Assessment Task - Written

Student Name	Kyle Kent	**Student Number**	465510139
Unit Code/s & Name/s	ICTPRG523 Apply advanced programming skills in another language		
Assessment Type	Written		
Assessment Name	Programming Assignment	**Assessment Task No.**	AT1
Assessment Due Date	17/9/18	**Date submitted**	
Assessor Name			

Student Declaration: I declare that this assessment is my own work. Any ideas and comments made by other people have been acknowledged as references. I understand that if this statement is found to be false, it will be regarded as misconduct and will be subject to disciplinary action as outlined in the TAFE Queensland Student Rules. I understand that by emailing or submitting this assessment electronically, I agree to this Declaration in lieu of a written signature.

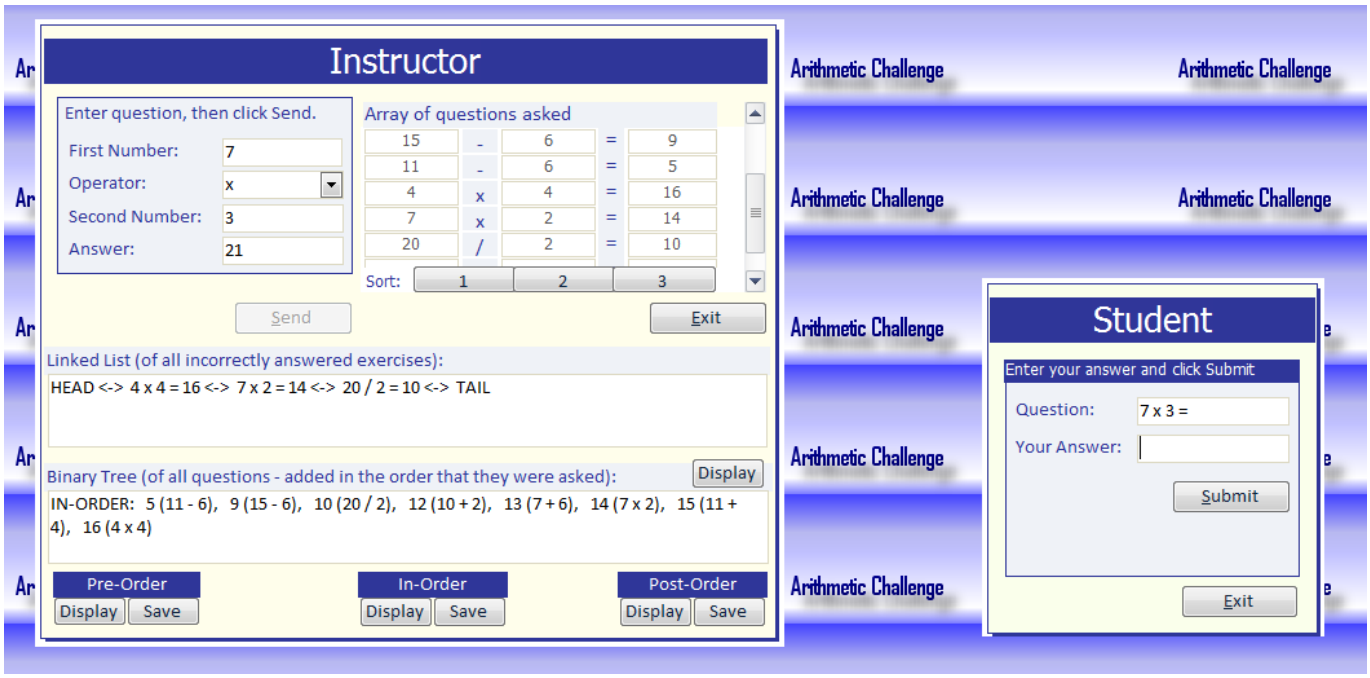
Student Signature	K Kent	**Date**	9/09/2018

