Technology

|  |  |
| --- | --- |
| **AS91896** | **2.7:** Use complex programming techniques to develop a computer program **(6 credits, Internal)** |

|  |  |  |  |
| --- | --- | --- | --- |
| **Learning Area:** | Digital Technology | **Context:** |  |
| **Assessor:** | AHR | **Learner’s Name:** |  |

**Introduction:**

This standard of work will take approximately 10 weeks of class and homework time.  You are required to:

1. writing code for a program that performs a specified task
2. using complex techniques in a suitable programming language
3. setting out the program code clearly and documenting the program with comments
4. testing and debugging the program to ensure that it works

**Notes from the standard:**

***Achieved Criteria***

* writing code for a program that performs a specified task
* using **advanced techniques** in a suitable programming language
* setting out the program **code clearly** and **documenting** the program with **comments**
* **testing** and debugging the program to ensure that it works on a sample of **expected cases**.

***Merit Criteria***

* **documenting** the program with **appropriate** **variable names** and **comments** that describe code function and behaviour
* following common conventions for the chosen programming language ([Python Conventions](https://realpython.com/python-pep8/))
* **testing** and debugging the program to ensure that it works on a sample of both **expected cases** and **boundary cases**.

***Excellence Criteria***

* ensuring that the program is a **well-structured**, **logical** response to the specified task
* making the program **flexible** and **robust**
* **comprehensively testing** and debugging the program.

***Program requirements:***

* uses **variables** storing at least two types of data (e.g. numeric, text, Boolean)
* uses sequence, selection and iteration **control structures** (e.g. conditionals, loops)
* **takes input** from either a user, sensors, or other external source
* **produces output**
* uses **two** or more **advanced programming techniques**.

***Advanced programming techniques:***

* **modifying** data stored in **collections** (e.g. lists, arrays, dictionaries)
* storing **multidimensional** data in **collections**
* **creating** methods, **functions**, or procedures that use parameters and/or return values
* responding to **events** generated by a graphical user interface (**GUI**)
* using non-trivial **string manipulation**
* using additional **non-core libraries**.

***Flexibility and Robustness:***

* using actions, conditions, control structures and methods, functions, or procedures effectively
* checking input data for validity
* correctly handling expected, boundary and invalid cases
* using constants, variables and derived values in place of literals.

***Test Cases:***

* **Expected** test cases are for testing that your program correctly handles users entering valid data
  + e.g. age inputted as a number from 0 to 117
* **Boundary** test cases are for testing that your program correctly handles the edge cases of your data
  + e.g. age >= 18, your edge cases would be age being inputted as 17, 18, and 19
* **Invalid** test cases are for testing that your program correctly handles invalid data
  + e.g. age inputted as -10, 1,000, “ten”, True, or left blank

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **Std** | **Tchr** | **Assessment evidence** | **Assessment strategies** | **Feedback** |
| **A** |  |  | My code fulfils the requirements laid out in the brief | Written Code and video of it running in portfolio |  |
|  |  | My code uses at least two advanced techniques | Written Code |  |
|  |  | My code is laid out in a clear manner and includes comments | Written Code |  |
|  |  | I have tested my code to ensure that it works on some expected inputs | Testing table in portfolio including:   * type of test (expected, boundary, invalid) * inputs * expected output * actual output * screenshots * next steps |  |
| **M** |  |  | My variables all have appropriate, descriptive names, and my code is well documented with comments describing code function and behaviour | Written Code |  |
| **M** |  |  | My code follows python styling conventions ([Python Conventions](https://realpython.com/python-pep8/)) | Written Code |  |
| **M** |  |  | I have tested my code to make sure it works on all boundary cases | Testing table in portfolio including:   * type of test (expected, boundary, invalid) * inputs * expected output * actual output * screenshots * next steps |  |
| **E** |  |  | My code is well structured, well thought out, and is a logical response to the problem requirements | Written code is logical, effective, and a good solution to the brief  Planning and structuring clearly shown and explained in portfolio |  |
| **E** |  |  | My code is flexible and robust | Written Code  Video demonstrating code handling all inputs, and errors effectively |  |
| **E** |  |  | I have tested my code thoroughly, including invalid cases | Testing table in portfolio including:   * type of test (expected, boundary, invalid) * inputs * expected output * actual output * screenshots * next steps |  |

Final Feedback:

Summative Feedback: