



Programming 1B
PROG6112
MODULE OUTLINE 2022
(First Edition: 2017)

This guide enjoys copyright under the Berne Convention. In terms of the Copyright Act, no 98 of 1978, no part of this manual may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording or by any other information storage and retrieval system without permission in writing from the proprietor.



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	3
Using this Module Outline.....	4
This Module on Learn.....	5
Icons Used in this Document and on Learn	6
Module Purpose.....	8
Module Outcomes.....	8
Module Pacer	13

Introduction

This module builds upon the introductory Programming 1A module and will extend the students' knowledge of Java as an Object-Oriented Programming (OOP) language. Students will create graphical user interfaces to given specifications. Students will also demonstrate the use of exceptions and apply file handling techniques and methods in programs.

At the end of each chapter in the textbook, there are several exercises that should be done to practise programming skills – students will benefit the most by doing all the exercises. In addition, students will be expected to convert pseudocode into programming language code, as this module is closely tied with the programming logic and design module.

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 difference 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 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.
- PROG6112 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
 Objectives	A list of what you should be able to do after working through the learning unit.
 Prescribed Work	Specific references to sections in the prescribed work.
 ThinkAbout	Questions to help you recognise or think about theoretical concepts to be covered.
 Active Learning	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.
 Connect the dots	Opportunities to make connections between different chunks of theory in the module or to real life.
 That is life!	Real life or world of work information or examples of application of theory, using online resources for self-exploration.
<p>REMEMBER:</p> <p>You need to log onto Learn to:</p> <ul style="list-style-type: none"> • 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	<p>Farrell, J. 2019. <i>Java Programming</i>. 9th edition. Cengage Learning.</p> <p>Free copy is available on EBSCOhost. eBook ISBN (Limited User Access: 1 copy available):</p> <p>https://ezproxy.iielearn.ac.za/login?url=http://search.ebscohost.com/login.aspx?direct=true&db=nlebk&AN=2281868&site=ehost-live</p>
Recommended Readings, Digital, and Web Resources	<p>Farrell, J. 2012. <i>Java Programming</i>. 8th edition. Cengage Learning.</p> <p>Burd, B. 2011. <i>Java For Dummies</i>. 5th edition. Place of publication: John Wiley & Sons, Sussex. ISBN: 978-0-470-37173-2</p> <p>Some useful web links:</p> <p>For students with no programming background:</p> <p>Home & Learn. 2013. <i>Java for Beginners</i> [Online]. Available at: http://www.homeandlearn.co.uk/java/java.html [Accessed 10 October 2013].</p> <p>Java Tutorials:</p> <p>Oracle. 2013. <i>The Java Tutorials</i>. [Online]. Available at: http://docs.oracle.com/javase/tutorial/ [Accessed 11 October 2013].</p> <p>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.</p>
Software required	Netbeans 8.1
Software Licence requirements	Freeware – Open Source
System Requirements	Lab spec sufficient
Lab minimum requirements	See IIE 2017 IT LAB – Software Activity Guide – Netbeans.docx
Module Overview	You will find an overview of this module on Learn under the <i>Module Information</i> 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 augment the skills acquired in Programming 1A by providing students with knowledge of applets, object inheritance and class manipulation and with the skills needed to apply this knowledge to finding programming solutions.

Module Outcomes

MO1	Demonstrate a comprehensive understanding of the OOP language in developing solutions.
MO2	Write programming solutions in an OOP language for a range of scenarios presented.
MO3	Apply basic and advanced OOP concepts and elements in programs.
MO4	Apply arrays and associated techniques in solution development.
MO5	Build graphical user interfaces to given specifications.
MO6	Demonstrate the use of exception handling in programs.
MO7	Apply file handling techniques and methods in programs.

