

Bubble and Insertion Sort Algorithms

Question No. 1

2(a)	<p>One mark for reference to:</p> <ol style="list-style-type: none">1 The use a variable as an index to the array2 A loop to iterate through the array3 An Inner loop (with a reducing range)4 Test if current element is greater than next element5 if so then swap elements6 Description of swap7 Attempt at efficient algorithm <p>Max 6 marks</p>	6
------	--	---

Question No. 2

1 mark per bullet point

- Correct outer loop stop
- Correct inner loop stop
- Correct < in the IF
- Correct `theArray(y + 1)`
- Correct `temp`
- Remainder matching pseudocode

Python

```
def bubbleSort():  
    for x in range (0, 10):  
        for y in range(0, 9):  
            if theArray[y] < theArray[y + 1]:  
                temp = theArray[y]  
                theArray[y] = theArray[y + 1]  
                theArray[y + 1] = temp
```

[6]

Bubble and Insertion Sort Algorithms

Question No. 3

5(a)	<p>Four from:</p> <ul style="list-style-type: none">• Uses a sorted and unsorted list• Takes first value and makes it sorted list // compare second item to first item• Find location of next value in the sorted list• ...description of suitable method (e.g. switching values, taking value out, comparing with sorted values)• Insert item in correct position in sorted list• Repeat until all items are in the sorted list (dependent on suitable method)	4
5(b)	<pre>Counter ← NumberOfItems - 2 REPEAT DataSwapped ← FALSE FOR CurrentValue ← 0 TO Counter // NumberOfItems - 2 IF DataList[CurrentValue] > DataList[CurrentValue + 1] THEN ValueTemp ← DataList[CurrentValue] DataList[CurrentValue] ← DataList[CurrentValue + 1] DataList[CurrentValue + 1] ← ValueTemp DataSwapped ← TRUE ENDIF ENDFOR UNTIL DataSwapped = FALSE</pre>	5

Bubble and Insertion Sort Algorithms

Question No. 4

```
(a) (i) FOR ThisPointer ← 2 TO 10
        // use a temporary variable to store item which is to
        // be inserted into its correct location
        Temp ← NameList[ThisPointer]
        Pointer ← ThisPointer - 1

        WHILE (NameList[Pointer] > Temp) AND (Pointer > 0)
            // move list item to next location
            NameList[Pointer + 1] ← NameList[Pointer]
            Pointer ← Pointer - 1
        ENDWHILE

        // insert value of Temp in correct location
        NameList[Pointer + 1] ← Temp
    ENDFOR
```

1 mark for each gap filled correctly

[7]

- (ii) The outer loop (FOR loop) is executed 9 times (1 mark)
 it is not dependant on the dataset (1 mark)

The Inner loop (WHILE loop) is not entered (1 mark)
 as the condition is already false at the first encounter (1 mark) [max 3]

- (b) (i) outer loop is executed 9 times (1 mark)
 inner loop is executed 9 times (for each iteration of the outer loop) (1 mark)
 not dependant on the dataset (1 mark) [max 2]

```
(ii) NumberOfItems ← 10
REPEAT
    NoMoreSwaps ← TRUE

    FOR Pointer ← 1 TO NumberOfItems - 1
        IF NameList[Pointer] > NameList[Pointer + 1]
            THEN
                NoMoreSwaps ← FALSE
                Temp ← NameList[Pointer]
                NameList[Pointer] ← NameList[Pointer + 1]
                NameList[Pointer + 1] ← Temp
            ENDIF
        ENDFOR
        NumberOfItems ← NumberOfItems - 1
    UNTIL NoMoreSwaps = TRUE
```

Mark as follows:

- change outer loop to a REPEAT/WHILE loop (1 mark)
- FOR loop has variable used for final value (1 mark)
- Initialise Boolean variable to TRUE (1 mark)
- set Boolean variable to FALSE in correct place (1 mark)
- number of items to consider on each pass decrements (1 mark)
- Correct stopping condition for REPEAT loop (1 mark)

