



Examination Answer Books

12 Pages

Instructions to Candidates

1. Please give the information requested in the boxes below.
2. Enter your name in the space provided to the right, fold over and seal down (using the label provided) to hide your name.
3. The answers must be legibly written. The Examiners can either ignore illegible script or have it typed and the expense of typing charged to the candidate.
4. Write your answers in ink.
5. Begin each answer on a fresh page. Do all rough work in the answer book and cross it through.
6. No part of this book is to be torn out; it must be handed in complete.
7. If more books than one are used, fasten them together using the tags available from the invigilator(s).
8. Every book must be handed in. In no circumstances may answer books, used or unused, be taken out of the examination venue.

Complete, fold and seal
(see note 2 below)

SURNAME: Corner

FORENAME(S): David Jonathan

1	Student Number	DC168717.
2	Title of Examination Paper	Software Development.
3	Module Code	ACCAG017.
4	Date and Time of Examination	09/01/19 5:10 - 7:10.
5	Examination Venue	G42.
6	Seat Number	
7	Number of Answer Books handed in	

List the numbers of the questions you have attempted	Office Use
1a	1
1b	2
1c	4
1d	4+
2a	10
2b	1+1
2c	10
3a	0
3b	0
4a	5
4b	2
4c	2
Total	42

✓
✓
✓
✓
✓
✓
✓
✓
✓
✓
✓

195 ✓

447 ✓

1a. This data structure ^{is data} is ~~similar to stacks in terms of its operation~~ ~~where~~ the data is inserted in a first in first out method, ~~the~~ the queue is open on ~~both~~ both ends - one end is open to insert new data and the other is to remove current data. An example of where this data structure would be used is in a program that displays ~~train train times~~ ~~before it reaches~~ the next departure of a train in a station, it will show the time when the next train will arrive, then soon as it departs the programme will remove the time from the memory then show the next ~~time~~ train time on the departure board.

1b. A Stack data structure is a data type which has a ~~fixed~~ predefined capacity overall, where data enters and exits in a Last in, first out method, unlike the Queue Data Structure, stacks can only insert and remove data from the top. An example of where this method is used is in a program which has an undo/redo function, this is accomplished by keeping a certain amount of text changes in a stack. This method is appropriate for this as it retains a specific amount of changes in the data structure.

1c. This is a ~~method~~ method that divides a problem into sub-problems, after ~~one~~ one recursion takes place the next recursion uses the output from the recursion before as the input. Quicksort may be used instead ~~because~~ because it is cache efficient as cache performs quicker than Random access memory, ~~so~~ that other sorting algorithms depend on. Quicksort is known as a 'divide and conquer algorithm', it ~~works~~ operates in the following stages:

1. Pick a 'pivot' element.
2. 'partition' the Array into three sections which are:
 - All elements in this section is less than the pivot
 - The pivot itself
 - All elements within this section are greater than or equal to the pivot
3. Finally apply the quicksort algorithm to the first ~~and~~ and third section recursively.

1d. This type of Paradigm is based on the concepts of methods and objects. Orientated programming Consists of the below four principles:

Polymorphism - This is the condition of occurrence in a number of forms.

Abstraction - Process of filtering out characteristics that are not needed.

Encapsulation - Hides the internal state and behavior of an object.

Inheritance - Ability to create a class that inherits attributes from an existing class.

2a. (Please check code drop box).

2b. ~~The example code was returning false as dbOutput 1 and dbOutput 2 are declared as 'double' values. In order for the program to return the value 'True' the variables needed to be declared as integers.~~

The example did not return 'True' because the output value was in decimal form. In order to rectify this we needed it to be a whole number.

2c. (Please check code drop box).

4a. - Whitebox testing - This tests the Software's internal structure, design, and coding. This type of testing is visible to the tester. Its primary focus is to verify the flow of inputs and outputs through the application that improves design and usability, strengthening and security. This type of testing is known as Clearbox, open box, Structural testing.

Blackbox Testing - This is found as behavioral testing. Black box testing is a Software testing method in which the internal structure, design and implementation is not known to the person testing hence why the method is called 'Black box'. ~~is that the tester~~ is methodically the tester is in a black box and is unable to see.

Grey box testing - This is a combination of white box and black box testing. The aim of this particular test is to search for defects if any due to improper structure or improper use of Application's

4b. This breaks the program into pieces and subjects each piece to a series of tests, usually these tests are run in separate programs such as various GUI's [↑] depending on the programming language:

Command-lines

Tests are usually run periodically or often after a change has been made to the source code, ~~these tests are usually run~~
The sooner the tests have been made the quicker the issues can be identified and patched.

4c. Incremental development - This is ~~like~~ a build model where ~~the~~ ^{the} product is designed, implemented and tested incrementally until the product is fully completed, this ~~development~~ ^{method} consists of development and maintenance and the product is ~~finished~~ ^{finished} when it ~~meets~~ ^{meets} the requirements.

Iterative development - This focuses on an initial, simplified implementation, which progressively gains more complexity and broader feature set until the final system is complete.