Assessments

Integrated Curriculum Engagement (ICE)	
Minimum number of ICE activities to complete	4
Weighting towards the final module mark	10%
Complete the first ICE task by the end of:	LU1
Complete the first three ICE tasks by the end of:	LU3
Complete <u>all</u> ICE tasks by the end of:	LU4

Assignments/ Projects	Assignment
Weighting	25%
Duration	15 hours
Submit after	LU2
Learning Units covered	LU1–2
Period	2
Resources required	Textbook

Tests/ Examination	Test 1	Examination
Weighting	30%	35%
Duration	1.5 hours	3 hours
Total marks	60	120
Open/ closed book	Take-Home Test (Open book)	Take-home Exam (Open book)
Resources required	Textbook	Textbook
Learning Units covered	LU1–3	ALL
Period	4	N/A

Assessment Preparation Guidelines		
	Format of the Assessment (The Focus/ Approach/ Objectives)	Preparation Hints (How to Prepare, Resources to Use, etc.)
Assignment	<p>This assignment will challenge you to apply your programming knowledge obtained thus far using a set of questions/ specifications to demonstrate your understanding of concepts such as:</p> <ol style="list-style-type: none"> 1. Sorting Arrays; 2. Using two-dimensional arrays and other dimensional arrays; 3. ArrayList class; 4. Apply enum methods; 5. Passing an array to a method and using the length field; 6. Using nested loops; 7. Using the ArrayList Class; 8. Apply the concept of Inheritance; 9. Override superclass methods; 10. Call constructors during inheritance; 11. Apply information hiding. 	<p>You will need to study similar problems in your prescribed textbook and other sources to solve the assignment.</p>
Test	<p>The test will consist of an open book test on the concepts covered in programming. It will include practical questions to write applications from given problem scenarios. For open book assessments, the students may have open access to all resources, including notes, books (hardcopy and e-books), and the internet. These resources may be accessed as hard copies or as electronic files on electronic devices. All electronic device batteries must be fully</p>	<p>To prepare effectively for this test, you can include the following in your preparation:</p> <p>1. Advanced Array Concepts:</p> <p>You should be able to:</p> <ul style="list-style-type: none"> • use a two-dimensional array to produce a report that has totals for each item; • declare and populate a two-dimensional array; • print of the rows and columns of the two-dimensional array;

	<p>charged before the assessment as no charging of devices will be permitted during the sitting of the assessment. The IIE and associated brands accept no liability for the loss or damage incurred to electronic devices used during open-book assessments.</p>	<ul style="list-style-type: none"> • accumulate totals using the two-dimensional array; • print the totals of the two-dimensional array; • produce a report of a two-dimensional array. <p>2. Inheritance</p> <p>You should be able to:</p> <ul style="list-style-type: none"> • Create a class interface: <ul style="list-style-type: none"> o Constructor; o Variables and Methods; and o Input and Output. • Extend a class that contains: <ul style="list-style-type: none"> o Constructor; o Print Method; • Instantiate a class; • Produce a report/output. <p>Revise past question papers as well.</p>
Exam	<p>The paper will cover All LUs and consist of a take-home exam on all the programming concepts covered. In addition, it will include writing applications from given problem scenarios.</p>	<p>To prepare effectively for the exam, you can include the following in your preparation:</p> <p>1. Java GUI Application</p> <p>You should be able to:</p> <ul style="list-style-type: none"> • Allow a user to view the information by using GUI forms (e.g. combo boxes, radio buttons, buttons, etc.) • Load/populate GUI controls • Declare objects, class definition, method calls and use of good programming logic. • comment program • Produce a report / Output using an effective form layout

	<p>2. Java Application using Two-Dimensional Array and a Single Array</p> <p>You should be able to:</p> <ul style="list-style-type: none"> • <i>Use arrays (2D and single array) to store data, search and produce a report with totals for each item.</i> • Declaration and Population of Two-Dimensional Array. • Printing of the rows and columns. • Accumulating totals (calculating). • Printing the totals. • Report produced / Output <p>3. Java GUI Application</p> <p>You should be able to:</p> <ul style="list-style-type: none"> • Create a sequential file that contains field's data. • Load the data from the text file and populate a GUI control. • Display the information • Save to the text file • Produce a report / Output using an effective form layout • Check if you are confident that you could solve problems similar to the exercise questions at the end of the chapters by writing code. • Work through the exercises at the end of the chapters. • Revise past question papers.
--	--

Module Pacer			
Code	Programme	Contact Sessions	Credits
PROG6112	BIS2	60 + 5 Learn	15
PROG6112	BCA1; BCAD1; BCIS1; BCN1; DIS1; DISD1		
Learning Unit 1	Advanced Array Concepts		

Overview:

Real-world business processes are, at least from a programming perspective, usually quite complex. To write code for such complexities, software developers must have an excellent foundation and thorough understanding of the basic principles of coding.

