



Marsden State High School

Business Education- 11ITS

Semester
1
Date Given
Week Beginning 12 May 2009
Due Date
Week Beginning 26 May 2009

Familiar Application
Problem Solving
Communication

Units of Competency
ICAU1128B ICAU2005B ICAU2006B ICAU2013B ICAD3218B

Name:
Form:
Teacher: WATT/FOUKAS
Head of Department: FURLAN

Assessment: 2 (Formative) Short Written Submission

Authorship Declaration

I _____ claim authorship of the attached works. The works acknowledge all sources and do not breach copyright and plagiarism policy.

Date Submitted: _____

Students Signature: _____

- Task Conditions**
- Program: Delphi 7, Microsoft Word
 - Time: 3 Weeks class time provided
 - Words: 600-800
 - This item must be submitted no later than 5 days after the week beginning date as teacher feedback (given as part of the learning process) will invalidate this assessment. Failure to do so will result in a Non-submission.
 - Failure to complete this item will result in automatic withdrawal of credit for this semester. i.e. The teacher will have insufficient evidence to make a valid judgment about coverage of this course of study.

Assessment 2 11 ITS

Elements	Date	Signature
<p>ICAU1128B Operate a personal computer</p> <ol style="list-style-type: none"> 1. Start the computer <ol style="list-style-type: none"> 1.1 Check <i>peripheral device</i> connections for correct position 1.2 Switch on power at both the power point and <i>computer</i> 2. Access basic system information <ol style="list-style-type: none"> 2.1 Insert user name and password as prompted and note access, privacy, security and related conditions of use displayed on introductory screens 2.2 Navigate through the <i>operating system</i> to access <i>system information</i> to identify system configuration and application versions in operation 2.3 Use <i>on-line help functions</i> as required 3. Navigate and manipulate desktop environment <ol style="list-style-type: none"> 3.1 Create and customise desktop icons 3.2 Select, open and close desktop icons to access <i>application programs</i> 3.3 Manipulate application windows and return desktop to original condition 4. Organise basic directory/folder structure and files <ol style="list-style-type: none"> 4.1 Create and name directories and subdirectories 4.2 Identify <i>attributes</i> of directories 4.3 Move subdirectories between directories 4.4 Rename directories as required 4.5 Access directories and subdirectories via different paths 5. Organise files for user and/or organisation requirements <ol style="list-style-type: none"> 5.1 Use <i>system browser</i> to search drives for specific files 5.2 Access the most commonly used types of files in the <i>directories</i> 5.3 Select, open and rename groups of files as required 5.4 Move files between directories 5.5 Copy files to <i>disk</i> 5.6 Restore deleted files as necessary 5.7 Erase and format <i>disks</i> as necessary 6. Print information <ol style="list-style-type: none"> 6.1 Add a printer if required and ensure correct <i>printer settings</i> 6.2 Change the default printer if appropriate 6.3 Print information from an installed printer 6.4 View and delete progress of print jobs as required 7. Shut down computer <ol style="list-style-type: none"> 7.1 Save any work to be retained and close all open application programs correctly 7.2 Shut down computer correctly 		
<p>ICAU2005B Operate computer hardware</p> <ol style="list-style-type: none"> 1. Identify computer hardware components <ol style="list-style-type: none"> 1.1 Identify <i>external hardware components</i> and <i>peripherals</i> 1.2 Identify <i>internal hardware components</i> 2. Understand the inter-relationship between computer hardware and software <ol style="list-style-type: none"> 2.1 Describe the functions of computer <i>hardware</i> and associated <i>OH&S standards</i> and <i>environmental considerations</i> around <i>hardware</i> use and disposal 2.2 Describe the function of a computer operating system 2.3 Describe the boot process 2.4 State the relationship between an application program, the operating system and <i>hardware</i> 2.5 State the general differences between the different <i>computer platforms</i> and their respective operating systems 2.6 Draw a simple block (schematic) diagram showing the interconnection of the various components of a computer 3. Use computer input equipment <ol style="list-style-type: none"> 3.1 Follow <i>OH&S standards</i> and <i>organisational policies</i> and <i>procedures</i> when using <i>computer input equipment</i> 		