Instructions to Student	**Learning Support** Additional support is available to help you achieve your learning goals. Speak to your teacher or a Learning Skills Centre team member if you feel that you may benefit from some extra support. The Institute provides extra support through the Disability Support Unit and the Learning Skills Centre. RPL (Recognition of Prior Learning) is available for this unit. Speak to your teacher/assessor to check if you qualify for RPL. **Conditions of Assessment** You will need to complete the learning and undertake all assessments satisfactorily to be deemed competent. You are responsible for complying with all assessment item instructions; submission and collection requirements; undertaking assessment tasks honestly and retaining a copy of all assessment items. You must submit assessment items by the due date, unless an extension has been granted by your teacher. Failure to submit assessment items by the due date will result in a "did not submit" being recorded and depending on your circumstances, you may be granted one final resubmission. To be judged competent in this assessment item the student is required to demonstrate competence in all indicators shown in the marking guide. **The Classroom as a Simulated Work Environment** Students must be aware and take responsibility for the problems of working in a shared IT environment. Problems such as noise levels, production flow, interruptions and time variances are common to workplaces. In the simulated environment provided in the classroom these problems can take the form of: • Other students who continually ask questions or talk aloud while thinking • Fire drills, projector not working, printers running out of paper or toner cartridge • Miscalculating how much work you can do in one day, missing classes and so on. Some things are unavoidable and you must devise strategies to overcome them, for example, we cannot stop students from asking questions or entering at exiting the class. Other things are unpredictable (e.g. fire drills). You need to be aware and plan and organise your work allowing some extra time for unavoidable and unpredicted events.

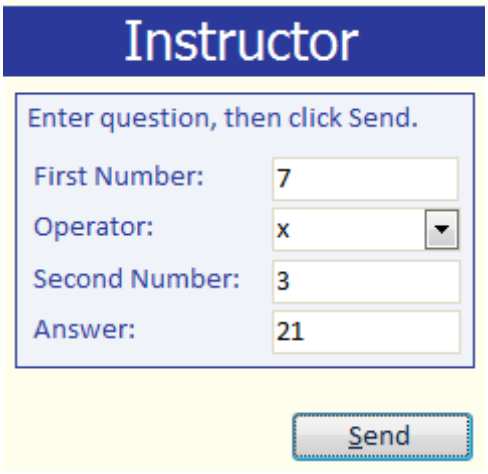
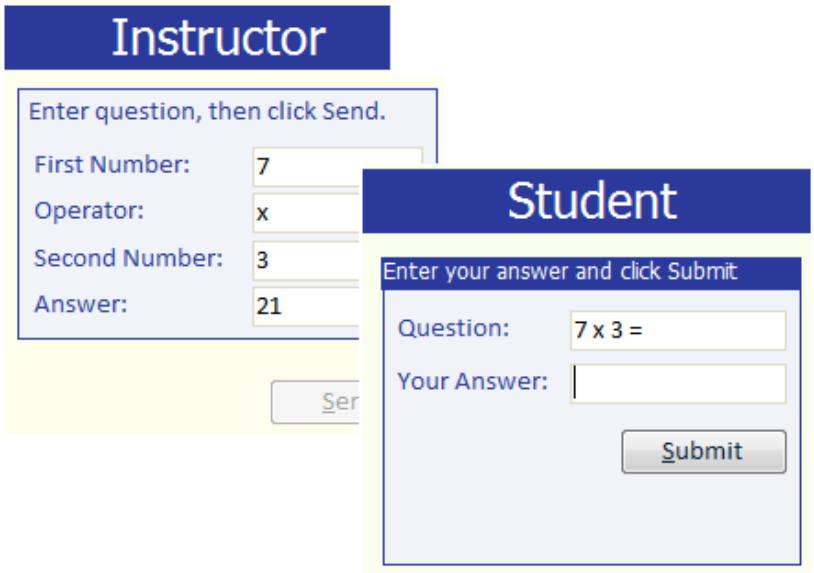
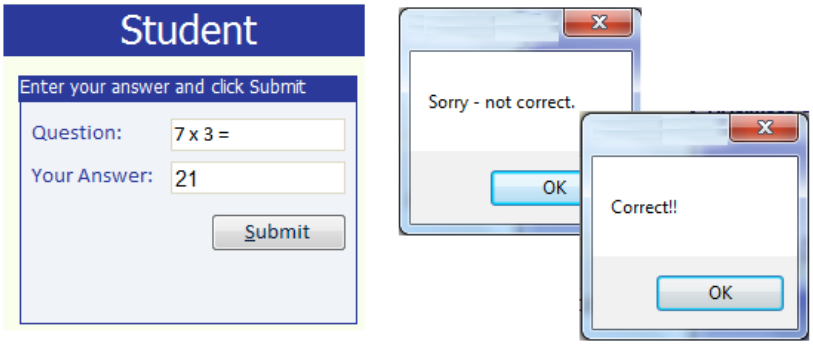
	Assessment Criteria: To achieve a satisfactory result, your assessor will be looking for your ability to demonstrate key skills/tasks/knowledge to an acceptable industry standard. Refer to the marking criteria document for a detailed list of items. **Number of Attempts:** You will receive up to two (2) attempts at this assessment task. Should your 1st attempt be unsatisfactory (U), your teacher will provide feedback and discuss the relevant sections / questions with you and will arrange a due date for the submission of your 2nd attempt. If your 2nd submission is unsatisfactory (U), or you fail to submit 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.**
Submission details	Submit your assessment to the allocated dropbox in **Connect** or to the allocated network folder. Your teacher will provide all the details for the submission system or network. Your assignment must be saved with your surname_student number_unit/cluster_AssessmentNumber. For example: • surname_1234567890_ICTPRG523_1 For re-submissions, an "R" must be added to the file name. For example: • surname_1234567890_ ICTPRG523_1_R The Marking Criteria Sheet must be signed and submitted with your work.
Instructions for the Assessor	To be judged competent in this assessment item the student is required to demonstrate competence in all indicators shown in the marking guide and need access to: • The software development environment • Technical documentation
Note to Student	An overview of all Assessment Tasks relevant to this unit is located in the Unit Study Guide.

