Resource Person

Semester

Course Title Object Oriented Programming

Course Code

Course Type: Foundation

Pre-Requisite: Introduction to computer programming

Counseling Hours: Weekdays 4:00-6:00 pm

Program Head: Imran Saleem

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Name** | **Signature** | **Date** |
| **Prepared By**(Resource Person) |  |  |  |
| **Checked By**(Program Head) |  |  |  |
| **Approved By**(Director SPA) |  |  |  |

**Course Description**

This course introduces the object oriented programming concepts, principles, and techniques, including classes, objects, inheritance, and polymorphism. All these concepts are illustrated via a contemporary object oriented programming language (C#). Starting with the preliminary structured background, students will be gradually taught advance techniques used in enterprise software. That are strongly designed, well organized, easy to alter, difficult to crash and more secure from bugs and other mal functions.

**Course Instructional Objectives**

* Describe the principles of object oriented programming.
* Be able to create, debug and run simple C# programs.
* Understand fundamentals of Object Oriented Programming
* Apply standard coding practices
* Apply algorithmic thinking to solve programming problems.

**Course Student Objectives**

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

* Formulate problems as steps so as to be solved in object oriented paradigm
* Design, create, build, and debug object oriented applications
* Extract and analyze complex software requirements
* Learn the concepts of core object oriented paradigm
* Develop object oriented software with team-work in mind

**Course Contents**

Following is the session-wise breakup of the course:

**Session 1: Introduction to Object Oriented Programming**

* One-to-one introduction
* Course Introduction, Teaching & Assessment Methodology
* Distribution of Course Outlines
* Discussion on Course Outline
* Setting up of Norms
* A brief introduction to Object Oriented Programming (objects and classes)

**Learning Objectives**

The learning objective of this session will be to provide a basic understanding of Object Oriented Programming to the students.

**Session 2: Introduction to C# - I**

* Overview and environment
* Program structure
* Basic syntax

**Learning Objectives**

Students will learn about the basics of C# ranging from environment setup to basic syntax commonly used in programs. Having studied the pre-requisite course in C++, the students will also be taught the basic similarities and differences between the two languages.

**Activities**

* Lab Practice

**Session 3: Introduction to C# - II**

* Methods/functions
* Arrays
* Strings

**Learning Objectives**

Students will practice relatively advanced features of the language.

**Activities**

* Lab Practice
* Assignment 1

**Session 4: Classes and Objects**

* Identifying classes and objects
* Creating classes and class diagrams
* Attributes and methods
* Encapsulation and abstraction

**Learning Objectives**

Students will learn how to identify classes from real life objects and then implement it in a programming environment.

**Activities**

* Lab Practice
* Quiz 1

**Session 5: Constructors, Class Diagrams**

* Constructors
* Destructors/Finalizers
* Access Functions and “this” pointer
* “new” operator, Getter and Setter

**Learning Objectives**

In this session students will be given an overview of constructors and destructors (also known as finalizers), their types and different uses.

**Activities**

* Lab Practice
* Assignment 2

**Session 6: Inheritance**

* Inheritance
* Multiple Inheritance (C++)
* Problems related to multiple inheritance

**Learning Objectives**

In this session, students will be introduced to the concept of Inheritance and multiple inheritance. This will be mainly a practice oriented session. Towards the end of the session problems associated with multiple inheritance will be discussed (Dreaded Diamond Problem).

**Activities**

* Assignment 3
* Quiz 2

**Session 7: Polymorphism**

* Polymorphism
* Operator Overloading
* Overriding

**Learning Objectives**

This session will focus on polymorphism and different concepts related to it. The difference between overloading and overriding.

**Activities**

* Lab Practice
* Assignment 4

**Session 8: Association**

* Association
* Aggregation
* Composition
* Association Diagrams
* Friend Function

**Learning Objectives**

The students will be introduced to the concept of Class Association and the topics related to it. Also, they will learn to visually identify and construct class associations through Association Diagrams.

**Activities**

* Quiz 3

**Session 9 Mid Term**

**Session 10: Interfaces, Generics and Delegates**

* Abstract Classes
* Interfaces
* Generic Programming
* Generic Algorithms
* Delegates

**Learning Objectives**

The students will understand the concept of Abstract classes and interfaces. Along with a basic overview of generics and delegates (we will not go in the specifics as they are advanced topics)

Additionally, semester project ideas will be floated and the deadline for group formation and proposal submission will be announced.

**Activities**

* Lab Assignment

**Session 11: Creating GUIs with C# -I**

* Introduction to windows forms
* WPF (Windows Presentation Foundation)
* Basic elements of a windows forms and WPF applications

**Learning Objectives**

This session will focus on visual programming so students can create applications with basic GUI functionalities.

