| Module Title: | Programming |
|---|---|

| Credits: | 10 |
|---|---|

| NFQ Level: | 6 |
|---|---|

| Module Delivered In | 1 programme(s) |
|---|---|

| Teaching & Learning Strategies: | Combination of lectures and practical laboratory sessions. Lectures will take the form of traditional theory and workshop activities. Workshop activities entail interaction with students whilst building programs from scratch using data projector facilities. Laboratory sessions take the form of formative assessment sheets with individual interaction with students |
|---|---|

| Module Aim: | To provide the student with: 1. the problem solving skills necessary for programming 2. the basic concepts of programming. 3. the capability to develop complete programs |
|---|---|

| Learning Outcomes |
|---|
| On successful completion of this module the learner should be able to: |

| LO1 | Comprehend and utilise problem solving techniques to analyse a problem and develop a solution for it; |
|---|---|
| LO2 | Write simple programs based on simple problem-solving algorithms they write; |
| LO3 | Utilise and comprehend core programming concepts |
| LO4 | Create programs to manipulate and store strings; |
| LO5 | Create programs to manipulate arrays; |
| LO6 | Comprehend and apply object-oriented programming concepts such as abstraction, encapsulation, inheritance and polymorphism; |

| Pre-requisite learning |
|---|
| **Module Recommendations** <br> This is prior learning (or a practical skill) that is recommended before enrolment in this module. |
| No recommendations listed |
| **Incompatible Modules** <br> These are modules which have learning outcomes that are too similar to the learning outcomes of this module. |
| No incompatible modules listed |
| **Co-requisite Modules** |
| No Co-requisite modules listed |
| **Requirements** <br> This is prior learning (or a practical skill) that is mandatory before enrolment in this module is allowed. |
| No requirements listed |

## Module Content & Assessment

### Indicative Content

**Introduction to problem solving**
Algorithms & pseudocode; translating pseudocode into program code; Introduction to a relevant language; identifiers, keywords, comments. Data types, variables, assignment statements, constants, arithmetic expressions and operators, operator precedence, using conditional expressions, console I/O. Program control constructs and their uses - sequence, iteration and selection, flow of control;

**Strings**
String manipulation, string classes and methods

**Arrays**
Concepts, declarations, creation, sorting and searching arrays, multidimensional arrays

**Objects**
Abstraction & encapsulation, classes, objects, methods, instance & local variables, scope, method parameters & return types, pass by value parameters, reference variables, access modifiers, object creation, object initialisation & constructors

**Inheritance**
Single inheritance; sub classing, overloading, overriding, this reference, polymorphism, casting, super keyword, parent constructors.

| Assessment Breakdown | % |
|---|---|
| Continuous Assessment | 100.00% |

### Continuous Assessment

| Assessment Type | Assessment Description | Outcome addressed | % of total | Assessment Date |
|---|---|---|---|---|
| Other | Lab Test 1: The student will be assessed on their ability to write a piece of pseudocode and translate it into a simple program | 1,2,3 | 10.00 | n/a |
| Other | Lab 2: The student will be assessed on their ability to write a program that utilizes a selection structure. | 1,2,3 | 10.00 | n/a |
| Other | Lab 3: The student will be assessed on their ability to write a program that utilizes a iteration structure. | 1,2,3 | 10.00 | n/a |
| Other | Lab 4: The student will be assessed on their ability to write a program that utilizes strings. | 1,2,3,4 | 10.00 | n/a |
| Other | Lab 5: The student will be assessed on their ability to write a program that utilizes arrays | 1,2,3,5 | 10.00 | n/a |
| Other | Written Test 1: The students will be given a written test to assess their knowledge of strings. | 1,2,3,4 | 15.00 | n/a |
| Other | Written Test 2: The students will be given a written test to assess their knowledge of object-oriented concepts. | 1,2,3,4,5,6 | 15.00 | n/a |
| Other | Written Assessment on Arrays of Objects & Methods | 1,2,3,5,6 | 10.00 | n/a |
| Performance Evaluation | Active participation in the Lab | 1,2,3,4,5,6 | 10.00 | n/a |

| No Project |
|---|

| No Practical |
|---|

| No End of Module Formal Examination |
|---|

**SETU Carlow Campus reserves the right to alter the nature and timings of assessment**

## Module Workload

| Workload: Full Time | | |
|---|---|---|
| *Workload Type* | *Frequency* | *Average Weekly Learner Workload* |
| Lecture | 30 Weeks per Stage | 2.00 |
| Laboratory | 30 Weeks per Stage | 3.00 |
| Tutorial | 30 Weeks per Stage | 1.00 |
| Estimated Learner Hours | 30 Weeks per Stage | 0.67 |
| | Total Hours | 200.00 |

## Module Delivered In

| Programme Code | Programme | Semester | Delivery |
|---|---|---|---|
| CW_KWCAP_C | [Higher Certificate in Computing](Higher Certificate in Computing) | 1 | Mandatory |