EL CAMINO COLLEGE
COURSE OUTLINE OF RECORD

I. COURSE DESCRIPTION

Course Title and Number: Computer Science 4

Descriptive Title: Computer Science Programming in C#

Discipline: Computer Science

Division: Mathematical Sciences

Course Length: ☒Full Term ☐Other (specify):

Hours Lecture: 3 Hours Laboratory: 3 Course Units: 4

Grading Method: ☒Letter ☐Credit/No Credit ☐Both ☐No Grade

Course Type: ☒Credit, Degree Applicable ☐Credit, Not Degree Applicable ☐Non-Credit

Transfer CSU: ☒Yes Effective Date _____ ☐No

Transfer UC: ☐Yes Approval Date _____ ☒Pending ☐No

Conditions of Enrollment:
Specify Prerequisite Corequisite, Recommended Preparation, Enrollment Limitation or None.

Prerequisite: Computer Science 1 with a minimum grade of C or equivalent

Catalog Description:

This course will introduce the student to the C# (C Sharp) computer programming language and the .NET framework. Students will write computer science programs using the C# language. Associated topics in the .NET framework will also be covered.

II. COURSE OBJECTIVES

List the major objectives of the course. These must be stated in behaviorally measurable terms.

1. Write C# programs that will use the inheritance and polymorphic capabilities of the C# language.
2. Write C# programs that will use C# exception handling capabilities.
3. Implement solutions to programming problems that require the C# features of indexers and properties.
4. Write C# programs using the Base Class Library's generic classes.
5. Write and implement programs using C# delegates and events.
6. Write and implement Graphical User Interface programs using the Base Class Library's Windows Forms Classes.
7. Write C# programs using both Base Class Library interfaces and programmer defined interfaces.
8. Implement solutions to programming problems that require file input and output involving both text files and object serialization.

III. OUTLINE OF SUBJECT MATTER

The topics should be detailed enough to enable an instructor to determine the major areas that should be covered and so that the course may have consistency from instructor to instructor and semester to semester.

Approximate Time in Hours	Major Topics
6	Introduction to the C# programming language • C# statements, looping and logic
24	C# classes • Fields and methods. • Public and private members. • Inheritance and polymorphism. • Overriding virtual functions
12	The Base Class Library • Generic Classes • Interfaces • Generic code
12	The Object class • Overriding the *ToString* and *Equals* methods. • Using the *GetType* method • Reflection
12	C# arrays, strings, properties and indexers
18	Delegates and events • Use in Windows Forms • Use in Asp.Net
12	File input and output • Text files • Object serialization
12	C# exception handling • Base Class Library exceptions • Custom exceptions

Total:108 hours

IV . METHODS OF EVALUATION

A. CREDIT, DEGREE APPLICABLE AND CREDIT, NOT DEGREE APPLICABLE COURSES

Check the PRIMARY method of evaluation for this course.

- ☐ Substantial writing assignments
- ☒ Problem solving demonstrations (computational or non-computational)
- ☐ Skills demonstrations

A minimum of one response in the categories 1, 2, or 3 below, as applicable, is required. However, you may check all that apply.

1. Indicate the types of writing assignments used as primary or secondary methods of evaluation for this course.
   - ☐ Essay exams
   - ☐ Written homework
   - ☐ Term or other papers
   - ☐ Reading reports
   - ☐ Laboratory reports
   - ☐ Other (specify)

2. Indicate the types of problem-solving demonstrations used as primary or secondary methods of evaluation for this course.
   - ☒ Exams
   - ☐ Laboratory reports
   - ☐ Quizzes
   - ☐ Homework problems
   - ☐ Fieldwork
   - ☒ Other (specify) Write computer programs

3. Indicate the types of skill demonstrations used as primary or secondary methods of evaluation for this course.
   - ☐ Class performance
   - ☐ Performance exams
   - ☐ Fieldwork
   - ☐ Other (specify)

4. If objective exams are also used, check all that apply.
   - ☐ Multiple choice
   - ☐ Completion
   - ☐ Matching items
   - ☐ True/false
   - ☐ Other (specify)

B. NON-CREDIT COURSE

Indicate the methods of evaluation that will be used to determine that stated objectives have been met.

V. COURSEWORK

A. TYPICAL ASSIGNMENT

Provide an example of a typical assignment. This assignment must correspond to the PRIMARY method of evaluation indicated in Section IV, Methods of Evaluation. That is, it must be a writing assignment or, if more appropriate, an assignment involving problem solving or skill demonstration.

Analyze and solve a programming problem that requires using a C# class. Write a C# class that models a rectangle. This class will represent the rectangle by storing two points – the top left point of the rectangle and the bottom right point of the rectangle. Write the following member functions:

1. Area – This function will calculate the area of the rectangle.
2. PointInRectangle – This function will determine if a given point is inside the rectangle.
3. Union – This function will calculate the smallest rectangle that contains two given rectangles.
4. Intersection – This function will calculate the largest rectangle that is contained in each of two given rectangles.

