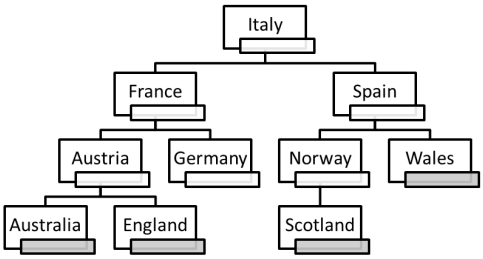
Question			Answer/Indicative content	Marks	Guidance
1	a		 1 mark for each of: - Scotland in correct place - Wales in correct place - Australia and England both in correct place	3 AO2.2 (3)	<u>Examiner's Comments</u> The majority of candidates had little difficulty with this question, but a surprising number of errors were made because candidates could not sort alphabetically past the first letter of the data values given to be inserted into the tree.
	b		1 mark per bullet to max • Italy • France, Spain • Austria, Germany, Norway	3 AO1.1 (1) AO2.1 (1) AO2.2 (1)	<u>Examiner's Comments</u> Most candidates found this to be straight-forward. Where there was confusion, candidates often performed a depth-first traversal, instead of a breadth-first traversal.

	c	i	1 mark per bullet to max 5
			
			`function searchForData(currentNode:byVal, searchValue:byVal)`
			`  thisNode = getData(`**`currentNode`**`)`
			`  if thisNode == `**`searchValue`**` then`
			`  return `**`true`**
			`  elseif thisNode < searchValue then`
			`  if currentNode.left () != null then`
			`  return (searchForData(currentNode.left (), searchValue))`
			`  else`
			`  return `**`false`**
			`  endif`
			`  else`
			`  if `**`currentNode.right()`**` != null then`
			`  return (searchForData(currentNode.right (), searchValue))`
			`  else`
			`  return false`
			`  endif`
			`  endif`
			`endfunction`
				5
				AO2.2
				(2)
				AO3.2
				(3)
					The line **`elseif thisNode < searchValue then`** should have read **`elseif thisNode > searchValue then`**
					
					If candidates attempt to correct the code and their answers are consistent with, and work with their amendment, such answers should be credited.
					
					<u>Examiner's Comments</u>
					
					Some candidates found the first two marks more difficult to access than the last three, because they did not fully understand that the parameters to the function were to be used.

Question			Answer/Indicative content	Marks	Guidance
		ii	• It's a binary tree • It's ordered / sorted	2 AO2.2 (2)	Examiner's Comments Many candidates recognised that a binary search tree was required, but some lost marks when they specified that 2 children were always required for each parent node, when it is a maximum of two children that are allowed, so that there can be 0, 1 or 2 children.
			Total	13	

Question			Answer/Indicative content	Marks	Guidance
2	a	i	Recognition • Identify there is a problem to be solved // what the problem is Decomposition • Splitting down a problem into sub-problems	2 AO1.1 (2)	<u>Examiner's Comments</u> Candidates found it easier to define decomposition than recognition. There were many circular definitions of the form problem recognition means recognising the problem that were not creditworthy.
		ii	e.g. • Divide and conquer • Abstraction	1 AO1.1 (1)	Accept other credible answers e.g.: Critical thinking, Modelling, Heuristics, Concurrency, Visualisation, Backtracking <u>Examiner's Comments</u> The most common answer that candidates gave was abstraction, but a wide range of valid responses that included the named computational methods in the specification were seen.
	b	i	• Turning large quantities of data into useful information / Finding patterns within large quantities of information	1 AO1.1 (1)	Must refer to large quantities of data <u>Examiner's Comments</u> Whilst many candidates had some familiarity with the term data mining there were a number of very vague answers. Candidates need to know that very large volumes of data are collected with the intention of finding patterns and trends.
		ii	1 mark per identifying data, 1 for use e.g. • Identify customer trends • To identify items to sell/offers to send customers • Identify which stores are making the most profit • To identify what the other stores are doing well • Which items are not selling well • To replace them with other items	4 AO2.2 (4)	Accept any valid responses <u>Examiner's Comments</u> Many candidates could give a good response that explained how data trends identified could be applied in a sales scenario, but some candidates only gave answers related to stock control which was not relevant.
	c	i	Simulate/test the behaviour of the system before it is used	1 AO1.1 (1)	