<p>ICAU2006B Operate computing packages</p> <ol style="list-style-type: none"> 1. Use appropriate software <ol style="list-style-type: none"> 1.1 Select <i>software</i> appropriate to perform activity 1.2 Use <i>software</i> to produce required outcome using a range of features related to the activities 1.3 Save documents in appropriate directories/folders 2. Access, retrieve and manipulate data <ol style="list-style-type: none"> 2.1 Select and access <i>files</i> 2.2 Amend and save <i>files</i> according to requirements 2.3 Produce documents and <i>files</i> that meet organisational needs 2.4 Save <i>files</i> in appropriate directories/folders 2.5 Exit <i>software</i> correctly without loss of <i>data</i> 3. Access and use help functions within each application <ol style="list-style-type: none"> 3.1 Identify the help resources available for basic difficulties with the <i>software</i> 3.2 Access user help documentation and other resources for basic difficulties with the <i>software</i> 4. Use keyboard and equipment <ol style="list-style-type: none"> 4.1 Follow <i>OH&S standards</i> and regulations to avoid injury or illness 4.2 Use wrist rests and document holders where appropriate 4.3 Use monitor anti-glare and radiation reduction screens where appropriate 4.4 Ensure user <i>equipment</i> is maintained and free from defects that could cause injury 		
<p>ICAU2013B Integrate commercial computing packages</p> <ol style="list-style-type: none"> 1. Determine work requirements <ol style="list-style-type: none"> 1.1 Identify the requirement of the task 1.2 Select appropriate <i>software</i> and file formats 2. Produce required data/documents to new format <ol style="list-style-type: none"> 2.1 Create a mailing list using a database, spreadsheet or address book, and merge mailing list with another document 2.2 Use a conversion tool of a <i>software application package</i> to convert data from one <i>format</i> to another to enable additional work on the converted data 2.3 Save data to a new file <i>format</i> 2.4 Import <i>objects</i> from another <i>software application package</i> and modify as required to produce a required outcome 2.5 Export data to another <i>software application package</i> to produce a required outcome 2.6 Create a link between one <i>software application package</i> and another, and use this to update information to a document 		
<p>ICAD3218B- Create User Documentation</p> <ol style="list-style-type: none"> 1. Determine documentation standards and requirements <ol style="list-style-type: none"> 1.1 Determine documentation requirements 1.2 Investigate documentation and industry standards for requirements and determine appropriate application to user documentation 1.3 Design documentation templates using appropriate software and obtain approval from appropriate person 2. Produce user documentation <ol style="list-style-type: none"> 2.1 Conduct a review of the subject system, program, network or application in order to understand its functionality 2.2 Gather existing technical, design or user specifications and supporting documentation 2.3 Create user documentation based on template to record the operation of the subject system, program, network or application 3. Review and obtain sign-off <ol style="list-style-type: none"> 3.1 Submit user documentation to target audience for review 3.2 Gather and analyse feedback 3.3 Make changes to user documentation 3.4 Submit user documentation to appropriate person for approval 		

