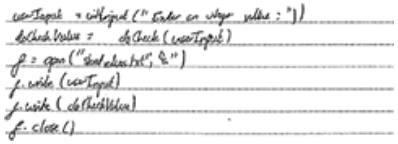
Mark scheme

Question			Answer/Indicative content	Marks	Guidance
1			• INP and STA two numbers correctly • Appropriate use of branching to exit the loop. • SUB ONE and STA NUM • LDA ANSWER and ADD NUMA • Output (correct) answer. • Set DAT for NUMA, NUMB, ONE, ANSWER after HLT **POSSIBLE SOLUTION:** INP STA NUMA INP STA NUMB LOOPSTART LDA NUMB BRZ LOOPEND SUB ONE STA NUMB LDA ANSWER ADD NUMA STA ANSWER BRA LOOPSTART	AO3.2 (6)	First and second number can be interchanged Accept suitable labels for DAT and Loop points For point 6 the 0s for ANSWER, NUMA, NUMB are optional but ONE must be initialised to 1. Allow FT for BP5 if logic is correct. Accept alternative mnemonics identified in the specification. **Examiner's Comments** Most candidates were given some of the available marks for this question. Most were able to take two inputs and perform the addition of two numbers, storing the result in the accumulator. Candidates that lost marks did not successfully loop through the instructions correctly, didn't output the correct answer and/or did not make use of DAT to subtract a value correctly.

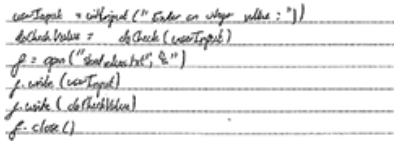
			LOOPEND LDA ANSWER OUT HLT ANSWER DAT 0 ONE DAT 1 NUMA DAT 0 NUMB DAT 0		
			Total	**6**	

2			1 mark for each statement
			
			`function findNode(toFind, headPointer, linkedList)`
			`  currentNode = headPointer`
			`  while(currentNode != NULL)`
			`  if linkedList[currentNode].data == toFind then`
			`  return currentNode`
			`  else`
			`  currentNode = linkedList[currentNode].pointer`
			`  endif`
			`  endwhile`
			`  return -1`
			`endfunction`
				5	Ignore case of identifiers in pseudocode
					
					Only penalise excessive spaces within identifier names if obvious.
					
					Examiner's Comments
					
					This question required exact answers only. Many candidates gained some marks, and there was a good distribution of marks. More successful programmers tended to get most of the marks available.

			Total	**5**	
3	a		• 9	1	**Examiner's Comments**

					This was a relatively simple algorithm question which candidates could answer using pseudocode or program code. Many had not read the first part of the Question 5 (a) where the doCheck function was already shown and tried writing the doCheck function instead of passing a parameter to it as stated in the question. Many candidates were unable to show an understanding of how to open a file and write to it. However there were very many excellent responses to all parts of this question. Exemplar 3  In this response the candidate has clearly shown input, calling the doCheck function and assigning the return value to a variable, File is opened and file name is enclosed in quotes and file is closed. The file name is then used to write to the file with the two values.

	b		• **input and store/use** a value from user
			• call **doCheck** function with value input from user **and save/use returned value**
			• open and close text file in write/append mode, if given
			• write value returned to text file
				5	Example code:
					`num = input("enter a number")`
					`value=doCheck(num)`
					`txtfile = openWrite("storedvalues.txt")`

			• Write value input to text file		`txtfile.writeLine(num)`
					`txtfile.writeLine(value)`
					`txtfile.close()`
					
					MP2 - **doCheck** is case sensitive
					
					MP3 - need speech marks around file name
					
					Examiner's Comments
					
					This was a relatively simple algorithm question which candidates could answer using pseudocode or program code. Many had not read the first part of the Question 5 (a) where the doCheck function was already shown and tried writing the doCheck function instead of passing a parameter to it as stated in the question. Many candidates were unable to show an understanding of how to open a file and write to it. However there were very many excellent responses to all parts of this question.
					