Question			Answer/Indicative content	Marks	Guidance
		ii	e.g. • Testing it with a large number of simultaneous orders (stress testing) • Testing it with a large number of customers/items/orders	1 AO2.2 (1)	Examiner's Comments Many candidates had clearly not come across the term 'performance modelling' which is in the specification, and they struggled to relate it to the context given. However, some very good responses were given, that included the use of Big O notation analysis and mathematical modelling to determine algorithmic performance.
	d		1 mark per bullet to max 2 e.g. • the components can be used in a future program… • they do not need to be rewritten / saves time • they have already been tested… • …it will save time	2 AO1.1 (1) AO2.1 (1)	Examiner's Comments Whilst many candidates could identify that reusable components would save time in future projects, fewer could identify specific reasons why this would be the case.
			Total	12	

Question			Answer/Indicative content	Marks	Guidance
3	a	i	Any one from: - A graph has cycles - A graph can be directed/undirected - A tree has a hierarchy (e.g. Parent/Child)	1 AO1.2 (1)	Allow any appropriate description e.g. graph can be weighted, tree has a root Examiner's Comments Many candidates scored well, but there was evidence of a lack of learning of technical vocabulary e.g. general descriptions of loops rather than cycles in graphs. Candidates are expected to be able to use the correct technical vocabulary such as weighted / directed in reference to graph structures.
		ii	1 mark per bullet to max 2 • The puzzle is not shown in the diagram • The graph shows different sequences of sub problems in the puzzle that can be solved to get to the final solution • The puzzle does not have all states visible at once	2 AO1.2 (1) AO2.1 (1)	Answers must be in context of the puzzle Examiner's Comments Most candidates scored poorly on this question because they did not contextualise their answers, and did not explain how the graph could be an abstraction of the puzzle in the scenario.
		iii	1 mark per bullet to max 2 e.g. • Visualisations benefit humans rather than computers • Visualisations present the information in a simpler form to understand • Visualisations can best explain complex situations	2 AO1.2 (1) AO2.1 (1)	Examiner's Comments Few candidates achieved both marks. Common answers that were creditworthy recognised the fact that humans find visualisations of problems easier to understand.

Question			Answer/Indicative content	Marks	Guidance
	b		1 mark per bullet • Mark A as the initial node and then visit B (5) • Node E (8) is then visited (chosen from C (13), D (14), E (8)) • Node I (12) is then visited after E • Node J (14) is then visited after I • Visiting G (15) from C – overriding the previous value of 18 • solution A-B-E-I-J path length 14	7 AO1.2 (3) AO2.1 (2) AO2.2 (2)	Examiner's Comments Whilst many candidates were familiar with the calculations involved in Dijkstra's algorithm, most confined their calculations to the shortest path ABEIJ and length 14. It was far less common for candidates to recognise that Dijkstra's algorithm does not stop until all nodes have been visited and that some nodes are over-ridden with lower values. Some candidates simply stated the shortest route without any explanation or calculation and this was not creditworthy. Candidates should be encouraged to produce clear answers in tabular format.

