## Introduction

This document contains the ECDL / ICDL Computing Syllabus 1.0 sample test.

The sample test is for use by candidates intending to take the ECDL / ICDL Computing certification test. The sample test aims to give candidates the opportunity to become comfortable with the style, approach, format and structure adopted within the ECDL / ICDL Computing tests.

The sample test must not, under any circumstances, be used in certification testing. The sample test should ONLY be used in an ECDL / ICDL Accredited Test Centre and is not for wider distribution.

The actual content of the ECDL / ICDL Computing test used for certification will vary to ensure appropriate syllabus coverage across all tests. For this reason, candidates should be informed that the sample test will not be an exact reflection of actual certification test content.

The sample test consists of 36 questions giving a total of 36 marks.

The ECDL / ICDL Computing certification test also contains 36 questions and the entire test represents a total of 36 marks. The candidate has passed the certification test if he/she scores 27 marks out of 36 marks. The pass mark for the module is 75%. The duration of the ECDL / ICDL Computing certification test is 45 minutes.

## Naming and Reference Conventions

The following naming and reference conventions have been adopted within the ECDL / ICDL Computing sample tests. References to file names, file extensions, folders, URLs (Uniform Resource Locators), hyperlinks, image links, web pages, e-mail messages, field identification names etc., together with textual insertions are presented in **bold** for ease of identification within the test paper. Naming or insertion actions for text or numeric data should be added without any formatting except where a formatting action is requested as part of the question item. Authorised ECDL / ICDL Testers are required to make these conventions known to Candidates in advance of the sample test.

## Candidate Drive

An allocated location on a Drive should be provided to Candidates for the sample test in ECDL / ICDL Computing. Where support files and folders, work files or an answerfile relevant to the specific sample test are required, these are to be provided on the allocated location on the Drive. The allocated location on the Drive is the property of the Test Centre and is the responsibility of the authorised ECDL / ICDL Tester. Candidates may save their work to removable media provided by the Test Centre. All sample test documentation and removable media must be handed up at the end of the sample test.

## Answerfile Instructions

An answerfile is provided in the answer folder for the sample test in your Candidate Drive. The Candidate should enter his / her candidate identification number near the top of the answerfile. Answers should be entered beside the relevant question number in the answerfile and saved.

## Answer Guide

An Answer Guide for the sample test is contained within the sample test folder.

## Security Warnings

When opening files during the test, depending on security settings, some security warnings may appear or the files may open in protected mode. The Authorised Tester must advise the Candidate accordingly to allow them to use the files if these warnings occur.

## Set-up Instructions

The Authorised ECDL / ICDL Tester needs to ensure that Python 3 (preferably 3.2.3) is installed on each computer and that .py files can be created and run on each computer.




Ref: ECDL / ICDL Computing - Syllabus - V1.0 - SampleMQTB - Python 3 - V1 - 0

## Sample Test 1

***This is a sample test for use by Candidates intending to take the ECDL / ICDL Computing certification test. The sample test aims to give candidates an opportunity to become comfortable with the style and structure of the certification test.***

***The sample test must not, under any circumstances, be used in certification testing.***

The following is the sample test for ECDL / ICDL Computing. This sample test consists of 36 questions with 1 mark available for each question. The total marks available are 36 marks, and the sample test duration is 45 minutes.

1 Which one of the following best defines computational thinking? **[1 mark]**

a. Thinking about computers and how much they cost. ☐
b. Analysing problems and identifying possible solutions. ☐
c. Using a calculator to evaluate a maths equation. ☐
d. Checking for errors in a program before it is released. ☐

2 Which one of the following best describes the term machine code? **[1 mark]**

a. A series of 1's and 0's created by the computer from source code. ☐
b. A pictorial way of representing an algorithm. ☐
c. A number of simple instructions that should be carried out one after the other. ☐
d. A description explaining what a program is designed to do and how it works. ☐

