

**BCS Higher Education Qualification**

**Certificate in IT**

**October 2023**

**EXAMINERS' REPORT**

**Software Development**

**Questions Report:**

<b>A1</b>	<p>This question proved a popular choice amongst candidates however many failed to score high marks.</p> <p>Part a) asked for an algorithm to describe the problem to be solved. Many candidates provided a reasonable algorithm, but for others there was an element of confusion in expressing the problem as an abstract representation. Many candidates directly used a coded solution or simply provided a narrative which often did not fully express the problem. Quite a number of candidates expressed repetition by repeatedly writing out each loop as a series of steps. This might work OK for 2 or 3 numbers but wouldn't work for say a 100 numbers in the set.</p> <p>A small number of candidates introduced a binary/chop search into their algorithm and code to facilitate the search process. The question did not require this as part of the answer.</p> <p>Part b) Translation of the algorithm into code became slightly problematic for many as they had not produced an adequate algorithm in the first instance. However, the program code was generally well done by a majority of candidates who had correctly answered part a).</p> <p>Part c) Most gave reasonable answers to explaining why pseudocode is often used in systems development.</p> <p>Part d) Many candidates found difficulty in providing a suitable alternative to pseudocode. However, a majority did (correctly) suggest the flowchart as an alternative. Very few suggested an activity diagram as a possible alternative, although this would have been a correct choice also. The example flowcharts in many cases lacked the formal symbols that should be part of a standard flowchart. Some representations were simple box shapes with feedback lines drawn at decision points. Marks were often lost with this type of example as the logic was not correctly defined.</p> <p>An initial overview describing the overall purpose of a flow chart was often missing or poorly expressed. Quite often flow chart symbols were provided but lacked a convincing example. Overall though it was an opportunity to gain fairly easy marks and generally full marks were attained if the flow chart example reflected the game scenario and the algorithm in part a).</p>
<b>A2</b>	<p>This question was not a particularly popular choice for candidates. This question referred to a maths game designed for children. It was apparent that many</p>

	<p>candidates did not fully follow the question preamble and game rules and subsequently found some difficulty in interpreting the four-part questions that followed.</p> <p>Part a) for this part candidates should show code that plays the game. Many candidates chose to skip this part of the question or made cursory attempts to form code that did not fully reflect the game rules. For those candidates who did correctly interpret the game rules, the code produced was well done and succinct. Only minor errors appeared in general. A couple of marks were lost due to minor errors in formulating the IF ELSE IF statements and the need to state or declare integers for the input variable and integer division for the calculation.</p> <p>Part b) This part required the comparison of IF and SWITCH statements. Some candidates ignored the game question context of using if and switch and provided simple abstract examples. Many answers were able to indicate the inclusion of ELSE in the IF statement, although in many answers where this was noted the example did not relate to the game. The essential recognition that SWITCH is preferred in cases where there are long lists to be compared was noted in a minority of answers. There was an opportunity to relate the SWITCH statement to part a) code that avoided the need to conceive of something new as an example. It was noticeable a number of candidates were unfamiliar with the SWITCH statement maybe they have either never used it in their programming or less likely there is a programming language they use/ are familiar does not support this feature.</p> <p>Part c) This part required identifying FOUR types of loops using examples. Most candidates showed the FOR loop with suitable counters and also WHILE and DO WHILE loops. For many candidates the fourth choice was absent and very few considered REPEAT UNTIL. As in previous parts the misinterpretation of the question meant only a few examples related to elements of the game-based question. Subsequently marks were lost for not providing examples. It was interesting to note candidates choices apart from the usual FOR NEXT/ WHILE examples. For example DO WHILE was popular REPEAT UNTIL and an old one FOR EACH (set). A number of candidates correctly suggested the use of recursion to express looping or iteration so long as they gave equivalent code to say the FOR loop they had already presented.</p> <p>Part d) This part required an explanation of the role that modules play in structured programs. Most candidates answered this part well and provided a good range of built-in modules supported by various programming languages.</p>
<b>A3</b>	
	<p>This question was the least popular choice for candidates with many struggling to score high marks. The question was based on an algorithm for producing the Fibonacci sequence.</p> <p>Part a) This part required a trace of the algorithm execution with an initial value being supplied. Many candidates omitted this part of the question. For those who attempted, the results tended to vary quite widely. Many answers did not correctly produce a step-by-step trace or incorrectly calculated the values. A minority of candidates did present a full and correct trace of the execution and provided a well-structured trace table.</p>

