AQA Computer Science A-Level

4.4.4 Classification of algorithms

Past Paper Questions

Additional Specimen Paper 1

0 1 **Table 1** contains a list of problems.

Table 1

Letter	Problem
A	Finding the shortest route that visits all nodes in a graph exactly once
B	Finding the shortest path between two nodes in a graph
C	Finding an item in a large unordered list
D	Finding a route between two nodes in a graph
E	Finding out if any program will eventually stop if given a particular input

0 1 . 1 State the letter (**A-E**) that corresponds to the problem that Dijksta's Algorithm is designed to solve.

Write the letter corresponding to the correct answer in the box provided in your Electronic Answer Document.

[1 mark]

0 1 . 2 State the letter (**A-E**) that corresponds to an intractable problem.

Write the letter corresponding to the correct answer in the box provided in your Electronic Answer Document.

[1 mark]

0 1 . 3 Explain the significance of problem **E** for computation.

[1 mark]

) 2 **Figure 1** contains pseudo-code for a recursive merge sort algorithm. **Figure 2** contains pseudo-code for an algorithm called `Merge` that is called by the merge sort algorithm in **Figure 1**.

Figure 1

```
FUNCTION MergeSort(L, S, E)
  IF S < E THEN
    M ← (S + E) DIV 2
    L1 ← MergeSort(L, S, M)
    L2 ← MergeSort(L, M + 1, E)
    RETURN Merge(L1, L2)
  ELSE
    RETURN Append([], L[S])
  ENDIF
ENDFUNCTION
```

Figure 2

```
FUNCTION Merge(L1, L2)
  L3 ← []
  WHILE Len(L1) > 0 AND LEN(L2) > 0
    IF L1[1] < L2[1] THEN
      L3 ← Append(L2[1], L3)
      L2 ← RemoveFirstItem(L2)
    ELSE
      L3 ← Append(L1[1], L3)
      L1 ← RemoveFirstItem(L1)
    ENDIF
  ENDWHILE
  WHILE Len(L1) > 0
    L3 ← Append(L1[1], L3)
    L1 ← RemoveFirstItem(L1)
  ENDWHILE
  WHILE Len(L2) > 0
    L3 ← Append(L2[1], L3)
    L2 ← RemoveFirstItem(L2)
  ENDWHILE
  RETURN L3
ENDFUNCTION
```

The `RemoveFirstItem` function takes a list and returns a list that contains all the items in the original list except the first one. For example, if `Names` is the list `["Gemma", "Richard", "Georgina", "Margaret"]` then the function call `RemoveFirstItem(Names)` will return the list `["Richard", "Georgina", "Margaret"]`.

The `Len` function takes a list and returns the number of items that are in the list. For example, if `Names` is the list `["Gemma", "Richard", "Georgina", "Margaret"]` then the function call `Len(Names)` will return the value of `4`.

The `Append` function takes an item and a list and returns a list that has all the items from the original list followed by the item. For example, if `Names` is the list `["Gemma", "Richard", "Georgina", "Margaret"]` then the function call `Append("Matt", Names)` will return the list `["Gemma", "Richard", "Georgina", "Margaret", "Matt"]`.

The first item in the list has an index of 1.

0 2 . 4 What is the time complexity for the `MergeSort` algorithm shown in **Figure 1**?

[1 mark]

Additional Specimen Paper 2

0 8 The `MOD` function calculates the remainder after an integer division has been carried out.

`MOD(x,y)` computes the reminder when `x` is divided by `y`.

For example, `MOD(13,3)` = 1 since 13 divided by 3 is 4 remainder 1.

0 8 . 1 What is the co-domain of the `MOD` function?

[1 mark]

June 2011 Comp 3

1 (c) Tick **one** box to indicate the order of time complexity of the binary search method.

Order of time complexity	Tick one box
$O(\log_2 n)$	
$O(n)$	
$O(n^2)$	

(1 mark)

June 2012 Comp 3

3 (a) Time complexity is one of the two measures that are used to describe the complexity of an algorithm.

What is the other measure?