					Exemplar 3
					
					 
					
					In this response the candidate has clearly shown

					input, calling the doCheck function and assigning the return value to a variable, File is opened and file name is enclosed in quotes and file is closed. The file name is then used to write to the file with the two values.
			Total	**6**	
4		i	• 8, 7 • 15 • 15,6	3	One mark per stack diagram **Examiner's Comments** This question was generally well answered although some candidates struggled to understand the concept of a stack and how data is pushed on to it and popped from it.
		ii	• 12 • 7 • 15	3	**Examiner's Comments** This was generally well answered, with the biggest misunderstanding being the way the subtraction occurs.
		iii	• S causes the two values inputted to be popped and only one value to be pushed back / 4 and 2 are popped and 2 is pushed • A causes an attempt to pop **two values but only one present / not two values there** • Causing a **stack** underflow	3	**Examiner's Comments** Many candidates were able to gain at least one mark on this question for stating that the addition would only be able to pop one number. The candidates who gained full marks were able to state the type of error correctly and explain why

					there was only one value able to be popped after the subtraction.
			Total	**9**	
5		i	1 mark per bullet to max 1 • Simulate/model behaviour of the system (before it is) used under load • Because it would be too expensive/unsafe/time critical to test the real system	1	**Examiner's Comments** The concept of stress testing a system by simulating heavy loads seemed to be poorly understood by most candidates. Many candidates often confused this with the general testing of a system to find bugs.
		ii	1 mark per bullet to max 2 e.g. • Test with large and small values • e.g. largest number of deliveries • e.g. largest number of possible routes • Model how well the system scales with increasing use.	2	**Examiner's Comments** There was much confusion over testing a system in general to determine whether an individual route was effective in the graph given, and the actual performance of the system as a whole in terms of the overall time taken to actually perform the calculations required. The majority of candidates did not appreciate that performance modelling looks at the effect of escalating loads on a system.
			Total	**3**	
6		i	1 mark per bullet to max • Queue has head pointer and tail pointer • When an item is enqueued the tail pointer increments • When an item is dequeued the head pointer increments	3	Max 1 mark for Enqueue/Dequeue operations if description of effect on tail/head pointers not given **Examiner's Comments** Many candidates identified the need to have a head/tail pointer but struggled to gain more than 1 mark by expanding on how enqueue and dequeue

					operations would be implemented. Some candidates continued to talk about push/pop operations for a queue rather than enqueue/dequeue and often gave properties of a queue in general such as First In First Out rather than answering the question.
		ii	**Mark Band 3 – High level** **(7-9 marks)** The candidate demonstrates a thorough knowledge and understanding of object-oriented and procedural programming; 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 object-oriented and procedural programming; 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.	9 AO1.1 (2) AO1.2 (2) AO2.1 (2) AO3.3 (3)	**AO1: Knowledge and Understanding** **Indicative content** • OOP defines an object as an independent entity • OOP defines the attributes of the object and the methods that can be applied to it • attributes could be private to restrict accidental changes • Procedural the statements are executed in the order they are written **AO2: Application** • OOP allows for an object to be created from the queue • Many instances of this queue can then be declared in the main program. • Procedural will need each queue to be declared individually • Procedural will need to make use of subroutines where the queue will need to be sent and returned each time. **AO3: Evaluation**

			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 object-oriented and procedural programming 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.		• OOP you can create multiple instances of the queue as required by the program without having to re-write all of the declarations etc. In procedural you would have to write separate code for each new stack • OOP reduces amount of code needed therefore fewer errors are likely as code is written once and then used multiple times • OOP can reduce mistakes because the subroutines are self-contained in procedural it would need to make sure the correct values are passed and returned, or global variables may be required which uses excess memory. **<u>Examiner's Comments</u>** Many candidates were able to identify some elements of OOP and Procedural programming to achieve a Level 1 response or were able to describe features in detail for a Level 2 response. Far fewer were able to apply this to the specific context to achieve a Level 3 response. Those with good knowledge of OOP stood out in terms of giving clear evaluations of multiple queues generated from instance of the class, encapsulation to reduce side effects and possibilities for inheritance for different types of queues.

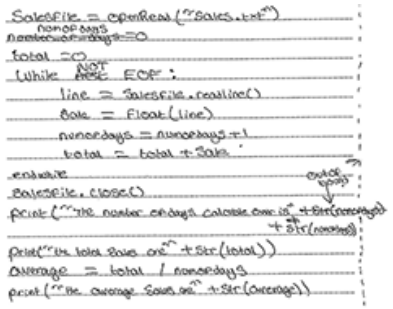
					Many did not describe the necessary creation of enqueue and dequeue subroutines in procedural programming for each separate queue or the need to pass queues to or returning queues from subroutines. **Misconception** There was a lot of confusion between inheritance and instantiation, e.g. “when creating several queues you can use inheritance, so all queues inherit attributes and methods”. Candidates need to be clear that each instance of a class is assigned the attributes of the class and has access to all associated methods.
			Total	**12**	
7	a		**Mark Band 3–High Level (7-9 marks)** The candidate demonstrates a thorough knowledge and understanding of both waterfall and the spiral model. 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. The candidate provides a thorough discussion which is well balanced. Evaluative comments are consistently relevant and well-considered.	9 (AO1.1), (2) (AO1.2), (2), (AO2.1) (2), (AO3.3). (3)	AO1 The spiral model has four quadrants (determine objectives, identify and manage risk, develop and test, plan next iteration). Client feedback then informs future development and prototypes which feedback into future revisions. Waterfall has a structured analysis/design/development/test flow. Progress to the next step is not made until the previous step is completed. AO2 The spiral model relies on frequent client