5. Draw – This function will draw the rectangle.

B. COLLEGE-LEVEL CRITICAL THINKING ASSIGNMENTS

Cite two specific assignments that demonstrate college-level critical thinking. (Required for degree applicable courses only.)

1. Analyze and solve a programming problem that requires using C# inheritance and file input and output.

Write a C# program that creates a class called Person with data fields name and age. Create a new class called Student that inherits from the class Person and contains the additional field gpa. Create an array that stores Person references. Add the ability to store data into this array, and the ability to read and write this array to a file.

2. Analyze and solve a programming problem that requires using C# exception handling.

Write a C# program that will test the capabilities of exception handling. Write an exception class called DivideByZero. Write a function called FindAverage which will take 2 arguments: an array of integers, and the number of values in the array. If the number of values in the array is zero, the FindAverage function will throw an exception of type DivideByZero. Test this in the main function by calling FindAverage several times with different values. Your program must use a C# try-catch block to *catch* the cases when division by zero occurs.

C. WORK OUTSIDE OF CLASS

Two hours work outside of class are required for each hour of lecture or equivalent. Each student in this course will be required to participate in the following work outside of class time. Check all that apply.

- ☒ Study
- ☒ Answer questions
- ☐ Skill practice
- ☐ Required reading
- ☒ Problem solving activity
- ☐ Written work (such as essay/composition/report/analysis/research)
- ☐ Journal (done on a continuing basis throughout the semester)
- ☐ Observation of or participation in an activity related to course content (such as theatre event, museum, concert, debate, meeting)
- ☐ Course is lab only - minimum required hours satisfied by scheduled lab time
- ☒ Other (specify) Write computer programs

VI. INSTRUCTIONAL METHODOLOGY

A. Check all planned instructional activities that apply:

☒ Lecture	☐ Group Activities
☒ Lab	☐ Role play/simulation
☐ Discussion	☐ Guest Speakers
☐ Multimedia presentations	☐ Field trips
☐ Demonstration	☒ Other (specify) Computer laboratory instruction

Note: In compliance with Board Policies 1600 and 3410, Title 5 California Code of Regulations, the Rehabilitation Act of 1973, and Sections 504 and 508 of the Americans with Disabilities Act, instructional delivery shall provide access, full inclusion, and effective communication for students with disabilities.

VII. TEXTS AND MATERIALS

If multiple selection is offered, only representative texts need be listed. An up-to-date list of required and recommended materials is maintained in the division office.

A. REQUIRED TEXTS (title, author, publisher, year)

Essential C# 2.0 , Mark Michaelis, Addison Wesley, 2006

B. REQUIRED SUPPLEMENTARY READINGS

C. OTHER REQUIRED MATERIALS

USB memory key.

VIII. CONDITIONS OF ENROLLMENT

If this course has a Prerequisite or Corequisite, complete section A. If this course has an Enrollment Limitation complete section B.

A. PREREQUISITE AND/OR COREQUISITE

1. Indicate if this course has a prerequisite or corequisite or both.
   ☒Prerequisite ☐Corequisite ☐Both

2. Indicate Type. Check all that apply.

   ☒Sequential ☐Computational/Communication Skills
   ☐Health and Safety ☐Non-Course
   ☐Standard (If this is a Standard prerequisite or corequisite, attach CCC Form D.)

3. Entrance Skills/Knowledge
   List the required skills and/or knowledge without which a student would be highly unlikely to receive a grade of A, B, C, or Credit (or for Health and Safety, would endanger self or others) in this course.

   1. Implement looping constructs in C++
   2. Create and use functions in C++
   3. Create and use simple C++ structures and classes

B. ENROLLMENT LIMITATION

1. Indicate the category which describes the Enrollment Limitation for this course.
   ☐Band/Orchestra
   ☐Theater
   ☐Speech
   ☐Chorus
   ☐Journalism
   ☐Dance
   ☐Intercollegiate Athletics
   ☐Honors Course
   ☐Blocks of Courses
   ☐Other (specify)
2. List Degree and/or Certificate requirements that are met by this course.
3. List all El Camino College courses that also satisfy the requirements listed above in

Originator: Joseph Hyman Submittal Date: September, 2001

BOARD APPROVAL DATE: _____

Reviewed and/or Revised by:

Joseph Hyman Date: September, 2007

_____ Date: _____

_____ Date: _____

REQUIRED SIGNATURES FOR NON-CREDIT COURSE

College Curriculum Committee Chair

_____

Vice-President - Academic Affairs

_____

CCC Form 1, 5/2006