|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Student Name | Kyle Kent | | Student Number | |  |
| Unit Code/s & Name/s | ICTPRG527 Apply intermediate object-oriented language skills | | | | |
| Assessment Name | Class Test | | Assessment Task No. | | AT2 |
| Date of test/exam | 14/09/2018 | | | | |
| **Student Declaration:**  I declare that this assessment is my own work. I am aware of and understand the rules related to assessment as outlined in TAFE Queensland Student Rules and acknowledge that failure to comply with these rules will be regarded as misconduct and will be subject to disciplinary action as outlined. | | | | | |
| Student Signature |  | | | Date |  |
| Assessor Feedback:  Student provided with feedback *(check box when completed)* | | | | | |
| Attempt 1 | Satisfactory | Unsatisfactory | | Date | / / |
| Attempt 2 | Satisfactory | Unsatisfactory | | Date | / / |
| Assessor Name |  | Assessor Signature | |  | |
| Note to assessor: Please record any reasonable adjustment below that has occurred during this assessment. E.g. written assessment given orally; scribe provided. | | | | | |
|  | | | | | |
| **PRIVACY DISCLAIMER:** TAFE Queensland is collecting your personal information for assessment purposes. The information will only be accessed by authorised employees of TAFE Queensland. Some of this information may be given to the Australian Skills Quality Authority (ASQA) or its successor and/or TAFE Queensland for audit and/or reporting purposes. Your information will not be given to any other person or agency unless you have given us written permission or we are required by law. | | | | | |

|  |  |
| --- | --- |
| Assessment rules | Only students enrolled in the unit of competency, the Assessor / Supervisor, and other authorised personnel may enter or remain in the room during a written test / exam. The Assessor/ Supervisor may ask you to produce photographic identification (e.g. student ID card, driver’s licence).  Unless approved by the Assessor / Supervisor prior to the written test / exam (e.g. for open-book exams) you may not bring any devices capable of conveying information relevant to the content (e.g. text books, course notes, mobile phones, pagers, notebook computers, and other devices). You must ensure mobile phones and other electronic devices are turned off prior to the commencement of the written test / exam.  You are required to comply with all directions:   1. Detailed in assessment material supplied; 2. Set out on any notice displayed in the room; and 3. Given by the supervisor.   During a written test / exam session you may not:   1. Communicate with any person other than the supervisor; 2. Assist another person to communicate with another person; and 3. Willingly receive communication from any person except with the approval of the supervisor.   Unless permitted by the supervisor, you may not take from the room any papers or other materials provided for use during the written test / exam.  You are expected to be considerate of other students when entering or leaving the room or when in the vicinity of the room.  If you consider that your performance in the written test / exam has been adversely affected by illness, disability, bereavement or other exceptional circumstances you may apply for special consideration. **For more information, refer to the Student Rules.** |
| Instructions to Student | **Number of Questions:** 6  **Time Allowed:** 4 hours  **Examination Conditions:**  This is a closed book examination; All questions must be attempted. |

|  |  |
| --- | --- |
|  | **Materials to be supplied:**  Examination paper  **Materials to be supplied by the Student:**  Paper for recording answers  **General Instructions:**  You are required to answer each of the questions provided. You must use a black or blue pen to provide answers, not pencil. Sketches, however, may be in pencil.  **Calculators:**  Calculators may be used during this examination. Before the examination commences, all memories must be fully cleared and programs erased.  **Number of Attempts:**  You will receive up to two (2) attempts at this assessment task. Should your 1st attempt be unsatisfactory (U), your teacher will provide feedback and discuss the relevant questions with you and will arrange a date your 2nd attempt. If your 2nd attempt is unsatisfactory (U), or you fail to attend the scheduled date for a 2nd attempt, you will receive an overall unsatisfactory result for this assessment task. Only one re-assessment attempt may be granted for each assessment task, with the exception of Apprentices or Trainees who are permitted an additional supplementary assessment. **For more information, refer to the Student Rules.** |
| Instructions for the Assessor | This is a closed book examination.Remind students of TAFE Queensland Student Rules assessment requirements. |
| Submission details  (if relevant) | Students are to submit written answers to the teacher at the end of the exam. |
| Note to Student | An overview of all Assessment Tasks relevant to this unit is located in the Unit Study Guide. |

Answer the following in paragraph format in roughly 100 words per question:

1. List and briefly describe **the different control structures used to explain algorithms**.

1. Describe **data structures** giving examples.

1. Explain the following aspects of **object**-**oriented** **programming**:

Classes and Objects

Instance

Abstraction

Inheritance

Polymorphism

Encapsulation.

Think of objects as real-world objects. All objects contain properties and methods. The properties determine what an object knows, so the data within that object. Its methods establish what the object can do with this data. All objects have classes. A class is an outline of an object. It specifies what goes into an object.

Each time an object runs, it is an instance of that object. The creation of any object is an instance. Where an object and class define a group of things, an instance is more specific and only defines a singular item from that group.

Abstraction is meant to display only relevant information within an object or class. It is designed to give an overview of that class or object, hiding any information that is irrelevant.

Inheritance is a concept in OOP programming that allows an object to obtain the properties of another if it is within the same class.

Polymorphism allows a method to be used numerous times, rather than only once.

Encapsulation means exactly what It says; It encapsulates data and methods, merging the two together.

1. Java and C# are the most popular **object**-**oriented** **programming** **language** in the market today, what makes them different from other non-object-oriented programming language?

One of the main reasons Java so popular is its diverse compatibility. Java can run on any OS, so long as the system has a Java environment on it. What makes them different from non-object-oriented languages is that they are object orientated. OOP revolves around objects, everything is carried out by using objects whereas non-object-oriented programming uses procedures. But another thing that makes them so different is their concepts OOP has concepts like inheritance, encapsulation, abstraction and polymorphism which procedural programming cannot do.

1. What aspects and criteria would you apply in considering the **design** of a **Graphical** **User** **Interface** **(GUI)** for an application for use by client users?

GUI definitions would need to be considered. Another thing to consider is what the client wants. To avoid recreating the GUI the client must be involved with the design of the GUI to ensure it meets their requirements. The right controls must also be considered, such as whether the GUI will feature buttons, radio buttons or grouped buttons. The right button must be selected to suit the scenario. Sensitive information must also be considered, implementing a password field over a text field for potentially sensitive information.

1. What are the techniques to **document** **applications**?

Documenting all work done on an application is a must. Throughout the course of a project, testing results, issues, flowcharts, program lists, and much more must be documented somehow. Software methodologies have their own procedures to application documentation. Agile focuses very little on documentation, where as a Waterfall methodology places importance on documentation. Regardless of methodology, diagrams for any databases being used will need to be created, diagrams of the applications layout, a preliminary analysis report and a project plan must all be made.