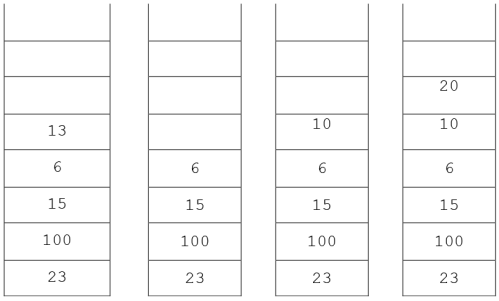
Question	Answer/Indicative content	Marks	Guidance
c	Mark Band 3 – High level (7-9 marks) The candidate demonstrates a thorough knowledge and understanding of Dijkstra's and A*; the material is generally accurate and detailed. The candidate is able to apply their knowledge and understanding directly and consistently to the context provided. Evidence/examples will be explicitly relevant to the explanation. *There is a well-developed line of reasoning which is clear and logically structured. The information presented is relevant and substantiated.* Mark Band 2 – Mid level (4-6 marks) The candidate demonstrates reasonable knowledge and understanding of Dijkstra's and A*; the material is generally accurate but at times underdeveloped. The candidate is able to apply their knowledge and understanding directly to the context provided although one or two opportunities are missed. Evidence/examples are for the most part implicitly relevant to the explanation. The candidate provides a reasonable discussion, the majority of which is focused. Evaluative comments are, for the most part appropriate, although one or two opportunities for development are missed. *There is a line of reasoning presented with some structure. The information presented is in the most part relevant and supported by some evidence.* Mark Band 1 – Low Level (1-3 marks) The candidate demonstrates a basic knowledge of Dijkstra's and A* with limited understanding shown; the material is basic and contains some inaccuracies. The candidates makes a limited attempt to apply acquired knowledge and understanding to the context provided. The candidate provides a limited discussion which is narrow in focus. Judgements if made are weak and unsubstantiated. *The information is basic and comunicated in an unstructured way. The information is supported by limited evidence and the relationship to the evidence may not be*	9 AO1.1 (2) AO1.2 (2) AO2.1 (2) AO3.3 (3)	AO1: Knowledge and Understanding Indicative content • Heuristic helps produce a solution in a faster time • A* uses estimated distance from final node • Dijkstra uses a weight/distance • A* chooses which path to take next based on lowest current distance travelled AO2: Application • Description of how A* will differ from Dijkstra, e.g. taking the shorter route A-B-E-I before exploring nodes from D and E • Description of the different number of comparisons that would be needed in this problem • A* doesn't need to find all possible solutions (saves time) AO3: Evaluation Candidates will need to evaluate the benefits and drawbacks of each algorithm • Small-scale problem • Quick to find a solution using either method • Difference in programming complexity is minimal • Don't know if this problem needs to scale • Most efficient route needed <u>Examiner's Comments</u> Candidates were well prepared and had obviously used both Dijkstra's algorithm and the A* algorithm, and could explain differences between them and give examples of possible heuristics. Very few candidates could evaluate and discuss how the scaling of the algorithms was relevant to access the top level mark band.

Question			Answer/Indicative content	Marks	Guidance
			clear. 0 marks No attempt to answer the question or response is not worthy of credit.		
	d		1 mark per bullet to max 4 e.g. • Underlines syntax errors dynamically • Can be corrected before running // saves times • Watch window • View how variables change during running of the program • Break points • Stop the program at set points to check the values of variables • Error message list • Tells you where errors are and suggests corrections • Step-mode • Executes program one statement at a time to watch variable values and program pathways • Traces • Print-outs of variable values for each statement execution within a program • Crash-dump/post-mortem routine • Shows the state of variables where an error occurs • Stack contents • Shows sequencing through procedures/modules • Cross-referencers • Identifies where variables/constants are used in a program to avoid duplications	6 AO1.1 (3) AO1.2 (3)	Examiner's Comments Many candidates did not read the question clearly and consequently did not specify IDE features that were specifically relevant to debugging the program and thus discussed editing features. Good answers included break points, variable watches and stepping.
			Total	27	

Question			Answer/Indicative content	Marks	Guidance
4	a		1 mark per bullet for working to max 6 • generate(7) return 7 + (generate(8) DIV 2) • generate(8) return 8 + (generate(9) DIV 2) • generate(9) return 9 + (generate(10) DIV 2) • generate(10) return 10 + (generate(11) DIV 2) • generate(11) return 10 • Rewinding: return 10 + (10 DIV 2) = 10 + 5 = 15 • return 9 + (15 DIV 2) = 9 + 7 = 16 return 8 + (16 DIV 2) = 8 + 8 = 16 • return 7 + (16 DIV 2) = 7 + 8 = 15	6 AO1.2 (1) AO2.2 (5)	Examiner's Comments A significant number of candidates struggled to clearly present a program trace for a recursive algorithm. Those who used diagrammatic or clearly indented structures showing the recursive levels and the return values faired best. Candidates need to be encouraged to work on producing a logical layout for a recursive function trace.
	b		• If the value is sent by value, num1 will not be overridden / it is a copy of the parameter that is used (1) and this will produce the correct output (1) • if the parameter had been passed by reference it would not produce the correct result (1) as num1 would be overridden / because it is a pointer to the address of the variable (1)	2 AO2.1 (1) AO2.2 (1)	Examiner's Comments Many candidates either defined passing by value or passing by reference and did not answer the question. Few could demonstrate a deeper understanding of the implications of the method of parameter passing chosen within the context of a recursive function.