3 Which one of the following best describes the activity of clearly defining the problems to be solved when creating a new program? **[1 mark]**

a. Design. ☐
b. Enhancement. ☐
c. Programming. ☐
d. Analysis. ☐

**Continued…**

## Sample Test 1 (Contd.)

4 You have been asked to design an online menu application for a pizza delivery service. Which one of the following best describes the major steps that need to be addressed in this problem? **[1 mark]**

a. 1. Who will deliver the pizza?
2. What types of smartphones will it be designed for?
3. Will the system remember customers for next time?
4. What types of pizza and side orders will be on the menu? ☐

b. 1. How many different food sections will there be?
2. Will the system remember customers for next time?
3. What types of pizza and side orders will be on the menu?
4. Will customers be able to pay for the pizza online? ☐

c. 1. What colour will the screens be?
2. How many different food sections will there be?
3. What types of pizza and side orders will be on the menu?
4. Will the recipes be for the different types of pizza be on the system? ☐

d. 1. How many buttons should it have?
2. What car will be used to deliver the pizza?
3. Will customers be able to pay for the pizza online?
4. Will the system remember customers for next time? ☐

5 You have been asked to design a computer system that will manage hotel reservations. The system will have to solve challenges such as storing guest details, room number and room type and also check in and check out dates of guests. Which one of the following has a pattern that can be repeated in the system? **[1 mark]**

a. A procedure for entering the name of the hotel on the front of the system. ☐
b. A procedure for creating guest accounts. ☐
c. A procedure for installing the system on a computer. ☐
d. A procedure for uninstalling the system from a computer. ☐

**Continued…**




Ref: ECDL / ICDL Computing - Syllabus - V1.0 - SampleMQTB - Python 3 - V1 - 0

**Sample Test 1 (Contd.)**

6 Which one of the following is a sequenced set of instructions used in computational thinking to solve problems? **[1 mark]**

a. Algorithms. ☐
b. Decomposition. ☐
c. Machine code. ☐
d. Procedures. ☐

7 Which one of the following is a pictorial way to represent a set of instructions to solve a problem? **[1 mark]**

a. Function. ☐
b. Machine code. ☐
c. Flowchart. ☐
d. Boolean expression. ☐

8 Which one of the following is a special type of variable that is used to influence what a subroutine does? **[1 mark]**

a. Float. ☐
b. Function. ☐
c. Parameter. ☐
d. Pseudocode. ☐

9 Which one of the following is a piece of text in the code that explains to people what the code does? **[1 mark]**

a. Specification. ☐
b. Comment. ☐
c. Array. ☐
d. Recursion. ☐

10 Which one of the following is a placeholder for actual values that can be retrieved for use multiple times? **[1 mark]**

a. A comment. ☐
b. A variable. ☐
c. A for loop. ☐
d. A comparison operator. ☐

11 Which one of the variable names below should be chosen to store a customer's private access code details? **[1 mark]**

