# Vocational Training Development Institute

**School of Applied Technology**

**Information and Communication Technology Department**

COURSE OUTLINE

**Introduction to Object Oriented Programming II CS262**

**75 hours, 3 credits**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Semester:** | 2 | **Academic Year:** | 2 | | **Programme:** | | ASc. ICT | |
| **Lecturer:** |  | | | | | | | |
| **Lecturer Contact Info.:** | *Email:* | | | *Tel:* | | | | |
|  | | |  | | | | |
| **Class Sessions:** | *Day(s)* | | *Time* | | | *Room* | | *Group (if applicable)* |
|  | |  | | |  | |  |
|  | |  | | |  | |  |
| **Modality** | Face to Face  Blended  Online | | | | | | | |
| **Pre-requisite** |  | | | | | | | |

**COURSE DESCRIPTION**

This course offers an introduction to object oriented programming concepts using a modern programming language that is popular in industry

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **At the end of this course, students will be able to:** | | | | |
| **COURSE OBJECTIVES** | * Set up classes in a modern Object Oriented language * Use and understand inheritance facilities in a modern Object Oriented language * Use and understand interfaces in a modern Object Oriented language * Demonstrate polymorphic behavior in a modern Object Oriented language * Apply generics and collections in a modern Object Oriented language * Use exception handling facilities in a modern Object Oriented language * Perform object serialization and handle random file in a modern Object Oriented language * Set up Graphical User Interfaces in a modern Object Oriented language | **ALIGNED PROGRAMME LEARNING OUTCOMES \*** | Apply the competencies required to function effectively as ICT professionals |  |
| Communicate clearly to the expert and the novice using a variety methods |  |
| Demonstrate the ability to adapt to new technologies and methods |  |
| Contribute to technological innovation and improvement |  |
| Exhibit high standards of professional and ethical responsibility |  |
| Demonstrate good analytical, design, and implementation skills required to formulate and solve computing problems |  |
| Display research and evaluation skills |  |
| Demonstrate the ability to successfully pursue graduate study in information systems or related disciplines |  |
| Demonstrate the main personal entrepreneurial characteristics (pecs) of successful entrepreneurs |  |
| \* Extract from programme’s *IRMA Curriculum Mapping Document* (**I** – Introduce; **R** – Reinforce; **M** – Mastery; **A** – Assessment) | | | | |

| **Aligned Graduate Profile** | |
| --- | --- |
| Competent  Self-directed  Solution Oriented  Effective Communicator | Ethical Professional  Responsible Citizen  Entrepreneurial  Technologically Fluent |

**EMPLOYABILITY SKILLS**

At the end of this course, students should be able to:-

* Work as software developers using any modern object oriented programming language

**EXPECTATIONS OF STUDENTS**

To aid their successful completion of this course, students are expected to:-

* Attend all classes
* Complete all practical assignments
* Prepare their personal machines with requisite software of use that provided by the institution
* Complete assigned readings