	<p>Many candidates could not present a trace table in a logical and accurate format. Some attempts at actually producing a trace table that followed the steps was expected but in the main many outcomes of the trace were incorrect even after the first or second iteration which meant successive iterations of the trace were also incorrect. A number of candidates only went as far as tracing one iteration. Only a very small number produced a clear and concise trace table with correct final values of each variable.</p> <p>Part b) This part required examples of debugging tools to highlight problems in code. Many answers showed a good appreciation of the debugging tool as something to run the code under control conditions. Most answers recognised step-through techniques and the use of breakpoints. Very few answers extended into the examination of CPU registers and examination of memory allocation. Most candidates did not suggest a debugging tool per se, but simply mentioned that an IDE usually comes bundled with them.</p> <p>Some candidates spent too much time on superficial characteristics to do with standard features of interpreters and compilers and assistance to highlight keywords for example, rather than actual debugging tools.</p> <p>Part c) This part required the use of recursion to code the Fibonacci sequence. Many ignored this part of the question, which was a third of available marks. In a minority of answers a suitable recursive function was presented that showed both the function and call. In some instances, candidates produced the function but did not fully provide the calling code.</p>
<b>A4</b>	
	<p>A fairly popular question but generally a relatively poor performance in terms of average mark.</p> <p>This question referred to a snippet of computational code.</p> <p>Part a) This part was to explain the purpose of the supplied code. The question further required suggestions on how the code layout could be improved. A majority of candidates failed to acknowledge that the code was a computation to find prime numbers. The code layout improvements were generally well answered by most candidates with suggested standard changes to have meaningful variable names and proper indentation with the use of comments to make code purpose clearer. Many answers only included one or at most two such suggestions.</p> <p>The explanation of what the code does varied between a single sentence to a longwinded explanation of each line of code. It was more important to explain the overall function and purpose of the code rather than go through each line of code.</p> <p>Part b) This part required the noting of identifiers, conditional expressions and repetitive loops in the code snippet provided. A good number of answers did not refer to the code snippet to identify the examples. For those candidates who did relate to the supplied code, many were able to provide only partial examples or did not fully list the occurrence of all.</p> <p>Some candidates produced code snippets as examples that actually mixed up BOTH iteration statements with conditional statements. Therefore lost marks as a result.</p>

	<p>Part c) required an explanation of necessary changes to relace FOR with WHILE. The majority of candidates found some difficulty. Many answers correctly indicated the introduction of a control variable but very few showed the updating required and did not provide an explanation, but simply wrote the code.  candidates read the question as simply providing the necessary code to replace the FOR loop without the need to give any explanation.</p> <p>Part d) required a function to check if a prime number has been provided by a user. Many answers showed an appreciation of the key points in integer division within a loop and seeking a remainder. Very few answers went further and provided optimal loop conditions and complete function with a suitable call.  Many candidates attempts simply copied some or all of the code sample supplied in the question, not appreciating the need to translate the code into a function (with a single parameter) along with the function call returning either TRUE or FALSE.</p>
<b>B5</b>	
	<p>Very few candidates attempted this question. All parts of this question posed problems for the very few numbers of candidates who attempted it.</p> <p>Candidates were asked to represent an algorithm using a step-by-step approach, but many appeared unable to translate the equation into an algorithm. There were very few complete answers to this question.</p> <p>Part a) Many answers demonstrated the first three terms of the supplied sequence with a supplied input value. The computation involved calculation and arithmetic errors were evident (the use of calculators being prohibited). Many answers were incomplete.</p> <p>Part b) Only a very few attempts were made at this part of the question where the answer provided in part a was translated into code. A small minority of candidates showed an algorithm or code that was correct together with appropriate looping structures and use of variables.</p>
<b>B6</b>	
	<p>This was a popular question with most candidates doing a good job of answering all three sections. The final section asked candidates to “copy out a statement” from the code provided to illustrate features of a programming language – many provided examples of their devising rather than copying them from the code provided.</p> <p>Part a) This part required the snippet to be rewritten to make it more readable. Most answers gained full marks with a few making minor errors.</p> <p>Part b) This required identifying data types used in the code. Most candidates provided correct answers.</p> <p>Part c) This part required identification of a range of operators, conditions functions and assignments. Many candidates lost marks by not fully reading this part of the question correctly and identify a statement from the snippet but gave generic explanations. The majority of candidates did correctly answer the question but in some cases did not attempt to identify all cases.</p>