This learning unit introduces you to advanced array concepts related to sorting, two and multi-dimensional arrays and the use of array classes as a powerful tool to manipulate arrays and programmer-defined data types (enumerations).

If you are a contact student, you will likely spend 10 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 following activities on Learn:

Activity		Estimated Time on Task
1.1.1	Write a program using a Bubble Sort	90 minutes
1.1.2	Bubble sort vs Insertion sort	60 minutes
1.2.1	Programming with Multi-Dimensional Arrays	120 minutes
1.2.2	Java Program with Nested Loops	60 minutes
1.3.1	Java program using the Arrays Class	90 minutes
1.3.2	Java program using the ArrayList Class	60 minutes
1.4.1	Java program using an enum	60 minutes
Estimated total time for this learning unit's activities		9 hours

Learning Unit 1		
Sessions: 1-10	Theme Advanced Arrays	Prescribed Material (PM)
Related Outcomes: MO3	Learning Objectives: <ul style="list-style-type: none"> • Sort an array using the bubble sort algorithm in ascending or descending order; • Describe how the bubble sort can be modified for improved efficiency; • Describe a more efficient method of sorting, namely the Insertion Sort; • Create two subscripts to access an element in a two-dimensional array; • View a two-dimensional array in memory; • Pass a two-dimensional array to a method; • Use the length field with a two-dimensional array; • Use the methods of the Array class; • Create containers that store lists of objects using ArrayList class; • Compare ArrayList Class to Arrays class; • Apply enum methods 	Chapter 9

Learning Unit 2		Introduction to Inheritance
<p>Overview:</p> <p>This learning unit addresses the concept of inheritance – an integral part of programming using object-orientation. We will investigate the concept of inheritance, learn how to extend classes, override and access superclass methods, call constructors during inheritance, and apply information hiding. We conclude with a look at the methods that you cannot override.</p> <p>If you are a contact student, you will likely spend 11 sessions on this learning unit.</p> <p>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 complete the following activities on Learn:</p>		
Activity		Estimated Time on Task
2.1.1	Inheritance in a Java Program	90 minutes
2.2.1	Overriding superclass methods	90 minutes
2.3.1	Calling Constructors and Using Superclasses	90 minutes
2.4.1	Information Hiding	90 minutes
Estimated total time for this learning unit's activities		6 hours

Learning Unit 2		
Sessions: 11-21	Theme Introduction to Inheritance	Prescribed Material (PM)
Related Outcomes: <ul style="list-style-type: none"> MO1 MO2 	Learning Objectives: <ul style="list-style-type: none"> Apply the concept of inheritance; Extend classes; Override superclass methods; Call constructors during inheritance; Access superclass methods; Apply information hiding; Explain which methods you cannot override. 	Chapter 10

Learning Unit 3		Advanced Inheritance Concepts and Exception Handling
<p>Overview:</p> <p>As suggested in the title, this learning unit introduces advanced inheritance concepts and exception handling. We will look at abstract classes, dynamic method binding, arrays of subclass objects, and the object class and its methods. We will also explore the use of inheritance to achieve good software design and conclude with an overview of interfaces and packages.</p> <p>If you are a contact student, you will likely spend 12 sessions on this learning unit.</p> <p>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 complete the following activities on Learn:</p>		
Activity		Estimated Time on Task
3.1.1	Learning to program using abstract classes	90 minutes
3.1.2	Creating subclasses and arrays of objects	180 minutes
3.3.1	Creating an interface	90 minutes
3.4.1	Creating Packages	90 minutes
Estimated total time for this learning unit's activities		7.5 hours