**COURSE SCHEDULE**

| **Week** | **Lessons** | **Competencies** | **Assignment Schedule** |
| --- | --- | --- | --- |
| 1 | **Module 1: Nature of Object Oriented programming language and specifics**  Lesson 1 – Setting up the Development Environment   * Installation, Requisite Software Tools, Supporting Documentation * Runtime Environment * Program Compilation * Project/Program Anatomy * Supported Data Types * Reserved Words * Supported Control Structures | * Understand the nature of the Object Oriented language * Understand the anatomy of a program and project in the Object Oriented language * Understand the runtime environment of the development platform * Understand the compilation process of the development platform * Know supported data types * Know reserved words * Know supported control structures * Understand how to model UML class diagrams | **PRACTICAL ASSIGNMENT** |
| 2 | Lesson 2 – Program Anatomy, I/O Operations,Mathematical Operations, Randon Numbers   * Main * Basic Input * Basic Output * Basic Mathematical Operations (Language API’s) * Generating Random Numbers * Terminal Operations | * Setup Main * Perform basic input * Make use of Mathematical API’s * Understand and demonstrate random number generation * Put to use terminal commands for development platform | **PRACTICAL ASSIGNMENT** |
| 3 | **Module 2: Libraries, Class definition**  Lesson 3 – Writing Classes   * Built in and Custom Libraries * Class Definition * Access Modifiers * Getters * Setters * Method Overloading * UML Modeling and Class Diagrams * Composition | * Make use of built-in-libraries * Assemble custom libraries * Setup a class * Put to use access modifiers * Setup getters * Setup setters * Setup other intrinsic class methods * Apply method overloading * Interpret UML class diagrams * Demonstrate composition | **PRACTICAL ASSIGNMENT** |
| 4 | **Module 3: inheritance, abstract classes and polymorphism**  Lesson 4 – Inheritance and Language Inheritance Hierarchy   * Method Over-riding * Abstract Classes * Abstract methods * Polymorphism with Super Classes | * Understand inheritance and language inheritance hierarchy * Make use of inheritance * Demonstrate inheritance * Apply method over-riding * Setup abstract classes and methods * Demonstrate polymorphism with super classes | **PRACTICAL ASSIGNMENT** |
| 5 | **Module 4: interfaces and polymorphism**  Lesson 5 – Interfaces and Polymorphism   * Interfaces * Enhancing class functionality with interfaces * Polymorphism wit Interfaces | * Understand interfaces in a modern Object Oriented Language * Apply interfaces to classes to enhance functionality * Demonstrate Polymorphism using interfaces | **PRACTICAL ASSIGNMENT** |
| 6 | Practical assessment  modules 2-4 | * PRACTICAL ASSESSMENT | **LAB TEST 1** |
| 7 | **Module 5: generics and collections**  Lesson 6 – Generics and Collections   * Arrays * Generics * Collections * Polymorphism with Arrays, Generics and Collections | * Make use of Arrays in modern Object Oriented language * Understand collections and collection history * Understand generics * Apply generics * Demonstrate polymorphism using generics | **PRACTICAL ASSIGNMENT** |
| 8 | **Module 6: exception handling**  Lesson 7 – Exception Handling   * Errors and Exception Hierarch * Exception Handling scopes and blocks * User Defined Exceptions * Declaring Exceptions | * Setup Custom Exceptions * Apply exception declaration | **PRACTICAL ASSIGNMENT** |
| 9 | **Module 7: Object Serialization and file management**  Lesson 8 – Object Serialization and Random File Management   * Object Serialization * Random File Handling and Management * Reading from files using Input streams * Writing to files using output streams | * Understand Object Serialization * Demonstrate Object Serialization * Perform file input using output streams * Perform file output using output streams | **PRACTICAL ASSIGNMENT** |
| 10 | practical assessment  modules 5-7 | PRACTICAL ASSESSMENT | **LAB TEST 2** |
| 11 | **Module 8: graphical user interfaces**  Lesson 9 – GUI   * Heavy Weight Components * Light Weight Components * Layout Managers * Windows, Frames and Containers * Labels * Text Box * Button | * Assemble GUI * Understand and make use of heavyweight components * Understand and make use of lightweight components * Understand and make use of layout managers * Understand and make use of Windows, Frames and Containers * Make use of labels, text boxes and buttons | **PRACTICAL ASSIGNMENT** |
| 12 | Lesson 10 –   * Event Programming * Message Dialogs * Text Area * Scroll Pane * Combo Box * Check Box * Radio Button * Button Groups | * Understand purpose of listeners * Use listeners * Perform Event Programming * Make use of text area, scroll pane, combo box, check box, radio button and button group | **PRACTICAL ASSIGNMENT** |
| 13 | practical assessment  module 8 | PRACTICAL ASSESSMENT | **LAB TEST 3** |
| 14 | **STUDY BREAK** | | |
| 15 | **FINAL ASSESSMENT PERIOD** | | |

**ASSESSMENT STRATEGY**

|  |  |  |
| --- | --- | --- |
| **Module** | **Assessment** | **Weighting** |
| Module 2-4`: | Practical Assessment 1 | 20% |
| Module 5-7: | Practical Assessment 2 | 20% |
| Module 8: | Practical Assessment 3 | 20% |
| All Modules | Final Exam | 40% |

Overall score will be obtained by finding the average of the scores obtained per module. **A maximum of one** coursework **and/or** **one** final exam is to be assigned per module.

**REFERENCE MATERIAL**