a. myName. ☐
b. myAge. ☐
c. myPIN. ☐
d. myTotal. ☐

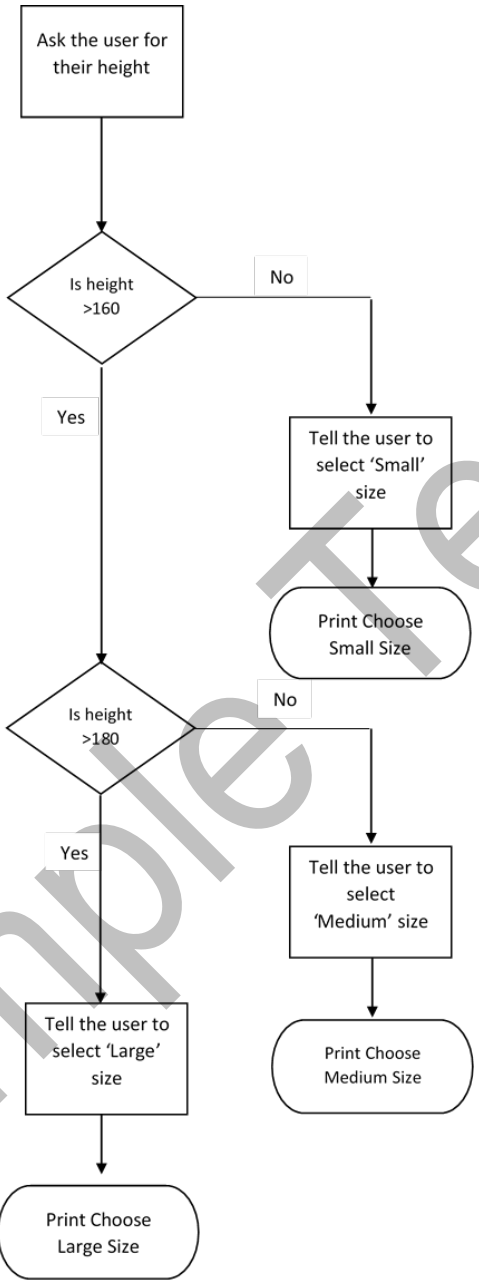
**Continued…**




Ref: ECDL / ICDL Computing - Syllabus - V1.0 - SampleMQTB - Python 3 - V1 - 0

## Sample Test 1 (Contd.)

12 Which one of the following best describes the sequence of operations represented by the flowchart? **[1 mark]**




a. Instructions on how to design three types of size labels. ☐

b. Instructions for a program which prints the height of a user depending on the size they choose. ☐

c. Instructions on how to input your height into a size check machine. ☐

d. Instructions for a program which prints the required size choice depending on the users' height. ☐

**Continued…**




Ref: ECDL / ICDL Computing - Syllabus - V1.0 - SampleMQTB - Python 3 - V1 - 0

**Sample Test 1 (Contd.)**

13 Which one of the following is the Boolean logic expression for "not equal to"? **[1 mark]**

a. Not ☐
b. <= ☐
c. < ☐
d. != ☐

14 Which one of the following describes a piece of code that is run repeatedly under certain conditions? **[1 mark]**

a. Pseudocode. ☐
b. Comment. ☐
c. Boolean expression. ☐
d. Loop. ☐

15 Which one of the following is a type of loop used for iteration? **[1 mark]**

a. Integer. ☐
b. While. ☐
c. Float. ☐
d. List. ☐

16 Which one of the following terms best describes the process of a subroutine dividing a problem into simpler parts and calling itself to solve those simpler parts? **[1 mark]**

a. Debugging. ☐
b. Recursion. ☐
c. Event. ☐
d. Variable. ☐

17 Which one of the following best describes a statement that evaluates an expression as a true or false value to determine what happens next? **[1 mark]**

a. Conditional. ☐
b. Iterative. ☐
c. Complex. ☐
d. Recursive. ☐

**Continued…**




Ref: ECDL / ICDL Computing - Syllabus - V1.0 - SampleMQTB - Python 3 - V1 - 0

**Sample Test 1 (Contd.)**

18 Which one of the following best describes a subroutine that executes an action in a program without returning a value? **[1 mark]**

a. Boolean. ☐
b. Iteration. ☐
c. Procedure. ☐
d. Variable. ☐

19 Which one of the following best describes a subroutine that calculates a value for the program in which it is contained? **[1 mark]**

a. Debugging. ☐
b. Procedure. ☐
c. For loop. ☐
d. Function. ☐

20 The code below is intended to print the time in the format 17:22:56.

```
from time import strftime, gmtime


# Update the code here


print ( strftime() )
```

Which one of the following lines of code should be added to ensure that the time is printed in the required format? **[1 mark]**

a. strftime("%H", gmtime() ) ☐
b. strftime("%M", gmtime() ) ☐
c. strftime("%Y", gmtime() ) ☐
d. strftime("%X", gmtime() ) ☐