Learning Unit 3		
Sessions: 22-33	Theme : Advanced Inheritance Concepts and Exception Handling	Prescribed Material (PM)
Related Outcomes: <ul style="list-style-type: none"> • MO1 • MO2 • MO3 • MO4 • MO6 	Learning Content <ul style="list-style-type: none"> • Create and use abstract classes; • Use dynamic method binding; • Create arrays of subclass objects; • Use the Object class and its methods; • Use inheritance to achieve good software design; • Create and use interfaces; • Create and use packages; 	PM: Chapter 11

Learning Unit 4		Exception Handling
<p>Overview:</p> <p>This learning unit introduces exception handling. We will start with the coding of catch exceptions and finally blocks. After this, we will trace exceptions through the call stack and use assertions.</p> <p>If you are a contact student, you will likely spend 5 sessions on this learning unit.</p> <p>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 following activities on Learn:</p>		
Activity		Estimated Time on Task
4.1.1	Use exception handling instructions	150 minutes
4.2.1	Using try and catch blocks	150 minutes
Estimated total time for this learning unit's activities		5 hours

Learning Unit 4		
Sessions: 34-38	Theme: Exception Handling	Prescribed Material (PM)
Related Outcomes: <ul style="list-style-type: none"> • MO1 • MO2 • MO3 • MO4 • MO6 • MO7 	Learning Content: <ul style="list-style-type: none"> • Try code and catch exceptions; • Use the final block; • Specify the exceptions a method can throw; • Trace exceptions through the call stack; • Use an assertion. 	Chapter 12

Learning Unit 5		File Input and Output
<p>Overview:</p> <p>This learning unit explains the fundamentals pertaining to file handling. We will start with Java input and output classes to write to and read from files, then create and use sequential and random access files in programs. Finally, we conclude with programs that write and read records from random access data files.</p> <p>If you are a contact student, you will likely spend 10 sessions on this learning unit.</p> <p>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 following activities on Learn:</p>		
Activity		Estimated Time on Task
5.1.1	Use InputStream and OutputStream in programs	120 minutes
5.2.1	Creating and using sequential files	180 minutes
5.3.1	Program using random access files	180 minutes
Estimated total time for this learning unit's activities		8 hours

Learning Unit 5		
Sessions: 39-48	Theme: File Input and Output	Prescribed Material (PM)
Related Outcomes: <ul style="list-style-type: none"> • MO1 • MO2 • MO3 • MO4 • MO6 • MO7 	Learning Content: <ul style="list-style-type: none"> • Use Java's input and output (IO) classes to write to and read from a file; • Create and use sequential data files; • Create and use random access files; • Write records to a random access data file; • Read records from a random access data file. 	Chapter 13

Learning Unit 6		Introduction to Swing Components and Advanced Graphical User Interface Topics
<p>Overview:</p> <p>This learning unit will address graphical user interface components and associated fundamental concepts that allow computer programs to be more user-friendly and fun to use. We will start with an overview of event-driven programming and swing components. Next, you will learn more about using and extending the JFrame and adding JTextFields, JButtons and tool tips to a JFrame. This is followed by using the JLabel Class, layout managers, content panels, colour and JPanels and JScrollPane to increase layout options. The handling of mouse events and the use of menus will conclude the learning unit and module.</p> <p>If you are a contact student, you will likely spend 12 sessions on this learning unit.</p> <p>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 following activities on Learn:</p>		
Activity		Estimated Time on Task
6.1.1	Program using Swing components	120 minutes
6.2.1	Programming using advanced GUIs	180 minutes
Estimated total time for this learning unit's activities		5 hours

Learning Unit 6		
Sessions: 49-60	Theme: Introduction to Swing Components and Advanced Graphic User Interface Topics	Prescribed Material (PM)
Related Outcomes:	Learning Content:	Chapter 14 and 15
<ul style="list-style-type: none"> • MO1 • MO2 • MO3 • MO4 • MO6 • MO7 	<ul style="list-style-type: none"> • Discuss the concept of Even-Driven Programming; • Describe the swing components; • Use the JFrame; • Extend the JFrame; • Add JTextField, JButtons and tool tips to a JFrame; • Use the JLabel Class; • Use a layout manager; • Use the content panels; • Use colour and JPanels to increase layout options; • Create JscrollPanels; • Handle mouse events and using menus 	