			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 waterfall and/or the spiral model; 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 sound 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 some aspects of either waterfall or the spiral model; the material is basic and contains some inaccuracies. The candidate 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. Judgments if made are weak and unsubstantiated. The		feedback. Spiral produces functional prototypes where features are added incrementally. Spiral model has more focus on risk; projects may be modified or even dropped if risk is too great. Waterfall is much more structured and very reliant on getting the definition of requirements correct at the start; changes are harder to add in at a later stage. However, this forces the definition to be well understood. AO3 Spiral involves client feedback, prototypes and evolving projects. Better option where requirements may change. Waterfall is better where requirements are very clear to begin with and outcomes known. Spiral is better for risk management. If the programmer has a large team then waterfall may be more appropriate due to the clearly defined responsibilities at each stage. **Examiner's Comments** Most candidates gained some marks on this question. Most could explain that spiral was iterative and waterfall was done in linear stages, but many did not expand on this. Few candidates linked their answer to the complex computer program mentioned in the question. Some candidates also described waterfall as iterative.

			information is basic and communicated 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.		
	b	i	• 40	1	CAO
		ii	• 70	1	CAO
		iii	• 300	1	CAO
			Total	**12**	
8	a		1 mark per bullet up to a maximum of 2 marks, e.g: • To setup a loop… • To repeat (the same block of code) • for each line in the text file / until each value has been read	2 (AO3.3) (2)	**Examiner's Comments** This was also answered well although some candidates used the word 'iteration' in their answer without going to describe what this means. As this was in the question, further detail was required such as repeating or looping through the text file.

	b		1 mark per bullet up to a maximum of 7 marks:
			
			• Suitable logic for initialising variables outside of loop
			• Suitable logic for opening and closing Sales.txt
			• Suitable use of a loop which ends when the last line has been reached in Sales.txt
			• Suitable logic for reading each line of data from file variable
			• Suitable logic for increasing total (and count) including conversion from string to numeric data type
				7 (AO3.1) (3) (AO3.2) (4)	Some points (for example 5 & 6) can be combined onto one line.
					
					Example Solution:
					`procedure salesAnalysis():`
					`  dayCount = 0`
					`  salesTotal = 0`
					`  dailySales = openRead("Sales.txt")`

			• Suitable logic for calculating averages by dividing total by count
			• Suitable logic for printing the number of days, the total sales and the daily average
					`while NOT dailySales.endOfFile()`
					`  dailySaleFigure = float (percentageFile.readLine())`
					`  salesTotal + = dailySaleFigure`
					`  dayCount + = 1`
					`endwhile`
					`dailyAverage = salesTotal /dayCount`
					`print("Over " + dayCount + " days there was a total sales of £" _+ salesTotal + " and an average daily sale of £" + dailyAverage)`
					`dailySales.close()`
					`endprocedure`
					
					There are many different ways that this procedure could have been achieved. Therefore other alternative methods should be given credit.
					
					Examiner's Comments
					Most candidates achieved a few marks, but few achieved the maximum marks available. The question states that pseudocode or program code could be used so a variety of answers were given.
					
					The OCR pseudocode guide gives guidance on how to write pseudocode including data structures, constructs, and reading/writing to file. Some candidates wrote neither pseudocode or program code and therefore were given no marks.