21 Which one of the following is a type of error where a construct in the programming language is written incorrectly? **[1 mark]**

a. Syntax. ☐
b. Recursion. ☐
c. Logic. ☐
d. Variable. ☐

**Continued…**




Ref: ECDL / ICDL Computing - Syllabus - V1.0 - SampleMQTB - Python 3 - V1 - 0

**Sample Test 1 (Contd.)**

22 Open the file **Algorithm_Error.docx** from your Candidate Drive. The flowchart algorithm describes a procedure for checking a user's PIN number and locking them out if three incorrect entries are made. However it is missing a program element. Identify the missing program element and select the correct repair from the options below. Close the **Algorithm_Error.docx** file. **[1 mark]**

a. Option A. ☐
b. Option B. ☐
c. Option C. ☐
d. Option D. ☐

23 Open the file **Flowchart.docx**. The flowchart algorithm partially describes the procedure **Rollercoaster Rules**. Complete the flowchart algorithm based on the information provided. Save and close the **Flowchart.docx** file. **[1 mark]**

24 Open the file **Division.py**. Update the short "split the bill" program with the appropriate code to carry out the calculation **bill divided by people**. Save and close the **Division.py** file. **[1 mark]**

25 Open the file **Comments.py**. Insert a comment in the program above the line **name1 = "Juan"** that will tell the reader the next piece of code will **Define the three names**. Save and close the **Comments.py** file. **[1 mark]**

26 Open the file **Initialising_String.py**. Insert code in the program below the line **# define and initialise myPet** that will define a variable called **myPet** and initialise it using the pet name **Felix**. Save and close the **Initialising_String.py** file. **[1 mark]**

27 Open the file **Assign_Value.py**. Insert code in the program below the line **# assign value to salePrice** that will assign the value **24** to the variable called **salePrice**. Save and close the **Assign_Value.py** file. **[1 mark]**

28 Open the file **Using_Integers.py**. Edit the code in the program so that the number **24** assigned to variable **myNumber** is given an **integer** data type. Save and close the **Using_Integers.py** file. **[1 mark]**

29 Open the file **Seasons.py**. Insert code which will populate an aggregated data variable called **mySeasons** and populate it as a tuple with the names of the four seasons **Spring, Summer, Autumn, Winter**. Save and close the **Seasons.py** file. **[1 mark]**

**Continued…**




Ref: ECDL / ICDL Computing - Syllabus - V1.0 - SampleMQTB - Python 3 - V1 - 0

**Sample Test 1 (Contd.)**

30 Open the file **Data_Output.py**. The program asks the user what height they are. Update the program to output the user's height to the screen. Save and close the **Data_Output.py** file. **[1 mark]**

31 Open the file **Boolean.py**. Modify the program so that the print output will return **That number is in the teens** if the number input by the user is in the teens (e.g. less than 20 and greater than or equal to 13). Save and close the **Boolean.py** file. **[1 mark]**

32 Open the file **If_Statement.py**. Insert code into the program which uses an If…then…else conditional statement to check if a student's grade result is **greater than or equal to 75, or less than 75**. Save and close the **If_Statement.py** file. **[1 mark]**

33 Open the file **Function.py**. Modify the program to define a function called **subtractNumbers**, which will subtract two numbers taken in from the user, then return the result of the calculation. Save and close the **Function.py** file. **[1 mark]**

34 Open the file **Random.py**. Modify the dice roll program to use the random library. Use the appropriate function in the random library to generate a random number between **2** and **12**. Save and close the **Function.py** file. **[1 mark]**

35 Open the file **Syntax.py**. Identify and fix one spelling and one punctuation error in the code. Save and close the **Syntax.py** file. **[1 mark]**

36 Open the file **Logic.py** Identify and fix one logic and one data type error in the program. Save and close the **Logic.py** file.

Save and close all open files and close any open applications. **[1 mark]**

**This is the end of the test.**

**If you have time, check the work you have done.**




Ref: ECDL / ICDL Computing - Syllabus - V1.0 - SampleMQTB - Python 3 - V1 - 0