

	HiLCoE School of Computer Science & Technology	Doc No. CD/HiL/001 Version: 002
Course Outline		Page 1 of 2

Program	B. Sc. In Computer Science										
Course Code / Title	CS223 / Windows Programming Using C#										
Year / Term	2019 / Winter										
Instructor	Andargachew A. (M. Sc.)										
Course Description	This course aims to providing the students with an understanding of visual programming concepts, principles and techniques involved in developing applications using C# as one of the windows programming tools. The students will become familiar with the C# design environment and the development of windows applications.										
Objectives	At the end of this course students are expected to: • Understand the .NET Framework and its components - Common Language Runtime (CLR), .NET Framework Class Library (FCL), .NET Common Language Specification (CLS), Assemblies, Assembly meta data and Global Assembly Cache (GAC) • Create, compile and run object oriented C# programs using Visual Studio • Write and understand C# language constructs, syntax and semantics • Produce programs for the desktop using solid multilayer architectures • Develop reusable .NET components via inheritance and interface realization • Leverage the major namespaces and classes of the .NET Framework										
Text Book	No text book										
Reference	• Fundamentals of Computer Programming with C#, Svetlin Nakov & Co., 2013 • The Complete Reference C# 4.0, the McGraw Hill, April 2010 • Professional C# 5.0 and .NET 4.5.1, Christian Nagel, Jay Glynn, Morgan Skinner, Wrox Programmer to Programmer, 2014 • A Programmer's Guide to C# 5.0, A. Eric Gunnerson and Nick Wienholt, 4th Ed.										
Assessment Method	<table border="0" style="width: 100%;"> <tr> <td>• Attendance & Class Appreciation</td><td style="text-align: right;">5%</td></tr> <tr> <td>• Assignments/Quiz:</td><td style="text-align: right;">10%</td></tr> <tr> <td>• Mid Exam:</td><td style="text-align: right;">20%</td></tr> <tr> <td>• Final Exam:</td><td style="text-align: right;">45%</td></tr> <tr> <td>• Lab/Project:</td><td style="text-align: right;">20%</td></tr> </table> NOTE: Missing four consecutive classes leads to RC grade Missing 25% of lecture classes leads to RC grade	• Attendance & Class Appreciation	5%	• Assignments/Quiz:	10%	• Mid Exam:	20%	• Final Exam:	45%	• Lab/Project:	20%
• Attendance & Class Appreciation	5%										
• Assignments/Quiz:	10%										
• Mid Exam:	20%										
• Final Exam:	45%										
• Lab/Project:	20%										

COURSE CONTENT

1. Introduction

- | | |
|--|---|
| <ul style="list-style-type: none"> • .NET Programming Framework & Languages • Common Language Runtime • What is C#? | <ul style="list-style-type: none"> • Some features of C#. • Visual design and event-driven programming. • Places to write codes in C#. |
|--|---|

	HiLCoE School of Computer Science & Technology	Doc No. CD/HiL/001 Version: 001
Course Outline		Page 2 of 2

2. Components of a .NET program
 - Statements, Number System
 - Naming Convention, Comments.
 - Variables, Constants.
 - Data Types, Operators.
3. Control Statements
 - Decision Statements.
 - Looping
 - Existing a control Structure.
4. OOP Concepts in C#
 - Classes & Objects
 - Constructors & Destructors
 - Functions / Methods in C#
 - Argument Passing Mechanisms
 - Properties & Indexers
 - Abstraction, Inheritance and Polymorphism
 - Interfaces
5. Error Handling Mechanisms / Exception Handling.
 - Why Error Handling Mechanisms.
 - Types of Errors, Error Handling
 - User-Defined Exception Classes
6. Windows Programming in C#
 - Windows Forms
 - Controls and Components
 - Windows applications
7. Collections
 - Collection Interfaces and Types
 - Lists, Queues, Stacks
 - Dictionaries
8. Introduction to Delegates, Events, Lambdas and Generics
 - Referencing Methods
 - Delegates, Events
 - Lambda expressions
9. Regular Expressions
 - Examining System.String
 - Regular Expressions
10. Files in C#
 - System.IO Namespace
 - Working with Directory and File Classes.
 - Working with Text Files. (sequential access file)
 - Working with Binary Files. (random access file)
11. Database Programming in C#
 - Data Access in C#.
 - The ADO.NET Data Architecture
 - Component Classes that makeup the Data Provider.
 - Working with Databases.
 - Working with Reports.
12. Introduction to Asynchronous Programming
 - Why Asynchronous Programming
 - Asynchronous patterns
 - Foundation of asynchronous programming
13. Introduction to Reflection
 - Manipulating and Inspecting Code at Runtime
 - Using Reflection
14. Introduction to Memory Management and Pointers
 - Memory Management
 - Unsafe Code