NVCC COLLEGE-WIDE COURSE CONTENT SUMMARY
ITP 136
C# Programming I (4 cr.)

Course Description
Presents instruction in fundamentals of object-oriented programming and design using C#.
Course content emphasizes program construction, algorithm development, coding, debugging,
and documentation of applications within the .NET Framework. Lecture 4 hours per week.

Recommended Co-requisites or Pre-requisites
ITP 100

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

- Acquire the fundamentals of object-oriented programming and design using C#
- Gain an understanding of concepts of object oriented language; encapsulation,
inheritance, and polymorphism
- Learn the C# programming within the .NET framework

Course Content
- Introduction to Computers and Programming Languages.
- C# language fundamentals.
- Classes and Objects
- Inheritance and polymorphism
- Interfaces
- Arrays and collections
- Strings and regular expressions
- Handling Exceptions

Student Learning Outcomes
Introduction to Computers and Programming Languages

- Learn the history of computers.
- Understand the difference between low level and high level programming languages.
- Understand difference between procedural languages and object-oriented language.
- Understand the basics of computer hardware.

C# Language Fundamentals

- Understand types, variables and constants
- Understand statements and operators
- Be able to work with namespaces

Classes and objects

- Understand how to define a class.
- Understand object-oriented concepts of classes
- Be able to create objects from the classes
- Understand method constructs.
- Understand how to pass parameters
- Understand the concept of data encapsulation

**Inheritance and Polymorphism**

- Understand object-oriented inheritance
- Be able to describe what polymorphism is and the advantages of it in a language
- Understand the concept of abstract classes
- Be able to describe boxing and unboxing

**Interfaces**

- Be able to describe what an interface is and why it is used
- Understand how to implement an interface

**Arrays and collections**

- Understand array basics
- Learn to declare and use one dimensional array
- Understand the effect of using array elements as parameter to methods
- Understand the Collection interfaces

**Strings and regular expressions**

- Understand C# strings
- Understand how to manipulate strings
- Be able to describe what regular expressions are and how to use them

**Handling exceptions**

- Understand how to throw and catch exceptions
- Be able to describe why you would want to use exceptions
- Learn how to re-throw exceptions