[max 5]

Bubble and Insertion Sort Algorithms

Question No. 5

3(a)	CardData is partially sorted/ordered // more items in order/sorted	1
3(b)	<p>1 mark for each correct statement</p> <pre>01 ArraySize ← 10 02 FOR Pointer ← 2 TO ArraySize // 10 03 ValueToInsert ← CardData[Pointer] 04 HolePosition ← Pointer 05 WHILE (HolePosition > 1 AND (CardData[HolePosition - 1] > ValueToInsert)) 06 CardData[HolePosition] ← CardData[HolePosition - 1] 07 HolePosition ← HolePosition - 1 08 ENDWHILE 09 CardData[HolePosition] ← ValueToInsert 10 ENDFOR</pre>	7

Bubble and Insertion Sort Algorithms

Question No. 6

2(a)	<p>1 mark for each completed statement</p> <pre> 01 MaxIndex ← 20 02 NumberItems ← MaxIndex - 1 // 19 03 FOR Outer ← 1 TO MaxIndex - 1 // 19 04 FOR Inner ← 1 to NumberItems 05 IF ItemList[Inner] > ItemList[Inner + 1] 06 THEN 07 Temp ← ItemList[Inner] 08 ItemList[Inner] ← ItemList[Inner + 1] 09 ItemList[Inner + 1] ← Temp 10 ENDFOR 11 ENDFOR 12 NumberItems ← NumberItems - 1 13 ENDFOR </pre>	7
2(b)(i)	<p>1 mark per bullet</p> <ul style="list-style-type: none"> <input type="checkbox"/> Iterations continue // it continues doing comparisons <input type="checkbox"/> ...after the array is sorted 	2
2(b)(ii)	<p>1 mark per bullet to max 3</p> <ul style="list-style-type: none"> <input type="checkbox"/> Use of a flag to indicate if any swaps have taken place <input type="checkbox"/> If the inner loop has made all comparisons with no changes <input type="checkbox"/> ...flag/value set accordingly <input type="checkbox"/> A comparison checks the flag/value at the end of each inner loop <input type="checkbox"/> ...if it is sorted it breaks out/stops 	3
2(c)	<p>1 mark per bullet to max 4</p> <p>e.g.</p> <ul style="list-style-type: none"> <input type="checkbox"/> When the list is almost sorted ... <input type="checkbox"/> ...because it will stop as soon as it is sorted <input type="checkbox"/> When there are a large number of data items ... <input type="checkbox"/> ...because it will perform fewer comparisons/loops 	4

Bubble and Insertion Sort Algorithms

Question No. 7

3(a)	<p>1 mark for each completed statement:</p> <pre> 01 FOR Outer ← LENGTH(List)-1 TO 0 STEP -1 02 FOR Inner ← 0 TO (Outer - 1) 03 IF List[Inner] > List[Inner + 1] 04 THEN 05 Temp ← List[Inner] 06 List[Inner] ← List[Inner + 1] 07 List[Inner + 1] ← Temp 08 ENDIF 09 ENDFOR 10 ENDFOR </pre>	7
3(b)(i)	Ascending (must match answer to 3(a))	1
3(b)(ii)	<p>Line 03 Change the operator in the IF statement to < or <= rather than ></p>	1
3(c)	<p>1 mark per bullet</p> <ul style="list-style-type: none"> • Use of a (Boolean) flag... • ...Remainder of bubble correct • Set flag when a swap has been made... • ...Loop until a swap has not been made and then exit all loops <pre> Outer ← LENGTH(List)-1 REPEAT Inner ← 0 Swap ← FALSE REPEAT IF List[Inner] > List[Inner + 1] THEN Temp ← List[Inner] List[Inner] ← List[Inner + 1] List[Inner + 1] ← Temp Swap = TRUE ENDIF Inner ← Inner + 1 UNTIL Inner = Outer - 1 Outer ← Outer - 1 UNTIL Swap = FALSE OR Outer = 0 </pre>	4