					Many candidates remembered to open the sales file, but many forgot to close the file. Initialising the variables before the loop was also missed by a number of candidates. Increasing the total was included in many responses, although many forgot to cast the total to a numeric data type to enable this to work correctly. Many candidates achieved the last 2 marks for calculating the average and then displaying the calculated values. Some missed the final output as the text was incorrectly concatenated with the variable values. **Exemplar 3**  Exemplar 3 was given full marks. The candidate has gained the first mark for initialising the variables “numofdays” and “total”. The second mark has been awarded for the first line, which opens the sales file and closing the file after the

					loop has finished. The while loop gains the third mark which repeats until the end of the file is reached. The fourth mark is awarded for reading each line of the file correctly. The fifth mark correctly increases total in each iteration. Sale has been correctly converted to a "float" data type beforehand to make this possible. The sixth mark is awarded for calculating the average on the penultimate line. The final mark is awarded for printing the number of days, total sales and the daily average. This has been concatenated correctly with a plus symbol. Some candidates used the comma which was also acceptable. The syntax provided in this example relates to Python whereas some candidates wrote pseudocode or another high-level language. Some candidates lost marks for writing the steps in everyday English which is not what the question stipulated.
	c		1 mark per bullet up to a maximum of 2 marks, e.g: • Float/real/double/single/decimal/currency… • …because the values in the text file contain a decimal space… • …representing monetary values… • ….the values will be used in calculations	2 (AO3.3) (2)	**Examiner's Comments** This was generally answered well although some candidates picked Integer for the data type which would not be accurate given that averages are likely to be a decimal value. There are a range of data types that support decimal values, any of which would be acceptable as an answer.
			Total	**11**	
9		i	• Object – instantiated from class	3 AO1.1	

			• Method – action object performs / link to procedure/functions • Attribute – value held by object / link to variable		

		ii	• Class definition statement
			• Defining name and attendance attributes
			• Appropriate get methods for name **and** attendance that return a value and have no parameter
			• Appropriate set methods for name **and** attendance that take a parameter
			• …that restricts attendance to be 0 to 100.
				5 AO3.2	Example answer
					`class Worker`
					`  private name`
					`  private attendance`
					
					`  public function getName()`
					`  return name`
					`  end function`
					
					`  public function`
					`  getAttendance()`
					`  return attendance`
					`  end function`
					
					`  public procedure`
					`  setName(newName)`
					`  name = newName`
					`  end procedure`
					
					`  public procedure`
					`  setAttendance(newAttend)`
					`  if newAttend >=0 and`
					`  newAttend <=100 then`
					`  attendance =`
					`  newAttend`
					`  end if`
					`  end procedure`
					`end class`

			Total	**8**	
10			• Input two numbers <u>into two separate variables / other suitable data structure</u> • Correctly calculate integer division • Correctly calculate remainder • Print out both	4 AO3.2	Can be completed either by using MOD / DIV or by using repeated subtraction as in LMC example
			Total	**4**	
11	a		• 2.99 • nonUKprice • numtickets • return	4 AO3.2	Correct answer only. Penalise spelling if incorrect. Do not accept £ sign in first bullet point.
	b	i	• processing done away from the user's control/Client side processing could be altered • Browser may not support client-side language/ scripting could be turned off • Booking fee calculation needs to be correct for all locations	3 AO2.1	
		ii	To the customer any 2x1: • No need to submit to server and wait for response • Website will work more quickly for user To the company any 2x1: • Reduces load on the server • Will need to spend less on processing power/bandwidth	2 AO1.2	

<table>
<tr><td></td><td></td><td></td><td>Total</td><td>9</td><td></td></tr>
<tr><td>12</td><td>a</td><td></td><td>1 mark per bullet

• Calculation of result to 3
• Call with <code>thisFunction(theArray, num1=4, num2=7, num3=35)</code>
• Result = 5
• call with <code>thisFunction(theArray,num1=6,num2=7,num3=35)</code>
• (Result = 6) return of value 6

<table><tr><th>Function call</th><th>num1</th><th>num2</th><th>num3</th><th>result</th></tr><tr><td><code>thisFunction (theArray,0,7,35)</code></td><td>0</td><td>7</td><td>35</td><td>3</td></tr><tr><td><code>thisFunction (theArray,4,7,35)</code></td><td>4</td><td>7</td><td>35</td><td>5</td></tr><tr><td><code>thisFunction (thisArray,6,7,35)</code></td><td>6</td><td>7</td><td>35</td><td>6</td></tr></table></td><td>5
AO2.1
(3)
AO2.2
(2)</td><td></td></tr>
<tr><td></td><td>b</td><td></td><td>Binary search</td><td>1
AO2.1
(1)</td><td></td></tr>
<tr><td></td><td>c</td><td></td><td>1 mark per bullet to max 4, e.g.

