PROGRAMMING 2B
PROG6212
MODULE OUTLINE 2022
(First Edition: 2018)

The Independent Institute of Education (Pty) Ltd is registered with the Department of Higher Education and Training as a private higher education institution under the Higher Education Act, 1997 (reg. no. 2007/HE07/002).
Company registration number: 1987/004754/07.

Table of Contents

Introduction

The skills and knowledge gained in Programming 2A will be developed further through the addition of advanced OOP, GUI and database concepts. These skills aim to provide you with the ability to develop complete software solutions for given business requirements.

- The key purpose of the module is to provide advanced C# programming language.
- The module structure follows the layout of the textbook to ensure that you absorb all the advanced topics of the language.
- The module should be approached by re-enforcing the basics you learnt in Programming 2A. You are now familiar with Visual Studio IDE as a development environment, so it is important that you practice and become more familiar with the advanced topics of C#. For every new topic covered, it is recommended that you should spend at least an additional five hours (in class and in your own time) doing examples and exercises.
- Success can be ensured by having many and varied exercises to practice the new knowledge.

Using this Module Outline

This module outline has been developed to **support your learning**. Please note that the content of this module is on Learn as well as in the prescribed material. You will not succeed in this module if you focus on this document alone.

- This document does not reflect all the content on Learn, the links to different resources, nor the specific instructions for the group and individual activities.
- Your lecturer will decide when activities are available/open for submission and when these submissions or contributions are due. Ensure that you take note of announcements made during lectures and/or posted within Learn in this regard.

This Module on Learn

Learn is an online space, designed to support and maximise your learning in an active manner. Its main purpose is to guide and pace you through the module. In addition to the information provided in this document, you will find the following when you access Learn:

- A Module Pacer;
- An assessment brief;
- A list of prescribed material;
- A variety of additional online resources (articles, videos, audio, interactive graphics, etc.) in each learning unit that will further help to explain theoretical concepts;
- Critical questions to guide you through the module's objectives;
- Collaborative and individual activities (all of which are gradable) with time-on-task estimates to assist you in managing your time around these;
- Revision questions, or references to revision questions, after each learning unit.

Kindly note:

- Unless you are completing this as a distance module, Learn does not replace your contact time with your lecturers and/or tutors.
- PROG6212 is a Learn module, and as such, you are required to engage extensively with the content on the Learn platform. Effective use of this tool will provide you with opportunities to discuss, debate, and consolidate your understanding of the content presented in this module.
- You are expected to work through the learning units on Learn in your own time – especially before class. Any contact sessions will therefore be used to raise and address any questions or interesting points with your lecturer, and not to cover every aspect of this module.
- Your lecturer will communicate submission dates for specific activities in class and/or on Learn.

Icons Used in this Document and on Learn

The following icons are used in all your modules on Learn:

Icon	Description
	A list of what you should be able to do after working through the learning unit.
	Specific references to sections in the prescribed work.
	Questions to help you recognise or think about theoretical concepts to be covered.
	Sections where you get to grapple with the content/theory. This is mainly presented in the form of questions which focus your attention and are aimed at helping you to understand the content better. You will be presented with online resources to work through (in addition to the textbook or manual references) and find some of the answers to the questions posed.
	Opportunities to make connections between different chunks of theory in the module or to real life.
	Real life or world of work information or examples of application of theory, using online resources for self-exploration.

REMEMBER:

You need to log onto Learn to:

- Access online resources such as articles, interactive graphics, explanations, video clips, etc. which will assist you in mastering the content; and
- View instructions and submit or post your contributions to individual or group activities which are managed and tracked on Learn.