Bubble and Insertion Sort Algorithms

Question No. 8

4(b)(i)	1 mark per shaded section	4																																																																																																									
	<table><tr><th rowspan="2">Count</th><th rowspan="2">TempValue</th><th rowspan="2">Sorted</th><th colspan="6">ArrayData</th></tr><tr><th>0</th><th>1</th><th>2</th><th>3</th><th>4</th><th>5</th></tr><tr><td>0</td><td>" "</td><td>TRUE</td><td>5</td><td>20</td><td>12</td><td>25</td><td>32</td><td>29</td></tr><tr><td>1</td><td>12</td><td>FALSE</td><td></td><td>12</td><td>20</td><td></td><td></td><td></td></tr><tr><td>2</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr><tr><td>3</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr><tr><td>4</td><td>29</td><td>(FALSE)</td><td></td><td></td><td></td><td></td><td>29</td><td>32</td></tr><tr><td>0</td><td></td><td>TRUE</td><td></td><td></td><td></td><td></td><td></td><td></td></tr><tr><td>1</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr><tr><td>2</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr><tr><td>3</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr><tr><td>4</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr></table>	Count	TempValue	Sorted	ArrayData						0	1	2	3	4	5	0	" "	TRUE	5	20	12	25	32	29	1	12	FALSE		12	20				2									3									4	29	(FALSE)					29	32	0		TRUE							1									2									3									4									
Count	TempValue				Sorted	ArrayData																																																																																																					
		0	1	2		3	4	5																																																																																																			
0	" "	TRUE	5	20	12	25	32	29																																																																																																			
1	12	FALSE		12	20																																																																																																						
2																																																																																																											
3																																																																																																											
4	29	(FALSE)					29	32																																																																																																			
0		TRUE																																																																																																									
1																																																																																																											
2																																																																																																											
3																																																																																																											
4																																																																																																											
4(b)(ii)	1 mark per bullet point <ul style="list-style-type: none">Initialising a counter variable to 0 and must be the same variable used to access array elementsWhile loop checking counter is < 5 or <= 4Incrementing counter inside the loop and outside IF, and remainder of algorithm completed (The IF to ENDIF) e.g. <pre>Count ← 0 WHILE Count < 5 IF ArrayData[Count] > ArrayData[Count + 1] THEN TempValue ← ArrayData[Count + 1] ArrayData[Count + 1] ← ArrayData[Count] ArrayData[Count] ← TempValue Sorted ← False ENDIF Count ← Count + 1 ENDWHILE</pre>	3																																																																																																									
4(b)(iii)	Bubble sort	1																																																																																																									
4(b)(iv)	One from: <ul style="list-style-type: none">Insertion sortMerge sortQuick sort	1																																																																																																									

