**Assessment Practical/Observation**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Student Name** | | Toton Liantoro | | **CIT Number** | CIT 241675 | |
| **Competency Title, Code and Banner Code**  **CRN** | | Apply advanced object-oriented language skills, ICTPRG532, INFT1033.  14126 | | | | |
| **Assessment Type** | | In the workplace  Simulated environment  Other | | | | |
| **Assessment Name** | | **Assessment 1** | | | | |
| **Assessment Date** | | Handed out Thursday 18 August, due midnight Thursday 8 September. | | | | |
| **Student Statement:** This assessment is my own work. Any ideas and comments made by other people have been acknowledged. I understand that by demonstrating this work for assessment, I agree with this statement. | | | | | | |
| **Student Signature** | | Not required for electronic submissions | | **Date** | 08/September/2022 | |
| **PRIVACY DISCLAIMER:** CIT is collecting your personal information for assessment purposes. The information will only be used in accordance with the CIT Privacy Policy. | | | | | | |
| **Assessor Feedback (also complete observation checklist and questions on last page)**  Feedback will be provided via the assessment upload link.  ❑ **Student provided with feedback** | | | | | | |
| **Attempt 1** | | **Satisfactory** | **Not Yet** **Satisfactory** | | **Date** | / / |
| **Attempt 2** | | **Satisfactory** | **Not Yet** **Satisfactory** | | **Date** | / / |
| **Assessor Name** | Don Coutts | | **Assessor Signature** | |  | |

**Information for Students:** You may have two (2) attempts for this assessment.

* If your **first** attempt is not successful, your teacher will discuss your results with you and will arrange a second attempt.
* If your **second** attempt is not successful, you will be required to re-enrol in this unit.

Only one re-assessment attempt will be granted for each assessment item.

**Time Allowed**: 3 weeks.

**Materials Provided by Assessor:** This document, notes and videos in Sessions 1 – 4, plus material in *Additional Resources*.

**Materials and Equipment to be Supplied by the Student**: This entire document with student name, ID, and date.

**Assessment Range and Conditions**: Additional time will be allocated to students who are registered with student support.

Access to techniques: Open book, student notes, subject material on eLearn, this handout.

**Assessment Criteria:** To achieve a Satisfactory result, your assessor will be looking for your ability to demonstrate the specific performance skills detailed below to industry standard.

You must submit your completed assessment **in a single zip file titled Assessment 1** to the Assessment 1 link in the Assessment block by no later than the due date.

Your zip file MUST include this entire document with your details (name, ID, and date) on page 1 filled in.

**Submissions without this document included cannot be marked.**

**Please do not convert this document to PDF.**

**Assessment 1 - specific performance skills required**

1. **Select and implement a sorting algorithm based on one of three sorting algorithms.**

For this task you have been provided with a text file Ass1.txt containing 40 random integers in the range 100 to 200.

You need to design a Windows Forms App (.NET Framework) in C# that reads the numbers from the text file and stores them in a 1D array. You must use an *OpenFile Dialog* to locate the text file, which should be in the same folder as your C# application.

You should then choose **one** of the coded sorting routines discussed in Session04 to sort the array and display the sorted contents in a list box. The list box should **not** have the *Sorted* property set to True.

**NOTE: As we are aware, recent versions of C# use a simple built-in method to sort arrays. However, the competency requires you to hard-code the sorting process from an algorithm.**

1. **Select and implement a search algorithm based on one of three search algorithms.**

For this task you should add some extra functionality to the app in paragraph 1. above to do a search for a value in the sorted array using the search method discussed in Session04.

**NOTE: As we are aware, recent versions of C# use a simple built-in method to do a binary search on an array. However, the competency requires you to hard-code the search process from an algorithm.**

1. **Observe graphical user interface design principles, including documenting the application according to documentation conventions.**

For this task, and for all assessment work that follows in this subject, you should design your form and the code page according to some of the relevant recommendations in the document *GUI Design Principles* in Session04.

**It is quite acceptable to use the code that I gave you in the sessions for this assessment - you just need to modify it to suit the specific assessment requirements. You should always acknowledge where you sourced the code from in the comments block at the top of the code page – for example, “Based on code provided in Session04”.**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Observation Checklist (to be filled in by Assessor)**  **During the demonstration of skills, did the student satisfactorily:** | **Attempt 1** | | **Attempt 2** | |
|  | **S** | **NYS** | **S** | **NYS** |
| 1. Select and implement a sorting algorithm based on one of three sorting algorithms |  |  |  |  |
| 1. Select and implement a search algorithm based on one of three search algorithms |  |  |  |  |
| 1. Observe graphical user interface design principles, including documenting the application according to documentation conventions. |  |  |  |  |