Question	Answer/Indicative content	Marks	Guidance
c	Mark Band 3 – High level (7-9 marks) The candidate demonstrates a thorough knowledge and understanding of parameters and global variables; the material is generally accurate and detailed. The candidate is able to apply their knowledge and understanding directly and consistently to the context provided. Evidence/examples will be explicitly relevant to the explanation. *There is a well-developed line of reasoning which is clear and logically structured. The information presented is relevant and substantiated.* Mark Band 2 – Mid level (4-6 marks) The candidate demonstrates reasonable knowledge and understanding of parameters and global variables; the material is generally accurate but at times underdeveloped. The candidate is able to apply their knowledge and understanding directly to the context provided although one or two opportunities are missed. Evidence/examples are for the most part implicitly relevant to the explanation. The candidate provides a reasonable discussion, the majority of which is focused. Evaluative comments are, for the most part appropriate, although one or two opportunities for development are missed. *There is a line of reasoning presented with some structure. The information presented is in the most part relevant and supported by some evidence.* Mark Band 1 – Low Level (1-3 marks) The candidate demonstrates a basic knowledge of parameters and global variables with limited understanding shown; the material is basic and contains some inaccuracies. The candidates makes a limited attempt to apply acquired knowledge and understanding to the context provided. The candidate provides a limited discussion which is narrow in focus. Judgements if made are weak and unsubstantiated. *The information is basic and comunicated*	9 AO1.1 (2) AO1.2 (2) AO2.1 (2) AO3.3 (3)	AO1: Knowledge and Understanding Indicative content • Parameter allows a value to be sent to a sub-program • Global variables can be accessed throughout the scope of the program • Local variables can only be accessed within the scope of the sub-program it's defined within ¡V a parameter becomes a local variable in the function AO2: Application • If global, equivalent of by reference -value would be over-ridden • Global variable takes more memory than a local variable/parameter • In recursion, each call produces a new local variable for num1 AO3: Evaluation Candidates will need to evaluate the benefits and drawbacks of each algorithm • Global would require altering the algorithm as the value would be over-ridden on each call • Global would mean that memory space is kept throughout the running of the program, not just the sub-program • Parameter enables memory to be reallocated • Many more memory spaces needed for parameter in recursion, 1 for each call <u>Examiner's Comments</u> Most candidates produced responses limited to the scope of global variables being accessible throughout the program or a discussion of the different methods of parameter passing available. Few made any references to either recursive functions or to the implications to memory usage of using parameters instead of global variables.

Question			Answer/Indicative content	Marks	Guidance
			in an unstructured way. The information is supported by limited evidence and the relationship to the evidence may not be clear. 0 marks No attempt to answer the question or response is not worthy of credit.		
	d		1 mark per bullet • Each recursive call stores the current state on the stack // creates new variables • Iteration reuses the same variables	2 AO1.2 (1) AO2.1 (1)	Examiner's Comments Candidates need to demonstrate technical knowledge. Those who answered well produced good descriptions of how a stack is used when a function is called recursively. Few candidates then went on to explain how iteration overwrites the same variables with new values.
			Total	19	

Question			Answer/Indicative content	Marks	Guidance
5	a			4 AO1.2 (2) AO2.2 (2)	Examiner's Comments The vast majority of candidates had no trouble executing a sequence of push/pop instructions successfully.
	b	i	1 mark per bullet, max 2 for insert, max 2 for remove push • Check if the stack is full (pointer = array.length/array.length+1) • If it is not ¡V insert the item • If it is ¡V return/error that the stack is full pop • Check if the stack is empty (pointer = 0/1) • If it is ¡V return/error that the stack is empty • If it is not ¡V return the item	4 AO1.2 (2) AO2.2 (2)	Examiner's Comments A significant number of candidates did not describe a conditional decision clearly and lost marks when merely describing push/pop operations. Where the first part of the question was answered well some candidates then failed to see the second part of the question and did not describe an impact of the condition. Candidates need to be reminded to read and analyse the wording of the whole question to access all marking points.
		ii	1 mark per line, 1 for change • line 02 • Include an OR with variations (e.g. `userAnswer = "PUSH" OR userAnswer = "Push"` etc.)/Convert input to uppercase/lowercase and just compare to equivalent	2 AO2.2 (2)	Examiner's Comments Most candidates answered well, but few gave answers that demonstrated an ability to combine separate logical statements with the OR operator. A significant number of candidates used a .lower() method being mostly familiar with Python syntax and methods.

