Course Description

Provides instruction in fundamentals of object-oriented programming and design using Visual C++ for GUI applications. Course content emphasizes software design and construction using the concepts of foundation classes. Lecture 4 hours per week.

General Course Purpose

Course Prerequisites/Corequisites

Prerequisites: 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 Visual C++ design for GUI applications
- Gain an understanding of software design and construction using the concepts of foundation classes
- Use Visual C++ to teach the C++ programming language and object-oriented concepts

Major Topics to be Included

- Programming Basics Using Visual C++
- Arithmetic Operators and Control Structures
- Arrays and Pointers
- Visual C++ Functions
- Classes
- Class Features and Design
- Overloading Operators
- Inheritance

Student Learning Outcomes

Programming Basics Using Visual C++

- Understand basic control structures
- Understand variables and named constants
- Understand declarations

Arithmetic Operators And Control Structures

- Understand Visual C++ binary arithmetic operators
- Use shortcut arithmetic operators
- Understand the logical OR, the while Loop and the for statement
- Use control structures with class object fields
Arrays And Pointers

- Use the address operator
- Understand arrays
- Store values in an array
- Access and use array values
- Create arrays of class objects

Visual C++ Functions

- Use functions and include files
- Use procedural abstraction
- Understand scope
- Construct function headers and prototypes
- Pass values and addresses to functions
- Pass arrays to functions

Classes

- Create classes with declaration and implementation sections
- Encapsulate class components
- Design classes
- Use private and public functions

Class Features And Design

- Understand member functions
- Understand constructors
- Using destructors

Overloading Operators

- Understand rules for overloading operators
- Overload math operators
- Overload output
- Overload input

Inheritance

- Understand inheritance
- Create derived classes
- Understand inheritance restrictions
- Override and overload parent class functions