• Recursion uses more memory…
• …iteration uses less memory
• Recursion declares new variables /variables are put onto the stack each time…</td><td>4
AO1.1
(2)
AO1.2
(2)</td><td></td></tr>
</table>

			• …iteration reuses the same variables • Recursive can run out of memory/stack space… • …while iteration cannot run out of memory • Recursion can express a problem more elegantly / in fewer lines of code… • …while iteration can take more lines of code / be harder to understand • Recursion will be self-referential / will call itself… • … whereas iteration does not		

	d		1 mark per bullet to max 6
			
			• Retains function call
			• Uses a loop
			• …that will loop until all elements inspected or value found
			• Updates num1 appropriately
			• Updates num2 appropriately
			• Returns -1 in the correct place if the value has not been found
			• Returns the result in the correct place if the value has been found
			
			e.g.
			`function thisFunction(theArray, num1, num2, num3)`
			
			`while (true)`
			`  result = num1 + ((num2 - num1) DIV 2)`
			`  if num2 < num1 then`
			`  return -1`
			`  else`
			`  if theArray[result] < num3 then`
			`  num1 = result + 1`
				6
				AO2.2
				(3)
				AO3.1
				(3)	

			elseif theArray[result] > num3 then
			num2 = result - 1
			else
			return result
			endif
			endif
			endwhile
			endfunction		

			Total	**16**	
13		i	• Iteration	1 AO2.2 (1)	
		ii	• 5	1 AO2.2 (1)	

		iii	1 mark per bullet up to a maximum of 4 marks, e.g:
			• Initialise Y and Z AND set X
			• Correct use of IF
			• Correct condition (e.g. if X >= Y then)
			• Assignment of Z in correct places
				4
				AO3.1
				(2)
				AO3.3
				(2)
					X Y Z variable alternatives are acceptable
					Solution:
					X = input()
					Y = 5
					Z = 0
					if X >= Y then
					Z = Y
					else
					Z = X
					Endif

			Total	**6**	
14			**Mark Band 3–High Level** **(7-9 marks)**	9 AO1.1	**AO1.1**

			The candidate demonstrates a thorough knowledge and understanding of relevant data collection methods. They have covered a range of ethical issues and relevant legislation; the material is generally accurate and detailed across all three. 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. The candidate has used a range of ethical implications and laws to justify their conclusion *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 relevant data collection methods. They have covered ethical issues and relevant legislation; 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 has either good consideration of ethical issues or legal issues or a weaker consideration of both. These are used to come to some justification although it may not be completely justifiable. *There is a line of reasoning presented with some structure. The information presented is in the most part relevant and supported by some evidence.*	(2) AO1.2 (2) AO2.1 (2) AO3.3 (3)	• Automatic Number Plate Recognition (ANPR) could be used. This makes use of OCRto turn number plates on camera into digital characters. • Use of a token/ticket could be used when cars enter the car park which is time stamped. • Staff could be employed to manually record the time cars enter the car park and then again when cars leave. • Sensor could be used to detect cars entering / leaving the car park. • Photographs of vehicles can be taken when cars enter / leave the car park. These can be time stamped. **AO2.1** • Data will be input into an algorithm that will work out the price based on amount of time car has spent in car park. • If the car park is considering storing videos / photographs of the cars then they will need to follow current data protection legislation as these may contain faces of people. • Car park users may see this as an invasion of privacy. Civil liberty issues as movements are being tracked / monitored. • Automated systems are likely to be more reliable and will produce statistical data that

			Mark Band 1-Low Level
			(1-3 marks)
			The candidate demonstrates a basic knowledge of relevant data collection methods. They have covered ethical issues or relevant legislation; the material is basic and contains some inaccuracies.
			The candidate makes a limited attempt to apply acquired knowledge and understanding to the context provided.
			The candidate provides comes to a conclusion though there may be little considerations of legal or ethical issues to back it up.
			
			The information is basic and communicated 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.
					can be analysed better.
					Manual methods may be prone to human error. Automatic methods will allow for quicker processing than if a human was doing it.
					• Lower overheads as no human to employ.
					• They could connect to the other databases (e.g. police and DVLA) to check that cars are genuine (e.g. can check if cars have been stolen)
					• If connecting the cameras over the internet, then this would make them susceptible to hackers and therefore security methods would need to be put in place.
					
					AO3.3
					
					• If automated systems are used then this can lead to improved customer satisfaction and lower costs due to streamlined system. However this could lead to less employment in the local area as computer systems taking over jobs.
					• Some people may be afraid of using carpark due to cameras.
					• Data protection laws should mean consumers are protected
					  ○ System secure
					  ○ Data only stored for a short time
					  ○ Only relevant data stored

			Total	**9**	