<table>
<tr><th colspan="3">Question</th><th>Answer/Indicative content</th><th>Marks</th><th>Guidance</th></tr>
<tr><td></td><td>c</td><td></td><td>1 mark per bullet to max 3

• Array size defined
• A stack pointer is used to point to the top of the stack
• When an item is pushed the stack pointer is incremented
• When an item is popped the stack pointer is decremented</td><td>3
AO1.2
(1)
AO2.1
(1)
AO2.2
(1)</td><td><u>Examiner's Comments</u>

Those candidates with experience of languages other than Python appreciated that a 1D array is a static structure that needs to be declared with a given size, and that a stack pointer variable would be required. It is of concern that significant numbers of candidates have only had experience of lists and their associated methods in Python. A number of candidates also confused their descriptions with those for a queue rather than a stack.</td></tr>
<tr><td></td><td>d</td><td>i</td><td>1 mark per row (after first row)

<table><tr><td>100</td><td>22</td><td>5</td><td>36</td><td>999</td><td>12</td><td></td></tr><tr><td>22</td><td>100</td><td>5</td><td>36</td><td>999</td><td>12</td><td>1 mark</td></tr><tr><td>5</td><td>22</td><td>100</td><td>36</td><td>999</td><td>12</td><td>1 mark</td></tr><tr><td>5</td><td>22</td><td>100</td><td>36</td><td>999</td><td>12</td><td>1 mark</td></tr><tr><td>5</td><td>12</td><td>22</td><td>36</td><td>100</td><td>999</td><td>1 mark</td></tr></table></td><td>5
AO2.2
(5)</td><td><u>Examiner's Comments</u>

Candidates should be encouraged to demonstrate sorting algorithms through the use of clear diagrams that show the steps/passes in the sorting algorithm. Where candidates used verbose text, it often made it far harder to follow whether or not the correct sorting algorithm had been applied. Most candidates did implement an insertion sort, but some did describe bubble or merge sorts instead.</td></tr>
<tr><td></td><td></td><td>ii</td><td>1 mark per bullet to max 7

• Repeat
• Calculating an array midpoint…
• …by adding the array lower bound to the array upper bound, dividing by 2 and rounding
• Compare array midpoint with value to search for…
• …if equal set found flag to true
• …if array midpoint < value to search for, change lowerbound to equal midpoint + 1
• …if array midpoint > value to search for, change upperbound to equal midpoint - 1
• Until lowerbound is greater than or equal to upperbound
• Return/output found flag</td><td>7
AO1.1
(2)
AO1.2
(3)
AO2.1
(1)
AO2.2
(1)</td><td><u>Examiner's Comments</u>

The paper title is 'Algorithms and programming'. Binary search is a standard algorithm that should be fully understood by candidates, and candidates should be able to program it. Many candidates produced very vague descriptions that were far too general to credit at this level. Candidates needed to be able to discuss how the upper and lower bound pointers are used in this algorithm (or equivalent for recursive solutions).</td></tr>
</table>

Question			Answer/Indicative content	Marks	Guidance

		iii	1 mark per bullet
			
			• Setting variable to start at 0
			• Suitable while structure (endwhile or clear indentation)
			• looping 50 times
			• Incrementing the variable within the loop
			
			e.g. 1
			`function searchItem(dataItem)`
			`  count = 0`
			`  while count < 50`
			`  if dataArray(count) == dataItem then`
			`  return(count)`
			`  endif`
			`  count = count + 1`
			`  endwhile`
			`  return(-1)`
			`endfunction`
			
			e.g. 2
			`function searchItem(dataItem)`
			`  count = 0`
			`  while count < 50 and dataArray[count]!=dataItem`
			`  count = count + 1`
			`  endwhile`
			`  if count==50`
			`  count=-1`
			`  endif`
			`  return(count)`
			`endfunction`
				4
				AO1.2 (1)
				AO3.1 (1)
				AO3.2 (2)
					Examiner's Comments
					
					It was a little disappointing to see a number of candidates using variations on for loops rather than rewriting the code using a while loop as required. A significant number of candidates still struggled to demonstrate a coherent logical response that would work for something that is relatively simplistic, thus showing a lack of proficiency in coding practice.