.....

(1 mark)

3 (b) A student has been asked to write a program to list duplicate entries in a file containing a list of words. **Figure 2** shows her first attempt at planning an algorithm. The algorithm will not work in all circumstances.

Figure 2

```
Open file
N ← Number of items in file
For Pos1 ← 1 To N Do
  Read item at position Pos1 in file into variable W1
  For Pos2 ← 1 To N Do
    Read item at position Pos2 in file into variable W2
    If W1 = W2 And Not (Pos1 = Pos2)
      Then Output 'Duplicate: ' , W1
    EndIf
  EndFor
EndFor
Close file
```

The basic operation in the algorithm is the `If` statement that compares two words.

The contents of a particular file are shown in **Figure 3**.

Figure 3

File position	Item
1	Rope
2	Dagger
3	Rope

3 (b) (ii) Tick **one** box in the table below to indicate the correct order of time complexity of the algorithm that the student has written.

Order of time complexity	Tick one box
$O(a^n)$	
$O(n)$	
$O(n^2)$	

(1 mark)

3 (b) (iii) Justify your answer to part (b)(ii).

.....

.....

.....

.....

(2 marks)

11 (a) Complete the missing parts of the question posed by the Halting problem in **Figure 10**.

Figure 10

Is it possible in general to ..... that can tell, given any program and its inputs and without ....., whether the given program with its given inputs will halt?

(2 marks)

11 (b) What is the significance of the Halting problem?

.....

.....

(1 mark)

June 2017 Paper 1

0 3 **Table 2** lists some well-known algorithms.

Table 2

Algorithm
Linear search
Merge sort
Binary search
Post-order tree-traversal

0 3 . 1 Which of the algorithms listed in **Table 2** has $O(n \log n)$ time complexity?

[1 mark]

0 3 . 2 How many of the algorithms listed in **Table 2** are algorithms used to solve tractable problems?

[1 mark]

0 3 . 3 State the time complexity for the bubble sort algorithm in terms of n, where n is the number of items in the list to be sorted.

[1 mark]

0 3 . 4 Explain why the bubble sort algorithm has the time complexity stated in your answer to **0 3 . 3**

[2 marks]

0 5 . 4 Explain what is meant by a heuristic technique, giving an example of a heuristic technique that might reduce the time taken by the shortest path algorithm.

[2 marks]

June 2013 Comp 3

6 An algorithm is a sequence of unambiguous instructions for solving a problem.

6 (a) Three different algorithms, A, B and C, have the following orders of time complexity:

Algorithm A: $O(a^n)$
Algorithm B: $O(n^2)$
Algorithm C: $O(n)$

List the algorithms A, B and C in order with the most efficient at the top of the list.

Most efficient: .....

.....

Least efficient: .....

(1 mark)

6 (b) Some problems are intractable.

6 (b) (i) What does it mean for a problem to be described as *intractable*?

.....

.....

.....

(2 marks)

6 (b) (ii) One of the problems listed in the table below is intractable.

Place **one** tick next to the intractable problem.

Problem	Intractable? (Tick one)
The travelling salesman problem	
The problem of sorting a list of names into alphabetic order	
The Halting problem	

(1 mark)

Specimen Paper 1

0 3 . 6 What is meant by an intractable problem?

[2 marks]

0 3 . 7 What approach might a programmer take if asked to solve an intractable problem?

[2 marks]

Table 6 contains a list of orders of time complexity (in no particular order).

Table 6

Order of time complexity
O(1)
$O(n^2)$
O(log n)
$O(k^n)$
O(n)

Which of the orders of time complexity given in **Table 6**:

0 4 . 4 could be the time complexity of an intractable problem?

[1 mark]

0 4 . 5 is the time complexity for a binary search?

[1 mark]

0 4 . 6 is the time complexity for getting the first item in a list?

[1 mark]

0 4 . 7 is the time complexity for a linear-search algorithm?

[1 mark]

0 4 . 8 Explain why a linear-search has the order of time complexity given in your answer to question 0 4 . 7 .

[2 marks]