

Secondary – 6

Computer Science

9618/4

Paper 4 - Practical Marking scheme

23rd, August 2024

1 hour 30 minutes

INSTRUCTIONS

This mark scheme is published as an aid to teachers and candidates, to indicate the requirements of the examination. It shows the basis on which Examiners were instructed to award marks. It does not indicate the details of the discussions that took place at an Examiners' meeting before marking began, which would have considered the acceptability of alternative answers.

Mark schemes should be read in conjunction with the question paper and the Principal Examiner Report for Teachers.

Cambridge International will not enter into discussions about these mark schemes.

Cambridge International is publishing the mark schemes for the May/June 2022 series for most Cambridge IGCSE, Cambridge International A and AS Level and Cambridge Pre-U components, and some Cambridge O Level components.

Q no	Answer	Marks
1	<p>1 mark per bullet point</p> <ul style="list-style-type: none"> • Correct outer loop stop • Correct inner loop stop • Correct < in the IF • Correct theArray(y + 1) • Correct temp • Remainder matching pseudocode <p>Example code:</p> <pre> 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 Java public static void bubbleSort(){ int temp; for (int x = 0; x < 10; x++){ for (int y = 0; y < 9; y++){ if(theArray[y] < theArray[y+1]){ temp = theArray[y]; theArray[y] = theArray[y+1]; theArray[y+1] = temp; } } } } </pre>	10
2 (a)	<p>1 mark per mark point</p> <ul style="list-style-type: none"> • Declaring variables: head pointer, tail pointer and number of items all initialised as 0 (integer) • QueueArray declared as 1D array as string with 10 elements <p>Example program code:</p> <pre> Java public static void main(String[] args){ String[] QueueArray = new String[10]; Integer QueueHeadPointer = 0; Integer QueueTailPointer = 0; Integer NumberOfItems = 0; } Python QueueArray = ['', '', '', '', '', '', '', '', '', ''] #string QueueHeadPointer = 0 #integer QueueTailPointer = 0 #integer NumberOfItems = 0 #integer </pre>	2

<p>2 (b)</p>	<p>1 mark per complete statement (5) 1 mark for function heading and end, dealing with ByRef 1 mark for remainder of function correct and following the logic</p>	<p>7</p>
	<pre> FUNCTION Enqueue(BYREF QueueArray[] : STRING, BYREF HeadPointer : Integer, BYREF TailPointer : Integer, NumberItems : INTEGER, DataToAdd : STRING) RETURNS BOOLEAN IF NumberItems = 10 THEN RETURN FALSE ENDIF QueueArray[TailPointer] ← DataToAdd IF TailPointer >= 9 THEN TailPointer ← 0 ELSE TailPointer ← TailPointer + 1 ENDIF NumberItems ← NumberItems + 1 RETURN TRUE ENDFUNCTION </pre> <p>Example program code:</p> <p>Java</p> <pre> public static Boolean Enqueue(String DataToAdd){ if(NumberOfItems == 10){ return false; } QueueArray[QueueTailPointer] = DataToAdd; if(QueueTailPointer >= 9){ QueueTailPointer = 0; }else{ QueueTailPointer = QueueTailPointer + 1; } NumberOfItems = NumberOfItems + 1; return true; } </pre> <p>Python</p> <pre> def Enqueue(Queue, Head, Tail, NumItems, InputData): if NumItems >= 10: return (False, Queue, Head, Tail, NumItems) Queue[Tail] = InputData if Tail >= 9: Tail = 0 else: Tail = Tail + 1 NumItems = NumItems + 1 return (True, Queue, Head, Tail, NumItems) </pre>	<p>8</p>
<p>2 (c)</p>	<p>1 mark per mark point to max 6</p> <ul style="list-style-type: none"> • Function header and end • checking if queue is empty ... • ... returning False • If not empty accessing and returning item at head pointer • ... incrementing head pointer ... • ... changing head pointer to 0 if it's more than 9 after incrementing • ... decrement number of items 	

	<p>Example program code:</p> <p>Java</p> <pre>public static String Dequeue(){ if(NumberOfItems == 0){ return "FALSE"; }else{ String ReturnValue = QueueArray[QueueHeadPointer]; QueueHeadPointer = QueueHeadPointer + 1; if(QueueHeadPointer >= 9){ QueueHeadPointer = 0; } NumberOfItems = NumberOfItems - 1; return ReturnValue; } }</pre> <p>Python</p> <pre>def Dequeue(Queue, Head, Tail, NumItems): if NumItems == 0: return (false, Queue, Head, Tail, NumItems) else: ReturnValue = Queue(Head) Head = Head + 1 if Head >= 9: Head = 0 NumItems = NumItems - 1 return(ReturnValue, Queue, Head, Tail, NumItems)</pre> <p>1 mark per mark point</p> <p>2 (d)</p> <ul style="list-style-type: none"> • Taking 11 inputs... • ... calling Enqueue with each of the 11 inputs ... • ... outputting an appropriate message if added or not added • Calling Dequeue twice ... • ... outputting return value each time <p>Example program code:</p> <p>Java</p> <pre>public static void main(String args[]){ String InputString; for(Integer x = 0; x < 11; x++){ System.out.println("Enter a string"); Scanner scanner = new Scanner(System.in); InputString = scanner.nextLine(); if(Enqueue(InputString)){ System.out.println("Successful"); }else{ System.out.println("Unsuccessful"); } } System.out.println(Dequeue()); System.out.println(Dequeue()); }</pre> <p>Python</p> <pre>for x in range(0, 11): InputString = input("Enter a string") ReturnValue, QueueArray, QueueHeadPointer, QueueTailPointer, NumberOfItems = Enqueue(QueueArray, QueueHeadPointer, QueueTailPointer, NumberOfItems, InputString) if ReturnValue == True: print("Successful") else: print("Unsuccessful") ReturnValue, QueueArray, QueueHeadPointer, QueueTailPointer, NumberOfItems = Dequeue(QueueArray, QueueHeadPointer, QueueTailPointer, NumberOfItems) print(ReturnValue) ReturnValue, QueueArray, QueueHeadPointer, QueueTailPointer, NumberOfItems = Dequeue(QueueArray, QueueHeadPointer, QueueTailPointer, NumberOfItems) print(ReturnValue)</pre> <p>2 (e)</p> <p>1 mark for showing inputs and outputs with screenshot</p>	<p>7</p> <p>1</p>
--	--	-------------------