Module Resources	
Prescribed Material (PM) for this Module	Troelsen, A. and Japikse, P. 2021. *Pro C# 9 with .NET 5: Foundational Principles and Practices in Programming*. 10th ed. Apress. ISBN: 978-1-4842-6938-1(Paperback) or ISBN: 978-1-4842-6939-8(eBook) Available on Ebscohost at: https://ezproxy.iielearn.ac.za/login?url=https://search.ebscohost.com/login.aspx?direct=true&db=nlebk&AN=2917701&site=ehost-live&scope=site [Accessed 13 July 2022].
Recommended Readings, Digital and Web Resources	Please note that a number of additional resources and links to resources are provided throughout this module on the Learn platform. You are encouraged to engage with these as they will assist you in mastering the various objectives of this module. They may also be useful resources for completing any assignments. You will not, however, be assessed under examination conditions on any additional or recommended reading material. The following titles include information related to this module and may be consulted as additional resources. Please note, however, that you will not be tested on any content from these titles. Troelsen, A and Japikse, P. 2017. *Pro C# 7: With .NET and .NET Core*. 8th ed. Apress. ISBN: 978-1-4842-3017-6 (Paperback) or ISBN: 978-1-4842-3018-3 (eBook) Farrell, J. 2017. *Visual C# 2017: An Introduction to Object-Oriented Programming*. 7th ed. Cengage Publishing. Whitaker, R.B. 2016. *The C# Player's Guide*. The Internet is a valuable resource for all programming students as it provides up-to-date developments on the language and tools as the language and Microsoft.NET IDE get updated. Bookmark the Microsoft website to keep abreast of new developments in the language and the other websites for comprehensive tutorials and exercises which can provide additional explanations.

	• Microsoft, 2020. C# programming guide. [Online] Available at: https://docs.microsoft.com/en-us/dotnet/csharp/programming-guide/ [Accessed 13 July 2022].
Software required	Microsoft Visual Studio 2019
Software Licence requirements	
System Requirements	7 GB – .iso is provided for the students on the FTP Server for downloading Run on Host Computer - Standalone Machine
Module Overview	You will find an overview of this module on Learn under the *Module Information* link in the Course Menu.
Assessments	Find more information on this module's assessments in this document and on the Student Portal.

Module Purpose	
The purpose of this module is to build on the skills and knowledge gained in Programming 2A and further develop applications through the addition of advanced OOP, GUI and database concepts. These skills aim to provide you with the ability to develop complete software solutions for given business requirements.	
Module Outcomes	
MO1	Write advanced OOP software applications to meet given business requirements.
MO2	Create GUI systems in an OOP language to meet given business requirements.
MO3	Create GUI systems that integrate Relational Databases Management Systems, file handling, and OOP in one software application to meet given business requirements.

Assessments

Integrated Curriculum Engagement (ICE)	
Minimum number of ICE activities to complete	4
Weighting towards the final module mark	10%

Formative 1	Part 1
Weighting	25%
Duration	15 hours
Total marks	100
Open/Closed book	Open book
Resources required	• Prescribed textbook; • Microsoft Visual Studio (C#); • Access to the Internet.
Learning Units	Period 3
Learning Units covered	LU1 to 2

Formative 2	Part 2
Weighting	30%
Duration	15 hours
Total marks	100
Open/Closed book	Open book
Resources required	• Prescribed textbook; • Microsoft Visual Studio (C#); • Access to the Internet.
Learning Units	Period 6
Learning Units covered	LU 1 to 4 (Theme 1)

Summative	POE
Weighting	35%
Duration	15 hours
Total marks	100
Open/Closed book	Open book
Resources required	• Prescribed textbook; • Microsoft Visual Studio (C#); • Access to the Internet.
Learning Units covered	All The final completed POE must be submitted after LU5.

Assessment Preparation Guidelines	
Format of the Assessment	Preparation Hints
POE	
The POE will assess all learning units in this module and will be application-type questions. The POE is composed of Part 1 and Part 2. Part 1 (weight = 25%) assesses LU 1 to LU2, while Part 2 (weight = 30%) assesses LU 1 to LU4 (theme 1).	• By completing parts 1 and 2 in the POE you will develop some series of application in C# that will be applied to everyday real-world situation. • Ensure that you work through all the activities, exercises and revision questions on Learn and in your textbook. • Make sure that you are comfortable in responding to all the objectives for all learning units. • Brainstorm possible questions based on the learning outcomes and objectives provided. • Ensure that you have covered all items listed in the marking rubric of the POE.

Module Pacer			
Code	Programme	Contact Sessions	Credits
PROG6212	BCAD2	60 + 5 Learn Hours	15
	DISD2		
	BCA2		
	BCIS2		
	DIS2		
Learning Unit 1	Advanced C# Programming		

Overview:

In this this learning unit, you will be deepening your understanding of the C# language by going through a few more advanced concepts in C#. You will broaden your knowledge of the .NET type system through investigating indexers, operator overloading, extension methods and pointers. Also, you will take a first look at Language Integrated Query (LINQ), and learn various advanced features of C#. You will also learn about the life cycle of a C# object and multi-threaded programming.