			Total	29	

Question			Answer/Indicative content	Marks	Guidance
6	a	i	1 mark per bullet to max 3 • Record is a data structure… • …A class is a template for making data structures (objects) • Class also has methods (which describes functionality) • Both store data of different types • Which can be accessed by their names • But classes can make them accessible via methods • Both can have multiple ¡¥instances¡¦ • Class can include visibility of properties / private	3 AO1.2 (3)	<u>Examiner's Comments</u> The concept of a record structure in a programming language was poorly understood – with many candidates interpreting it as a database record structure. Those who did score well had a far better understanding of OOP.

		ii	1 mark per space
			
			`recordStructure` **`items`**
			`    itemName : String`
			`    cost : Currency`
			`    dateArrival : Date`
			`    transferred : Boolean`
			`endRecordStructure`
				5
				AO2.2
				(2)
				AO3.2
				(3)
					<u>Examiner's Comments</u>
					
					Most candidates answered well, but weaker candidates put examples for the Currency and Date fields rather than creating suitable variable names.

		iii	1 mark per bullet to max 3
			
			• Declaring box1 as an item
			• Using Box1. (or equivalent) for each variable
			• Setting each variable (matching 6aii) correctly
			
			e.g.
			`Box1 : Items`
			`Box1.itemName = "Box"`
			`Box1.cost = 22.58`
			`Box1.dateArrival = "1/5/2018"`
			`Box1.transfered = True`
				3
				AO2.2
				(2)
				AO3.2
				(1)
					Ensure variable names for cost and dateArrival are consistent with variable names given in a(ii)
					
					<u>Examiner's Comments</u>
					
					The structure type was not well understood by many candidates, who used "recordidentifier" as the record identifier, rather than appreciating that *box1* could be assigned the data type *items*. Some candidates incorrectly omitted the quotation marks around the string data or added quotation marks to the numerical data fields.

	b	i	1 mark per bullet to max 2 • A data structure • FIFO (first in first out)	2 AO1.1 (2)	<u>Examiner's Comments</u> Most candidates had learned the definition for *queue* and answered successfully.

Question			Answer/Indicative content	Marks	Guidance
		ii	1 mark per bullet to max 2 • Properties (are encapsulated) and can only be accessed through their methods • Enforce validation through the method // inappropriate data can be caught before entered • Cannot be changed/accessed accidentally	2 AO1.2 (2)	Examiner's Comments Many candidates termed their answers by restating terms in the question. Whilst the term encapsulation was often cited it was less often explained well – few candidates knew that getter() and setter() methods could be used to access the private attributes of a class. Some candidates stated that private attributes could not be changed which demonstrated a clear lack of understanding of the paradigm.

		iii	1 mark per bullet to max
			
			• Constructor method/`new`
			• Setting `head` and `tail` to 0 within constructor method
			
			e.g.
			`public procedure new()`
			`  head = 0`
			`  tail = 0`
			`  numItems = 0`
			`endprocedure`
				2
				AO2.2
				(1)
				AO3.2
				(1)
					Examiner's Comments
					
					Few candidates appeared to have had practical experience of programming in an Object Oriented Programming (OOP) languages. Those that did, answered well. It is advisable to ensure that candidates are prepared for section B of the paper by having implemented programs using the OOP methodology.

Question			Answer/Indicative content	Marks	Guidance

		iv	1 mark per bullet to max 6
			
			• Function declaration, taking item as a parameter
			• Checking if the queue is full…
			• …outputting/reporting error and returning `false`
			• Adding the item to the `tail` position
			• Correctly updating the `tail` pointer (either before or after addition)
			• Incrementing `numItems` and returning `true` if successful
			
			e.g.
			`public function enqueue(newItem : items) : boolean`
			`  if numItems = 10 then`
			`  print("Error: The queue is full")`
			`  return false`
			`  else`
			`  theItems[tail] = newItem`
			`  if tail = 9 then`
			`  tail = 0`
			`  else`
			`  tail += 1`
			`  endif`
			`  numItems += 1`
			`  return true`
			`  endif`
			`endprocedure`
				6
				AO2.2
				(3)
				AO3.1
				(1)
				AO3.2
				(2)
					Examiner's Comments
					
					Candidates struggled to apply all the information given in the stem of question 6. The stem defined the class *itemQueue* with the private attributes and methods it held. Candidates who did not use this information were unable to produce correct solutions. Only a few of the strong candidates realised that it was a circular queue that was implemented, and therefore checked the increment of the tail pointer.

		v	e.g. `myItems = (new) itemQueue()`	1 AO2.1 (1)	Allow follow through if they have parameters in 6(b)(iii) Examiner's Comments Lack of practical experience and lack of correct solutions for 6b(iii) often led to incorrect answers for this question.