Bubble and Insertion Sort Algorithms

Question No. 9

2(f)(i)	Insertion sort	1																																																																																																								
2(f)(ii)	One from: <ul style="list-style-type: none">• Bubble sort• Merge sort	1																																																																																																								
2(f)(iii)	1 mark per shaded section <table><tr><th rowspan="2">Item</th><th rowspan="2">NumberOfScores</th><th rowspan="2">InsertScore</th><th rowspan="2">Index</th><th colspan="5">ArrayData</th></tr><tr><th>0</th><th>1</th><th>2</th><th>3</th><th>4</th></tr><tr><td></td><td></td><td></td><td></td><td>99</td><td>125</td><td>121</td><td>109</td><td>115</td></tr><tr><td>1</td><td>5</td><td>125</td><td>0</td><td>(125)</td><td></td><td></td><td></td><td></td></tr><tr><td>2</td><td></td><td>121</td><td>1</td><td></td><td></td><td>125</td><td></td><td></td></tr><tr><td></td><td></td><td></td><td>0</td><td></td><td>121</td><td></td><td></td><td></td></tr><tr><td>3</td><td></td><td>109</td><td>2</td><td></td><td></td><td></td><td>125</td><td></td></tr><tr><td></td><td></td><td></td><td>1</td><td></td><td></td><td>121</td><td></td><td></td></tr><tr><td></td><td></td><td></td><td>0</td><td></td><td>109</td><td></td><td></td><td></td></tr><tr><td>4</td><td></td><td>115</td><td>3</td><td></td><td></td><td></td><td></td><td>125</td></tr><tr><td></td><td></td><td></td><td>2</td><td></td><td></td><td></td><td>121</td><td></td></tr><tr><td></td><td></td><td></td><td>1</td><td></td><td></td><td>115</td><td></td><td></td></tr></table>	Item	NumberOfScores	InsertScore	Index	ArrayData					0	1	2	3	4					99	125	121	109	115	1	5	125	0	(125)					2		121	1			125						0		121				3		109	2				125					1			121						0		109				4		115	3					125				2				121					1			115			7
Item	NumberOfScores					InsertScore	Index	ArrayData																																																																																																		
		0	1	2	3			4																																																																																																		
				99	125	121	109	115																																																																																																		
1	5	125	0	(125)																																																																																																						
2		121	1			125																																																																																																				
			0		121																																																																																																					
3		109	2				125																																																																																																			
			1			121																																																																																																				
			0		109																																																																																																					
4		115	3					125																																																																																																		
			2				121																																																																																																			
			1			115																																																																																																				

Bubble and Insertion Sort Algorithms

Question No. 10

5(a)	<p>Four from:</p> <ul style="list-style-type: none">• Uses a sorted and unsorted list• Takes first value and makes it sorted list // compare second item to first item• Find location of next value in the sorted list• ...description of suitable method (e.g. switching values, taking value out, comparing with sorted values)• Insert item in correct position in sorted list• Repeat until all items are in the sorted list (dependent on suitable method)	4
5(b)	<pre>Counter ← NumberOfItems - 2 REPEAT DataSwapped ← FALSE FOR CurrentValue ← 0 TO Counter // NumberOfItems - 2 IF DataList[CurrentValue] > DataList[CurrentValue + 1] THEN ValueTemp ← DataList[CurrentValue] DataList[CurrentValue] ← DataList[CurrentValue + 1] DataList[CurrentValue + 1] ← ValueTemp DataSwapped ← TRUE ENDIF ENDFOR UNTIL DataSwapped = FALSE</pre>	5

Bubble and Insertion Sort Algorithms

Question No. 11

8(a)	<p>One mark for each correct marking point (Max 2)</p> <ul style="list-style-type: none"> The initial order of the data The number of data items to be sorted The efficiency of the sorting algorithm 	2
8(b)	<p>One mark for each marking point (max 6)</p> <p>MP1 Use of FOR loop to cycle through the <u>whole year group</u></p> <p>MP2 Temporary storage of the score being 'inserted'</p> <p>MP3 Temporary storage of the corresponding name elements</p> <p>MP4 Use of WHILE loop with correct exit clause</p> <p>MP5 Moving of all three elements of data to next array elements</p> <p>MP6 Correct updating of counter variable</p> <p>MP7 Final insertion of all three data elements</p> <p>Example algorithm</p> <pre> YearSize ← 249 FOR Student ← 2 to YearSize Temp1 ← Score[Student] Temp2 ← Name[Student,1] Temp3 ← Name[Student,2] Counter ← Student WHILE Counter > 1 AND Score[Counter - 1] < Temp1 Score[Counter] ← Score[Counter - 1] Name[Counter,1] ← Name[Counter - 1,1] Name[Counter,2] ← Name[Counter - 1,2] Counter ← Counter - 1 ENDWHILE Score[Counter] ← Temp1 Name[Counter,1] ← Temp2 Name[Counter,2] ← Temp3 NEXT Student </pre>	6