If you are a contact student, you will likely spend 14 sessions on this learning unit.

Please work through Themes 1 to 5 on Learn, together with the relevant sections of your prescribed source/s. To ensure that you are working towards mastering the objectives for this learning unit, please also ensure that you complete the activities on Learn.

Learning Unit 1: Theme Breakdown		
Sessions: 1-14	Theme 1: Object Lifetime	Prescribed Material (PM)
Related Outcomes: MO001	LO1: Explain the garbage collection mechanisms in .NET; LO2: Develop C# programs using finalizable objects; LO3: Develop C# programs using disposable objects; LO4: Use lazy object initialisation to solve programming problems.	PM: Chapter 9 of the Prescribed Textbook.
	Theme 2: Advanced C# Features	PM: Chapter 11 of the Prescribed Textbook.
	LO5: Develop C# programs that make use of indexers; LO6: Develop C# programs that make use of operator overloading; LO7: Develop C# programs that make use of custom types; LO8: Develop C# programs that make use of extension methods; LO9: Develop C# programs that make use of anonymous types; LO10: Develop C# programs that make use of pointer types.	
	Theme 3: Language Integrated Queries	PM: Chapter 13 of the Prescribed Textbook.
	LO11: Use Language Integrated Queries to perform data operations in the C# environment; LO12: Use anonymous types with Language Integrated Queries in C# to represent the state and functionality of a given model item.	
	Theme 4: Processes, AppDomains and Object Contexts	PM: Chapter 14 of the prescribed textbook.
	LO13: Discuss the purpose of Windows Processes in C#; LO14: Use the C# AppDomains to monitor activities; LO15: Explain the use of the C# ObjectContext class.	

	Theme 5: Multithreaded-, Parallel- and Asynchronous Programming	PM: Chapter 15 of the prescribed textbook.
	LO16: Develop applications using Multithreading in C#; LO17: Execute parallel programming and Async Programming in C#; LO18: Differentiate between Multithreaded-, Parallel- and Asynchronous Programming.	

Learning Unit 2	Programming with the .NET Assemblies

Overview:

.NET Assemblies are the building blocks of the .NET Framework. They form the basic unit of deployment, reuse, version control, activation scoping, and security permissions. The .NET assemblies provide the information the common language runtime (CLR) needs to be cognisant of type implementations. It is also a collection of types and resources that are built to work together and form a logical unit of functionality. To the runtime, a type does not exist outside the context of an assembly.

If you are a contact student, you will likely spend 12 sessions on this learning unit.

Please work through Themes 1 to 4 on Learn, together with the relevant sections of your prescribed source/s. To ensure that you are working towards mastering the objectives for this learning unit, please also ensure that you complete the activities on Learn.

Learning Unit 2: Theme Breakdown		
Sessions: **15 - 26**	**Theme 1: Building and Configuring Class Libraries**	**Prescribed Material (PM)**
Related Outcomes: MO001	LO1: Use Class Libraries in C# to share code among applications; LO2: Explain under what circumstances Class Libraries can be used in C#; LO3: Discuss the benefits of using Class Libraries in C#.	PM: Chapter 16 of the Prescribed Textbook
	Theme 2: Type Reflection, Late Binding and Attribute-Based Programming	PM: Chapter 17 of the Prescribed Textbook.
	LO4: Use type reflection in C# to programmatically obtain metadata information; LO5: Develop C# programs to perform late binding using reflection.	
	Theme 3: Dynamic Types and the Dynamic Language Runtime	PM: Chapter 18 of the prescribed textbook.
	LO6: Use dynamic types in C# to simplify complex coding; LO7: Explain what Dynamic Language Runtime is and how to use it in C#.	
	Theme 4: The Role of Dynamic Assemblies	PM: Chapter 19 of the prescribed textbook.
	LO8: Discuss the role of Common Intermediate Language in C#; LO9: Implement dynamic assemblies in C#; LO10: Explain the role of Dynamic Assemblies in C# applications.	

Learning Unit 3	Files and Data

Overview:

By now, you should have a solid understanding of the C# language and the details of the .NET platform. In this learning unit, we will explore various types of commonly used services found within the base class libraries, including file I/O and database access using ADO.NET. This learning unit also covers data access using Entity Framework Core.

If you are a contact student, you will likely spend 13 sessions on this learning unit.

Please work through Themes 1 to 4 on Learn, together with the relevant sections of your prescribed source/s. To ensure that you are working towards mastering the objectives for this learning unit, please also ensure that you complete the activities on Learn.