Question			Answer/Indicative content	Marks	Guidance

		vi	1 mark per bullet to max 5
			
			• Procedure declaration for insertItems
			• Asking for input of data items for a new item …..
			• …using record structure correctly
			• Use of myItems.enqueue
			• Looping while the queue is not full
			
			e.g.
			`procedure insertItems()`
			
			`newItem : Items`
			`itemCount = myItems.getnumItems()`
			
			`while itemCount < 10`
			`newItem.itemName = input("Enter the item name")`
			`newItem.cost = input("Enter the item cost")`
			`newItem.dateArrival = input("Enter the date of arrival")`
			`newItem.transferred = input("Has it been transferred?")`
			`myItems.enqueue(newItem)`
			`itemCount = itemCount + 1`
			`endwhile`
			
			`myItems.setnumItems(itemCount)`
			
			`endprocedure`
				5
				AO2.2
				(2)
				AO3.1
				(1)
				AO3.2
				(2)
					Examiner's Comments
					
					Few candidates realised that you cannot use a single input statement to read in four data items, and only stronger candidates realised that they should use the enqueue method created in b(iv) to add the inputted items into the queue.

		vii	1 mark per bullet to max 2 • Store the items and queue to an external file (when the program closes) • Load the items and queue from the file when it starts	2 AO2.1 (1) AO2.2 (1)	Examiner's Comments Most candidates answered this well, realising that the data in the queue had to be written and retrieved from secondary storage.
	c		Mark Band 3 – High level (7-9 marks) The candidate demonstrates a thorough knowledge and understanding of caching and concurrent processing; the material is generally accurate and detailed. The candidate is able to apply their knowledge and understanding directly and consistently to the context provided. Evidence/examples will be explicitly	9 AO1.1 (2) AO1.2 (2) AO2.1 (2) AO3.3 (3)	AO1: Knowledge and Understanding Indicative content Caching • Previously used data is stored in a location… • that can be quickly accessed … • to speed up retrieval if needed in future

Question	Answer/Indicative content	Marks	Guidance
	relevant to the explanation. *There is a well-developed line of reasoning which is clear and logically structured. The information presented is relevant and substantiated.* **Mark Band 2 – Mid level** **(4-6 marks)** The candidate demonstrates reasonable knowledge and understanding of caching and concurrent processing; the material is generally accurate but at times underdeveloped. The candidate is able to apply their knowledge and understanding directly to the context provided although one or two opportunities are missed. Evidence/examples are for the most part implicitly relevant to the explanation. The candidate provides a reasonable discussion, the majority of which is focused. Evaluative comments are, for the most part appropriate, although one or two opportunities for development are missed. *There is a line of reasoning presented with some structure. The information presented is in the most part relevant and supported by some evidence.* **Mark Band 1 – Low Level** **(1-3 marks)** The candidate demonstrates a basic knowledge of caching and concurrent processing with limited understanding shown; the material is basic and contains some inaccuracies. The candidates makes a limited attempt to apply acquired knowledge and understanding to the context provided. The candidate provides a limited discussion which is narrow in focus. Judgements if made are weak and unsubstantiated. *The information is basic and comunicated in an unstructured way. The information is supported by limited evidence and the relationship to the evidence may not be clear.* **0 marks** No attempt to answer the question or response is not worthy of credit.		Concurrent Processing • several processes work simultaneously to solve a problem **AO2: Application** Caching • search for previously searched for data items in a faster secondary storage device/RAM • Speed up access for that item …Relies on same item being searched for multiple times …Kamran needs to decide how feasible this is based on the number of item Concurrent • Computer would have multiple processors… • Each searching part of the data structure at one time… • This would be limited by bottlenecks such as accessing the storage device • The n processors could potentially mean an increase of up to 1/n of time…realistically speed increase is likely to be less than that • Only useful if using linear search // binary search cannot be performed concurrently **AO3: Evaluation** Candidates will need to evaluate the benefits and drawbacks of caching and concurrent processing Allow any point of view (caching / concurrent / both) as long as argument is presented suitably. <u>Examiner's Comments</u> Most candidates seem to know a little bit about each of the techniques of caching and concurrency, but rarely went beyond basic points. Most candidates identified cache with cache memory rather than disk caching of recently retrieved records. Stronger candidates did access the higher mark bands by contextualising their answers, explaining how concurrent processes or parallel algorithms could be applied to searching large volumes of data.

Question			Answer/Indicative content	Marks	Guidance
			Total	40	