Instructions to Students

You have been asked by a group of teachers to build a network-based Arithmetic Challenge Game. The functionality is provided in the prototype they have prepared.



Instructions are as follows:

The **Instructor** enters an arithmetic question (possible operations include: + - x /) For instance: 5 + 8 =, 10 / 2 = … The answer is automatically calculated and presented to the instructor.	
The **Instructor** then clicks on their **Send** Send button. The **Send** button then becomes inactive. The **student** who is on another networked computer sees the question appear in their **Question:** text field.	
The **student** would enter their answer into the **Your Answer:** text field. The **student** would then click on the **Submit** Submit button so the system can compare the **student's** answer with the correct answer. The **student's** **Submit** button would become disabled and a popup message box would indicate to the student whether their answer was correct or not.	

As questions and answers are entered and checked they are saved within different data structures:

- **All** questions and answers are stored in an **array of objects**. The content of this array is then displayed on the top right of the **Instructor's** screen in a **table**.

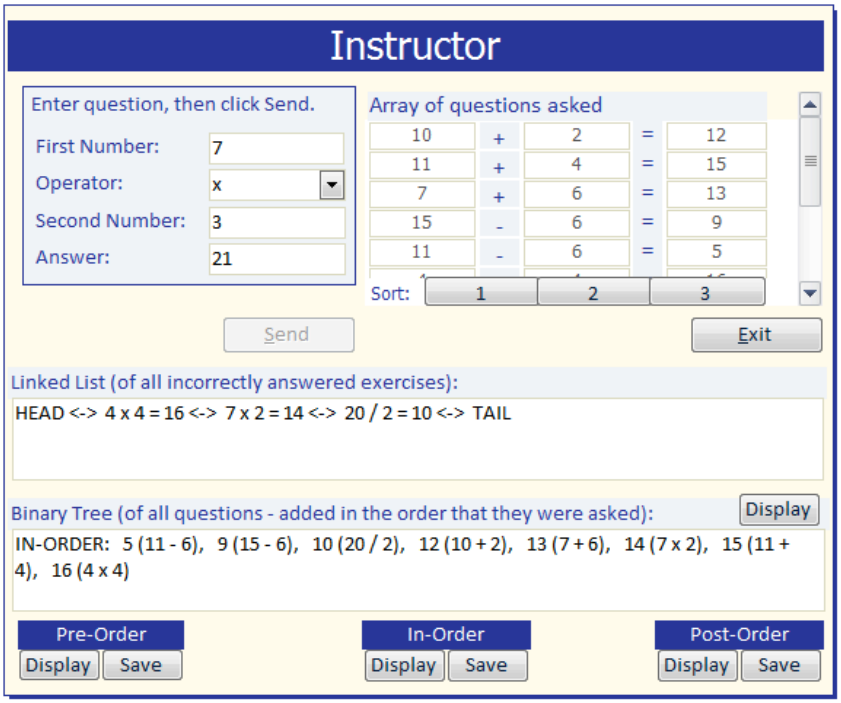
  Three different sort algorithms are to be implemented and can be run via the **Sort** buttons - so that the Array(s) can be sorted by operation (+ - x /). *(For visual effect, you may wish to have Sort 1 sort in ascending order, Sort 2 in descending order, and Sort 3 in ascending order.)*

- All **incorrectly answered questions** are to be added to a **doubly linked list**. The content of this doubly linked list is presented in a text area in the centre of the **Instructor's** screen. Include code for **searching** the doubly linked list.

- All questions used are added in order of entry to a **binary tree,** sorted by their respective answers. The content of this binary tree can be displayed in a text area at the bottom of the **Instructor's** screen. Buttons on the bottom of the screen allow the content of the binary tree to be displayed in Pre, Post or In–Order and to be saved in corresponding text files.

