developing Web pages
 3. Developing HTML server, validation, custom and composite controls

16. Methods of Instruction:

The course will combine lecture, class discussion, computer-aided presentations, collaborative (group involvement) work and computer lab participation.

17. Assignments and Methods of Evaluation:

Students will complete a variety of exercises, programming projects, and examinations. Programming projects and/or exercises, and programming examinations will consist of modifying existing programs and/or creating new programs.

Programming projects and exercises: 30 - 50%

Programming examinations:	30 - 50%
In-class work:	0 - 20%
Collaborative group work:	0 -20%

18. Distributed Education Methods of Instruction: None