Learning Unit 3: Theme Breakdown		
Sessions: 27 - 39	Theme 1: C# File I/O and Object Serialization	Prescribed Material (PM)
Related Outcomes: MO001 MO003	LO1: Create desktop applications to store data; LO2: Manipulate files and directories using core I/O types; LO3: Manipulate text data.	PM: Chapter 20 of the prescribed textbook
	Theme 2: Data Access with ADO.NET	PM: Chapter 21 of the prescribed textbook
	LO4: Develop data-centric applications using ADO.Net; LO5: Interact with relational databases using the core types and namespaces of ADO.net data providers.	
	Theme 3: Entity Framework Core	PM: Chapter 22 of the prescribed textbook.
	LO6: Explain the purpose of an object-relational mapper; LO7: Identify the building blocks of Entity Framework.	
	Theme 4: Data Access with Entity Framework Core	PM: Chapter 23 of the prescribed textbook.
	LO8: Model a database in memory; LO9: Develop data-centric applications using Entity Framework Core; LO10: Differentiate between code first and database first implementations; LO11: Create records using Entity Framework Core; LO12: Update records using Entity Framework Core; LO13: Delete records using Entity Framework Core.	

Learning Unit 4	Windows Presentation Foundation

Overview:

Windows Forms was the initial and earlier desktop GUI API that the .NET Platform supported. Although this API is still fully supported within the framework, an astounding API called Windows Presentation Foundation (WFP) has been introduced to programmers by the .NET 3.0, and this has quickly become the heir apparent to the Windows Forms desktop programming model. WPF allows you to build desktop applications that incorporate vector graphics, interactive animations, and data-binding operations using a declarative markup grammar called XAML. Additionally, the WPF control architecture provides an effortless way to restyle the look and feel of a typical control radically using little more than some well-formed XAML.

If you are a contact student, you will likely spend six sessions on this learning unit.

Please work through Themes 1 and 2 on Learn, together with the relevant sections of your prescribed source/s. To ensure that you are working towards mastering the objectives for this learning unit, please complete the activities on Learn.

<table>
<tr><th colspan="3">Learning Unit 4: Theme Breakdown</th></tr>
<tr><th>Sessions:
40 - 45</th><th>Theme 1: Model-View-ViewModel</th><th>Prescribed Material (PM)</th></tr>
<tr><td rowspan="3">Related Outcomes:
MO002</td><td>LO1: Use the Model View ViewModel pattern in the Windows Presentation Foundation;
LO2: Discuss the three important items used in implementing the Model View Model pattern in the Windows Presentation Foundation.</td><td rowspan="3">PM: Chapter 28 of the prescribed textbook.</td></tr>
<tr><td>Theme 2: Notifications, Validations and Commands</td></tr>
<tr><td>LO3: Use the binding notification system in the Windows Presentation Foundation;
LO4: Explain how to perform Windows Presentation Foundation Validation;
LO5: Create custom commands in a Windows Presentation Foundation app.</td></tr>
</table>

Learning Unit 5	ASP.NET Core Web Development

Overview:

A significant percentage of applications developed today are web-based. In this final Learning Unit, we will learn to construct Internet-based applications using ASP.NET Core. First, we will explore the model-view-controller pattern. Then, we will create RESTful webservices and web based user interfaces using ASP.NET Core.

If you are a contact student, you will likely spend 15 sessions on this learning unit.

Please work through Themes 1 to 3 on Learn, together with the relevant sections of your prescribed source/s. To ensure that you are working towards mastering the objectives for this learning unit, please complete the activities on Learn.

Learning Unit 5: Theme Breakdown		
Sessions: **46 - 60**	**Theme 1: Introducing ASP.NET Core**	**Prescribed Material (PM)**
Related Outcomes: MO002 MO003	LO1: Explain the Model View Controller (MVC) pattern. LO2: Identify the features of ASP.NET Core. LO3: Create an ASP.NET Core application.	PM: Chapter 29 of the prescribed textbook.
	Theme 2: RESTful Services with ASP.NET Core	PM: Chapter 30 of the prescribed textbook
	LO4: Develop RESTful web services using ASP.NET Core.	
	Theme 3: MVC Applications with ASP.NET Core	PM: Chapters 31 of the prescribed textbook.
	LO5: Develop a web application using ASP.NET Core.	