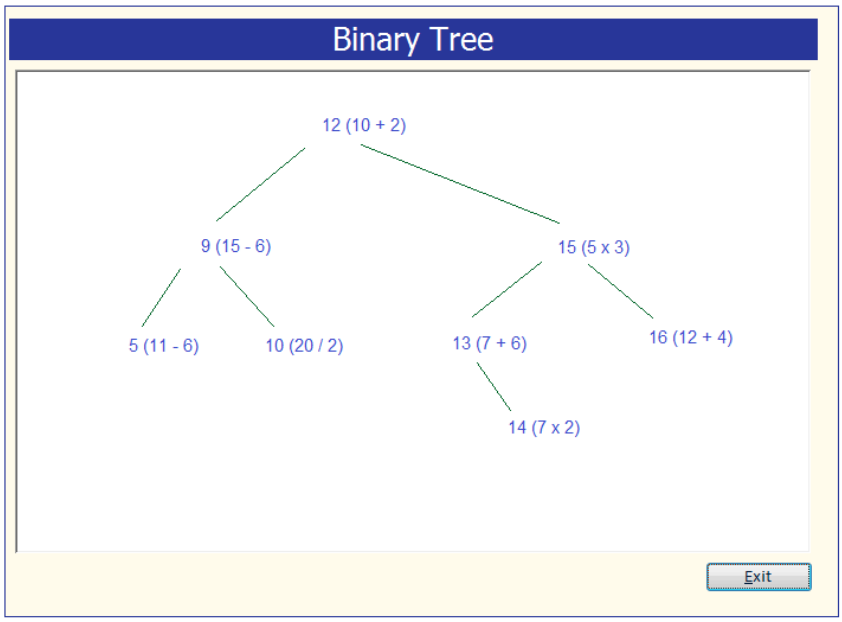
  Include code for **searching** the binary tree.

  Apply a **hashing algorithm** to the output of each of these files.



An **extension activity** would be to graphically display the binary tree - as a tree - within a graphics screen.

Note: This task is <u>not</u> required for you to achieve competence in this assignment.



Programming Obligations:

In relation to this project, you are required to implement and demonstrate clear competence in the following aspects of programming:

1. Design and build the application based on the specifications on the previous pages with appropriate low coupling and high cohesion, and object-oriented programming techniques wherever possible. Outline / present a procedure for the development of your application. (Note that this procedure needs to account for this being developed in and for a GUI environment.). Includes testing and debugging issues.

2. Appropriate design and application of the following data structures:
   - Arrays of objects
   - Double Linked Lists of objects
   - Binary Trees of objects
   - Hash table
   - …plus a search facility for each.

   Includes testing and debugging issues. (Elements 1, 2, 4, 5, 6, 7 & 8)

3. Appropriate implementation and application of **three (3) common sort algorithms**.

   (Elements 2, 5, 6, 7 & 8)

4. Appropriate implementation and application of **program communication**:
   - **inter-process communication** through at least one mechanism.
   - **operating system 'signals'** to be captured and responded to. (Elements 3, 4, 6, 7 & 8)

5. Utilise a third-party library in the construction of your application indicating how and where you have used the library, and how and where you have referenced the third-party library's documentation. For instance, you might locate and utilise a third party library that:
   - graphs the matching of the associated pairs, or:
   - assists with the graphical presentation or analysis of your doubly linked list, binary tree and/or hashing algorithm.

   Includes testing and debugging issues.

6. Clear evidence of appropriately planned and structured **testing** and **debugging**. (Element 7)

7. Appropriate internal and external technical **documentation:** (Element 6)

   - add **code headers** presenting section headings, author, version control, etc.
   - add **useful comments** throughout your code that are likely to prompt others as to the purpose and focus of each code segment and/or to remind you of your current thinking / logic at a later time.
   - generate applicable **external documentation** such as class diagrams and programming documentation
   - *outline in detail and provide evidence of how you have managed version control throughout the development of your project solution.*

8. Maintain appropriate ongoing communication with your manager and client *(including emails)* to ensure you stay on track and receive and action client and manager feedback. The technical jargon and detail within your emails needs to be appropriate to your recipients. Specifically, you would be required to:

   a) Email your manager before the commencement of this project to secure the project specification.

   b) Prepare an email to your manager and then (separately) to your client regarding your proposed third party library, notifying them about the associated licence agreement. The client will need to understand the benefit(s) of the inclusion of this library in their project.

   c) Prepare a closing email to your manager and then (separately) to your client regarding the success of your project development endeavours. You should suggest any areas of concern or risk to your manager for inclusion in the development team's risk matrix. (Element 6)