<b>B7</b>	
	<p>This was a popular question. Most candidates made a good job of the first part, but many gave only superficial answers to the second part. The question required examples of OO code or pseudocode to explain all answers. Many candidates did not fully take on board the requirement stated in the question preamble and subsequently for all parts of the questions that followed lost marks.</p> <p>Part a) This required an explanation of basic OO concepts. A large minority of candidates showed a good enough appreciation of Objects, Classes, Methods and Properties and gave reasonable definitions, but most candidates failed to provide examples and many only provided a limited number of definitions. In particular there was some hesitation in answers that provided the relationship between Class and Object. In many answers the (incorrect) suggestion was the Objects are the defining entity and Classes are subordinate. Most attempts at this part achieved less than half of the available marks.</p> <p>Part b) This required a discussion of inheritance. Most gave an adequate definition of inheritance, but many failed to provide a suitable example. Very few answers indicated the particular benefit of inheritance in enabling reuse.</p>
<b>B8</b>	
	<p>This question was relatively unpopular as a choice amongst candidates.</p> <p>Very few candidates attempted the third part, which required an algorithm, the answers to the first two parts, which required diagrams to be presented, were better but some were unclear lacking any supporting explanation.</p> <p>Part a) asked for a diagram to show a linked list. Very few candidates provided a correct diagram, and many answers were only able provide a narrative. Very few candidates gained marks for this part.</p> <p>Part b) This part required a diagram and an example. Answers tended to be in the narrative many diagrams were incorrect in showing an insertion into the linked list. In only a small number of cases did candidates correctly indicate insertion after the last node. A very small minority of candidates gave fully correct diagrams and example.</p> <p>Part c) This required pseudocode for insertion. Given the general uncertainty concerning the linked list structure only a very few number candidates were able to show suitable code.</p>
<b>B9</b>	
	<p>This was a popular question amongst candidates.</p> <p>The open-ended nature of the question led to a wide variety of answers but most of these were well thought through. There were some misunderstandings – for example, one or two candidates discussed issues related to software, rather than interface, design, and some discussed user interface testing rather than design principles.</p> <p>This question preamble concerning user centred design required four design principles that would achieve a user-friendly interface that encourages exploration without fear of bad consequences. Many candidates did not fully read the question</p>

	<p>requirements and subsequently did not specifically address the particular issues the question posed.</p> <p>A majority of candidates did provide adequate answers to the question and achieved good marks. A significant number of candidates did not address the question but provided UX design principles and related the answers to the benefits to a business in having attractive interfaces. In a number of answers candidates chose to frame the discussion in terms of good software design principles or the marketing opportunities of a particular interface layout. These answers tended to gain few marks.</p>
<b>B10</b>	
	<p>This question was concerned with testing techniques. The question was a popular choice amongst candidates with a majority of candidates achieving a pass mark.</p> <p>Most candidates provided a good answer to Part A although some seemed confused about the difference between functional and unit testing. The second part was less well addressed. Only a few candidates were able to provide a complete trace table without mistakes.</p>
	<p>Part a) This question on Whitebox/black box testing caused confusion in the definition for many candidates who incorrectly identified Whitebox testing with functional testing. Approximately half of all answers confused the purpose of unit testing v functional testing.</p> <p>Part b) this question required the production of a trace table working through the steps of supplied question code. Most of the candidates provided a trace table with some minor errors and omissions. Very few candidates completed all entries of the trace table often confusing the next state of the tables with the current state. Very few candidates achieved full marks by inclusion of the null initial state and the null end state, thus losing a mark or so. Most answers indicated correctly that the actual output from the table did not match the expected, but many could not provide a convincing reason as to why that was the case.</p>
<b>B11</b>	
	<p>This question was concerned with the appropriate documentation for stakeholder groups in a project that has been released.</p> <p>This was a popular question with most candidates providing lengthy answers, many of which were good, but a few were unclear about the differing needs of each stakeholder group. Some answers were repetitious with candidates identifying the same documentation for more than one stakeholder.</p> <p>For many candidates, they showed little grasp of the relevant documentation particularly for IT administrators/directors of IT. Suggesting (incorrectly) in many cases, that technical documentation is required with detail on the implementation and testing. Many candidates did not pay sufficient attention to the details of the question which specifically stated that the project was completed and subsequently cited the type of documentation required from the stakeholders during a project rather than the documentation they would be supplied with.</p>

<b>B12</b>	
	<p>This question concerned aspects of quality.</p> <p>Many candidates made a good job of Part A in this question but gave more perfunctory answers to Part B. Candidates were asked to discuss two of three quality assurance procedures, but many provided a discussion of all three.</p> <p>Part a) This dealt with meaning of quality in a development context. Most candidates gave reasonable accounts citing zero defects, fitness of purpose, satisfying requirements and so on. Many answers did however did not demonstrate more than two or three examples where the possibility of six gave the opportunity for full marks.</p> <p>Part b) This part required the explanation of two of three quality assurance procedures. Many candidates did not articulate the correct descriptions for the procedures or gave fairly terse definitions. Most answers to stress testing were able to correctly elaborate on the procedure. For Peer review a large proportion of answers showed a deal of confusion in defining the status of a peer in the development context and incorrectly identified peers as experts in the domain. Few identified customers or clients as peer stakeholders. A significant number of candidates either avoided this part of the question completely or gave minimal definitions which gained few if any marks. This part of the question accounted for half of the total marks available for the question.</p>