**Activities**

* Semester project proposal submission.
* Quiz 4
* Lab Exercise

**Session 12: Creating GUIs with C# -II**

* Event Handlers
* Exception Handling

**Learning Objectives**

The students will learn about Event handlers and exception handling while creating GUIs

**Activities**

* Assignment 5
* Quiz 5

**Session 13 Revision**

* Revision of whole course

**Learning Objectives**

This session will be mostly a discussion and revision of the whole course. Any queries from the students regarding the course material will be handled during this session.

**Session 14 LAB EXAM**

**Session 15 Project Presentations**

**Activities**

* Project presentations, demo and viva

**Recommended Book (s) & Text:**

The primary source of information in this course will be the MSDN website (referenced below), The following books are for reference only and their ebooks will be available on LMS.

* Beginning C# Object-Oriented Programming, 2nd Edition by *Dan Clark*
* Learning Object-Oriented Programming in C# 5.0 by *B.M.Harwani*

**Web Resources:**

* LMS course page
* [Tutorialspoint](https://www.tutorialspoint.com/csharp/)
* [MSDN](https://docs.microsoft.com/en-us/dotnet/csharp/programming-guide/concepts/object-oriented-programming)

**E-Resources:**

Lecture slides and relevant reading material along with eBooks for the course will be available on Moodle.

**ASSESSMENT METHODOLOGY**

|  |  |
| --- | --- |
| Assignments  | 15 |
| Class Participation | 10 |
| Quizzes | 10 |
| Semester Project | 10 |
| Mid Term | 15 |
| Lab Exam | 15 |
| Final Term Exam | 25 |
| Total | 100 |

* Assignments and quizzes will follow the best N-1 strategy
* Use of unfair means in any of the assessment activities is strictly forbidden. Anyone who violates this rule will be dealt with strictly according to the UMT policy.
* There will be no re-take of quizzes, semester project, midterm and lab exam.
* Class participation marks will be given based on healthy and constructive discussion done in class. This includes asking and answering questions logically keeping the course objective in mind.
* Any disruptive behavior which may result in distractions for the rest of the class will result in marks deductions from class participation.

**CALENDAR OF ACTIVITIES**

|  |  |  |  |
| --- | --- | --- | --- |
| **Session** | **Sub-Topic** | **Readings** | **Activities** |
| 1 | **Introduction to Object Oriented Programming** * One-to-one introduction
* Course Introduction, Teaching & Assessment Methodology
* Distribution of Course Outlines
* Discussion on Course Outline
* Setting up of Norms
* A brief introduction to Object Oriented Programming (objects and classes)
 | Selected topics from internet\* | Discussion  |
| 2 | **Introduction to C# - I*** Overview and environment
* Program structure
* Basic syntax
 | Selected topics from internet\* | Discussion / Code Practice |
| 3 | **Introduction to C# - II*** Methods/functions
* Arrays
* Strings
 | Selected topics from internet\* | Discussion / Code Practice**Assignment1** |
| 4 | **Classes and Objects*** Identifying classes and objects
* Creating classes and class diagrams
* Attributes and methods
* Encapsulation and abstraction
 | Selected topics from internet\* | Discussion / Code Practice**Quiz1** |
| 5 | **Constructors, Class Diagrams*** Constructors
* Destructors/Finalizers
* Access Functions and “this” pointer
* “new” operator, Getter and Setter
 | Selected topics from internet\* | Discussion / Code Practice**Assignment 2** |
| 6 | **Inheritance*** Inheritance
* Multiple Inheritance (C++)
* Problems related to multiple inheritance
 | Selected topics from internet\* | **Quiz 2****Assignment 3** |
| 7 | **Polymorphism** * Polymorphism
* Operator Overloading
* Overriding
 | Selected topics from internet\* | Discussion / Code Practice**Assignment 4** |
| 8 | **Association*** Association
* Aggregation
* Composition
* Association Diagrams
* Friend Function
 | Selected topics from internet\* | Discussion / Code Practice**Quiz 3** |
| 9 | **MIDTERM** |
| 10 | **Interfaces, Generics and Delegates*** Abstract Classes
* Interfaces
* Generic Programming
* Generic Algorithms
* Delegates
 | Selected topics from internet\* | Discussion **Project Announcement****Lab Assignment** |
| 11 | **Creating GUIs with C# -I*** Introduction to windows forms
* WPF (Windows Presentation Foundation)
* Basic elements of a windows forms and WPF applications
 | Selected topics from internet\* | Discussion / Code Practice**Quiz 4** |
| 12 | **Creating GUIs with C# -II** * Event Handlers
* Exception Handling
 | Selected topics from internet\* | Discussion / Code Practice**Quiz 5****Assignment 5** |
| 13 | **Revision*** Revision of whole course
 |  | Discussion  |
| 14 | **LAB EXAM** |
| 15 | **Project Presentations**  |  | **Project submission** |
| 16 | **FINALTERM** |

\* *The reading topics will be provided from the web resources mentioned above and the links will be available on LMS for each week’s discussion.*