BCS Higher Education Qualification

Certificate/Diploma/Profession Graduate Diploma

April 2022

EXAMINERS' REPORT

Examination title

General comments

Questions Report:

Qu.	Comment
A1	
a)	Most candidates were able to establish a stackEmpty() function that returned a Boolean value. However, most of the candidates opted for a much longer function using an IF statement rather which could have been shortened to be more efficient.
b)	Some candidates were able to establish a stackFull() function that returned a Boolean value, others struggled to grasp the idea of how to check if the stack was full. However, most of the candidates opted for a much longer function using an IF statement rather which could have been shortened to be more efficient.
c)	Most candidates struggled to grasp the concept of adding a value to the stack using a push() function. Candidates should have used the previously identified functions to check if the stack was full before adding in a new item.
d)	Most candidates struggled to grasp the concept of returning the top value of the stack using a pop() function. Candidates should have used the previously identified functions to check if the stack was empty before returning the top value of the stack.
A2	
a)	Most candidates were able to state four advantages of using functions in programming, with most scoring well in this question. A lot of candidates however did write the same advantage multiple times, just worded in a slightly different way, answers should have been significantly different from the others.
b)	Most candidates struggled to grasp the idea of a multiplication table, with most not being able to write pseudocode that included a counter and a while loop to run through the full times tables.
c)	Candidates generally scored well on this question and were able to explain both iteration and recursion, although most failed to identify the main benefits of iteration when writing in an imperative programming language. Answers should have included the overheads in both time and space explaining that programmers prefer iteration as it is often executed faster and uses less memory that recursion.
A3	
a)	Not answered

b)	Not answered
c)	Not answered
A4	
i.)	Not answered
ii.)	Not answered
iii.)	Not answered
iv.)	Not answered
v.)	Not answered

Question number: B5

This question on Hashing was the least popular choice amongst candidates.

Part a) Many candidates found difficulty in writing out the hash table given the key values. In many instances the candidates failed to continue beyond (correctly) identifying the index before and after resolving collisions, losing marks in the process.

Part b) Most were able to write the hash function that returns an index value. In many cases candidates went beyond computing the index value and spent a good deal of time writing an algorithm to present a hash table. (Gaining no extra marks).

Part c) This proved problematic for most. Very few candidates could give a narrative on the process of searching for a key.

Question number: B6

This question on flowchart representation had two parts

Part a) Drawing a flowchart. Almost all candidates who attempted this question were able to construct a flowchart. The flowchart symbols used did not always follow the ISO: 1028 standards and subsequently lost some marks for ambiguity. Using non-conventional symbols often result in the nodes being unclear.

Part c) Most candidates gave good answers to the construction of a sub routine.

Question number: B7

This question on software was in four parts.

Part a) Most candidates were able to describe the difference between system and application software.

Part b) The difference between open source and closed source, caused some confusion between open source being largely described as freeware where open source was defined as fee paying. Whilst it can be a feature it is not the only defining feature.

Part c) Most were able to give an example of a product where there are alternatives from open/closed source versions. A few candidates did not answer the question posed but gave examples of a different product from each category.

Part d) Most candidates were able to explain a range of factors to be considered in choosing between open and closed source application software.

Question number: B8

This question on project management had three parts.

Part a) The question asked for the range of documentation that would be supplied by a project manager to them. Many candidates' misinterpreted this as the documentation they would give to team members.

Part b) A majority of candidates were able to give a good explanation of documentation to be put forward for development activities. For those candidates who mis-interpreted part a. Difficulty arose in distinguishing between the answers already given and often simply repeated the same answers, subsequently losing marks.

Part c) A majority of candidates found this question difficult. A few did correctly mention test plans and cases and time sheets. A very small minority of candidates mentioned quality criteria

Question number: B9

This question on software reuse had two parts.

Part a) Most candidates were able to answer with at least three descriptions. However, many could not provide more than one or two benefits of software reuse.

Part b) This part of the question was answered successfully by most candidates.

Question number: B10

This question on software requirements had three parts.

Part a) This part proved difficult for almost half of the candidates. Many cited the feasibility stage as the point of specification

Part b) This proved difficult for many. The distinction between functional and non-functional requirements was unclear and this led to many answers being unable to explain the importance of both being in the SRS.

Part c) Those candidates who found part b difficult also found this part difficult as it required candidates to appreciate the distinction between functional and non-functional requirements. Subsequently candidates who did well in part b did well in this part and were well able to describe three (often more) non-functional requirements that affect quality.

Question number: B11

This question on interface design had two parts.

Part a) Most candidates gave good answers to this part. A few did not relate the question to the design aspects having the screen to adapt to the various display technologies. Answers tended to concentrate on appearance and layout from a marketing perspective and ignored the design considerations.

Part b) In general almost all candidates provided good explanations of appropriate design adaptions for a range of user impairments. A few candidates had a single adaption (typically, text to speech) to accommodate all impairment classes.

Question number: B12

This question on software terminology had three sub sections.

i) Many candidates found difficulty in explaining the difference between elementary types and derived types. Many did not correctly define an elementary type such as integers, characters etc'.

ii) Descriptions of the linked list and array structures posed some difficulty for many candidates. A good many answers correctly identified differences but not similarity. Many answers gave a good account of linked list structures, but surprisingly did not articulate a good account of the array structure.

iii) Cloud v Local storage. The similarities and differences were well described by most candidates.