C# Programming I

Course Information

Division: Business
Course Number: CMP 128
Title: C# Programming I
Credits: 3
Developed by: Lydia Mata
Lecture/Lab Ratio: 2 Lecture/2 Lab
Transfer Status:

<table>
<thead>
<tr>
<th></th>
<th>ASU</th>
<th>NAU</th>
<th>UA</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>CSE 182, CST 100, Computer/Stats (CS)</td>
<td>CIS 220</td>
<td>CSC Departmental Elective</td>
</tr>
</tbody>
</table>

Activity Course: No
CIP Code: 11.0100
Assessment Mode: Pre/Post Test (50 Questions/100 Points)
Semester Taught: Spring
GE Category: AAS degree only
Separate Lab: No
Awareness Course: No
Intensive Writing Course: No

Prerequisites
None

Educational Value
C# Programming I is intended for computer majors in both vocational and transfer degree programs. It also benefits community students who are interested in learning object-oriented programming, the Visual Studios development environment, and the C# programming language.

Description
C# Programming I is an introductory programming course for vocational computer majors and students transferring to universities in a business program. Course concentration includes the Visual Studios integrated development environment including user interfaces and controls, and object-oriented programming concepts including data types, classes, objects, methods, decision and repetition structures, and string and array manipulation. Computer literacy is assumed.

Supplies
USB/flash drive
Competencies and Performance Standards

1. Develop techniques for effective program design.
   Learning objectives
   What you will learn as you master the competency:
   a. Design the user interface.
   b. Plan the properties of each object in the interface design.
   c. Write pseudocode to represent C# programming language statements.
   Performance Standards
   You will demonstrate your competence:
   o By successful completion of multiple choice questions.
   o By successful completion of practical application projects.
   Your performance will be successful when:
   o Learner designs the user interface.
   o Learner plans the properties of each object in the interface design.
   o Learner writes pseudocode to represent C# programming language statements.

2. Build a user interface using the Visual Studio integrated development environment.
   Learning objectives
   What will learn as you master the competency:
   a. Use the various IDE windows.
   b. Choose correct controls for events.
   c. Determine the correct properties for each control.
   Performance Standards
   You will demonstrate your competence:
   o By successful completion of multiple choice questions.
   o By successful completion of practical application projects.
   Your performance will be successful when:
   o You use the various IDE windows.
   o Learner chooses correct controls for events.
   o Learner determines the correct properties for each control.

3. Determine appropriate use of data types including constants and variables.
   Learning objectives
   What you will learn as you master the competency:
   a. Categorize appropriate data types according to intended usage.
   b. Name data types according to good programming standards.
   c. Declare data types effectively.
   Performance Standards
   You will demonstrate your competence:
   o By successful completion of multiple choice questions.
   o By successful completion of practical application projects.
Your performance will be successful when:
- Learner categorizes appropriate data types according to intended usage.
- Learner declares data types effectively.
- Learner names data types according to good programming standards.

4. Design conditions, loops, lists, arrays, and structures.

   Learning objectives
   What you will learn as you master the competency:
   a. Build condition statements including various If and Case structures.
   b. Build and fill lists.
   c. Build various repetition structures.
   d. Store and access data using arrays.
   e. Build and use structures.

   Performance Standards
   You will demonstrate your competence:
   - By successful completion of multiple choice questions.
   - By successful completion of practical application projects.

   Your performance will be successful when:
   - Learner builds condition statements including various If and Case structures.
   - Learner builds and uses structures.
   - Learner stores and accesses data using arrays.
   - Learner builds various repetition structures.
   - Learner builds and fills lists.

5. Use the Menu Editor to add components to a component tray.

   Learning objectives
   What you will learn as you master the competency:
   a. Create menu items.
   b. Code menu items.
   c. Create a context menu.

   Performance Standards
   You will demonstrate your competence:
   - By the successful completion of multiple choice questions.
   - By the successful completion of practical application projects.

   Your performance will be successful when:
   - Learner creates menu items.
   - Learner creates context menus.
   - Learner codes menu items.
6. **Use C# programming language tools to access data in a database.**

   **Learning objectives**
   What you will learn as you master the competency:
   a. Create a connection object.
   b. Set up a data adapter.
   c. Define a dataset.

   **Performance Standards**
   You will demonstrate your competence:
   o In successful completion of multiple choice questions.
   o In successful completion of practical application projects.

   Your performance will be successful when:
   o Learner creates a connection object.
   o Learner defines a dataset.
   o Learner sets up a data adapter.

7. **Explain concepts of object-oriented computer programming.**

   **Learning objectives**
   What you will learn as you master the competency:
   a. Instantiate objects.
   b. Design classes.
   c. Create properties in a class.
   d. Apply the concepts of inheritance.
   e. Build constructors and destructors.

   **Performance Standards**
   You will demonstrate your competence:
   o By the successful completion of multiple choice questions.
   o By appropriate essay and short answer assessments.
   o By practical application projects.

   Your performance will be successful when:
   o Learner instantiates objects.
   o Learner builds constructors and destructors.
   o Learner applies the concepts of inheritance.
   o Learner creates properties in a class.
   o Learner designs classes.

8. **Develop data streams for data file handling.**

   **Learning objectives**
   What you will learn as you master the competency:
   a. Utilize data file terminology.
   b. Handle files using streams.
   c. Read and write data to files.
Performance Standards
You will demonstrate your competence:
  o By successful completion of multiple choice questions.
  o By successful completion of practical application assignments.
Your performance will be successful when:
  o Learner utilizes data file terminology.
  o Learner handles files using streams.
  o Learner reads and writes data to files.

9. Create graphics and animations using C# programming language.

Learning objectives
What you will learn as you master the competency:
  a. Draw objects using the C# graphics environment.
  b. Utilize the random number tool.
  c. Create simple animations.

Performance Standards
You will demonstrate your competence:
  o By multiple choice quizzes and practical application projects.
Your performance will be successful when:
  o Learner draws objects using the C# graphics environment.
  o Learner utilizes the random number tool.
  o Learner creates simple animations.

Types of Instruction
Classroom Presentation
On-Campus Lab

Grading Information

Grading Rationale
The post-test will count as the final exam and be represented as 10% of the overall grade.

Grading Scale
A  90-100%
B  80 - 89%
C  70 - 79%
D  60 - 69%
F  Below 60%