Module: Programming 261

Module name:	Programming 261
Code:	PRG261
NQF level:	6
Type:	Fundamental – Diploma in Information Technology (Software Development stream)
Contact time:	48 hours
Structured time:	8 hours
Self-directed time:	64 hours
Notional hours:	120 hours
Credits:	12
Prerequisites:	Programming 161

Purpose

The aim of this course is to introduce the student to the diverse possibilities of professional User Interface Development for desktop applications. The course will focus on Windows Forms and students will learn how to create professional aesthetics and interactivity. Students will learn how to work with a large number of controls and containers, how to customize them and the powerful event handling exposed by Windows Forms. Students will have a sound understanding of programming paradigms with an emphasis on Object Oriented Programming. This course brings together all the concepts learnt in the various pre-requisite programming offerings and adds a few more advanced topics such as event driven programming.

Outcomes

Upon successful completion of this module, the student will be able to demonstrate:

- Detailed knowledge of Object Oriented Programming, concurrency and graphical user interfaces, including an understanding of and the ability to apply key concepts such as exception handling facilities in an Object Oriented Programming paradigm.

- The ability to evaluate, select and apply appropriate methods, procedures, or techniques relevant to Object Oriented Programming.

- The ability to identify, analyse and solve problems in real world contexts, gathering evidence and applying object orientated solutions based on evidence and procedures appropriate to object-orientated design.

- The ability to effectively apply concurrency techniques relevant to Object Oriented Programming.

Assessment

Assessment is performed using a variety of instruments:

- Continuous evaluation of theoretical work through a written assignment, a formative test, and a summative test.
- Continuous evaluation of project work.

- Final assessment through a written examination.
- The assignments or projects collectively will count 30% of your class mark.
- All tests will collectively account for 70% of your class mark.
- Your class mark contributes 30% towards your final mark for the subject, while the final assessment accounts for 70% of your final mark.

Teaching and Learning

Learning materials

Prescribed books (EBSCO)

- 📖 *Christy, V. (2015)* **Programming in C#**. *New Delhi, India: Laxmi Publications Pvt Ltd.*
- 📖 *Ringler, R. (2014)* **C# Multithreaded and Parallel Programming**. *Birmingham: Packt Publishing (Professional Expertise Distilled).*
- 📖 *Miller, R. (2012)* **C# Collections : A Detailed Presentation**. *Falls Church, Va: Pulp Free Press. Learning activities*

Learning activities

Learning will be facilitated by the lecturer with student centred activities that involve problem-based learning where pupils are presented with challenges that replicate the situation in the real-world environment. One mandatory assignment and one project must be completed during the course. This will be achieved through a combination between presentation of theoretical concepts, guided exercises, group work and discussions during the module.

Notional learning hours

Activity	Units	Contact Time	Structured Time	Self-Directed Time
Lecture		40.0		28.0
Formative feedback		5.0		
Project	1	3.0		9.0
Assignment	1			3.0
Test	3		6.0	11.0
Exam	1		2.0	13.0
		48.0	**8.0**	**64.0**

Syllabus

- Introduction to OOP
- Object-Oriented Programming
    - Classes and objects
    - Methods and messages
    - Classification, generalization and specialization
    - Inheritance
    - Interfaces and inner classes
    - Polymorphism
    - Abstraction

- Events and delegates
- Concurrency
  - o Threads
  - o Background worker
  - o Thread pool
- Architecture of Windows Forms
- Windows Forms basics
  - o Controls and containers
- Exception Handling