Assessment 2 11 ITS

Criteria	Standard A	Standard B	Standard C	Standard D	Standard E
Familiar application <ul style="list-style-type: none"> • knowledge • simple application 	The student: <ul style="list-style-type: none"> • recalls a wide range of facts, terminology, methods and procedures, concepts, processes and principles relating to software programming, developing user documentation and communicating with a known audience. • effectively selected and applied knowledge to produce a quality user document. 	The student: <ul style="list-style-type: none"> • recalls a range of facts, terminology, methods and procedures, concepts, processes and principles relating to software programming, developing user documentation and communicating with a known audience. • consistently applied related knowledge to produce a user document. 	The student: <ul style="list-style-type: none"> • recalls facts and terminology and some related methods and procedures, concepts, processes and principles relating to software programming, developing user documentation and communicating with a known audience. • applied related knowledge to produce a satisfactory user document. 	The student: <ul style="list-style-type: none"> • recalls facts, terminology, and related concepts relating to software programming, developing user documentation and communicating with a known audience. • applied knowledge to produce a user document. 	<ul style="list-style-type: none"> • The student recalls some facts and terminology relating to software programming, developing user documentation and communicating with a known audience.
Problem solving <ul style="list-style-type: none"> • analysis • synthesis • evaluation 	The student: <ul style="list-style-type: none"> • developed an effective and efficient solution to an unrehearsed and complex problem • evaluated contexts, inputs, processes and products, with detailed justification and against appropriate criteria. 	The student: <ul style="list-style-type: none"> • developed an effective solution to unrehearsed and complex problem • evaluated contexts, inputs, processes and products, with justification and against appropriate criteria. 	The student: <ul style="list-style-type: none"> • developed a solution to unrehearsed problems • evaluated contexts, inputs, processes and products against appropriate criteria. 	The student: <ul style="list-style-type: none"> • produced a simple or partial solution to the problem • evaluated superficially. 	The student was unable to provide a solution
Communication <ul style="list-style-type: none"> • representing information • using language 	The student: <ul style="list-style-type: none"> • constructed and presented a high quality user friendly document. • used a wide vocabulary with discrimination and applied conventions of language to convey meaning appropriate to the short written task. 	The student: <ul style="list-style-type: none"> • constructed and presented a quality user friendly document. • used a wide vocabulary with discrimination and applied conventions of language to convey meaning appropriate to the short written task. 	The student: <ul style="list-style-type: none"> • constructed and presented a user friendly document. • used vocabulary and conventions of language to convey meaning appropriate to the short written task. 	The student: <ul style="list-style-type: none"> • constructed and presented a user friendly document. • used language to convey meaning, although the meaning conveyed is not always appropriate to the short written task. 	The student: <ul style="list-style-type: none"> • presented a user friendly document. • used language, although meaning is sometimes unclear or inappropriate.

Context:

As programmers we often get carried away with the technical aspect of creating an application and forget that the end user does not necessarily have the expertise to understand and use the product. This is why it's important to create user documentation that is simple, helpful and free from assumptions about user ability level.

Task:

You are required to develop a User Manual, User Documentation and Reference Guide for the "Love Calculator" application you have created in your first assessment item. The guide will include the following sections;

Title Page

Contents Page

User Manual

Getting on the Computer

Locating the Delphi Exe

Running the Program

Using the Love Calculator Application

User Documentation

Understanding how the application works

Explanation of code used

Reference Guide

Let's create your own "Love Calculator"

The Delphi Interface

Programming

Shortcut Keys

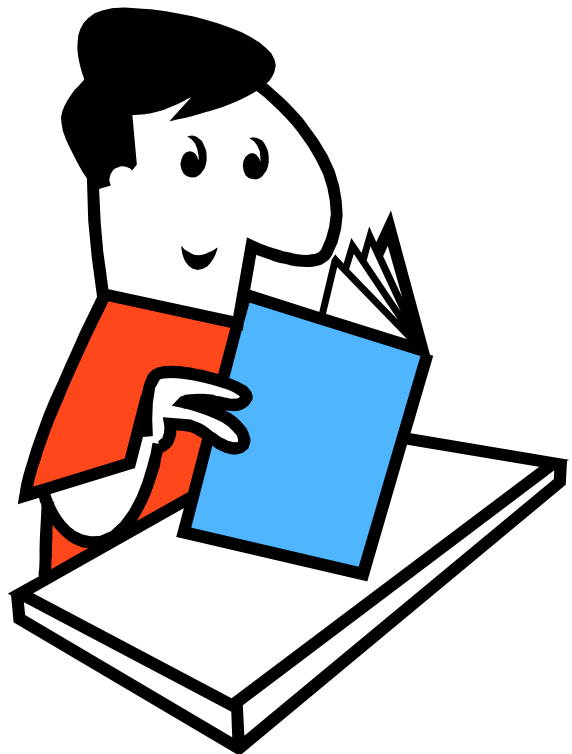
Testing

Errors

Help

Running

Glossary



Submission:

The overall submission should be 600-800 words in length and should incorporate graphics, screen dumps and diagrams to help explain the topics. You will be given 3 weeks class time to complete the task. Formatting of the document should be Arial 12 Font, 1.5 spacing, headers/footers, page numbers, word art and borders.