Oxford Cambridge and RSA

Wednesday 18 June 2025 – Morning

A Level Computer Science

H446/02 Algorithms and programming

Time allowed: 2 hours 30 minutes

You can use:
- a ruler (cm/mm)
- an HB pencil

Do not use:
- a calculator

Please write clearly in black ink. **Do not write in the barcodes.**

Centre number | | | | | | Candidate number | | | | |

First name(s) _____

Last name _____

INSTRUCTIONS
- Use black ink. You can use an HB pencil, but only for graphs and diagrams.
- Write your answer to each question in the space provided. If you need extra space use the lined page at the end of this booklet. The question numbers must be clearly shown.
- Answer **all** the questions.

INFORMATION
- The total mark for this paper is **140**.
- The marks for each question are shown in brackets **[ ]**.
- Quality of extended response will be assessed in questions marked with an asterisk (*).
- This document has **32** pages.

ADVICE
- Read each question carefully before you start your answer.

© OCR 2025 [601/4911/5]
DC (PQ/CGW) 337417/5
OCR is an exempt Charity

Section A

1 Data in a computer program needs to be sorted.

(a)
(i) Describe the steps a bubble sort will take to sort items into **ascending** order.

.....

.....

.....

.....

.....

.....

.....

.....

.....

..... **[5]**

(ii) Darcie and Felix have both written a program that will perform a bubble sort to put these numbers into **ascending** order.

2	1	3	4	5	6	7

Darcie's version will complete **six** passes. However, Felix's version will only need to complete **two** passes.

Explain how Felix may have made his version more efficient than Darcie's version.

.....

.....

.....

..... **[2]**

© OCR 2025

(b) A second type of sorting algorithm is an insertion sort.

Describe how an insertion sort can be used to put the following data into **ascending** order.

20	8	15	36

.....

.....

.....

.....

.....

.....

.....

.....

.....

..... **[5]**

© OCR 2025

(c) A third type of sorting algorithm is a quick sort.

The following quick sort algorithm is made up of two functions called `partition()` and `quickSort()`. The algorithm will sort an integer array into **ascending** numerical order.

```
01   function partition(array, low, high)
02      pivot = array[high]
03      index = low - 1
04      for count = low to high-1
05         if array[count] < pivot then
06            index = index + 1
07            temp = array[index]
08            array[index] = array[count]
09            array[count] = temp
10         endif
11      next count
12      temp = array[index + 1]
13      array[index + 1] = array[high]
14      array[high] = temp
15      return index + 1
16   endfunction
17
18   function quickSort(array, low, high)
19      if low < high then
20         partitionIndex = partition(array, low, high)
21         quickSort(array, low, partitionIndex - 1)
22         quickSort(array, partitionIndex + 1, high)
23      endif
24   endfunction
```

(i) Identify which of the **two** functions is recursive **and** give all of the line numbers where recursive calls are made.

Recursive function name .....

Recursive call line numbers .....

[2]

© OCR 2025

(ii) Identify the type of iteration used in the function `partition()`.

..... **[1]**

(iii) Describe the role of the function `partition()` in the quick sort algorithm.

.....

.....

.....

.....

.....

..... **[3]**

(iv) The algorithm needs to be changed to sort the integer array into **descending** numerical order.

Identify the line number that needs to be changed **and** write the amended line of code.

Line number .....

Amended line of code .....

.....

[2]

(v) The quick sort is an example of a divide-and-conquer algorithm.

Describe what divide-and-conquer means.

.....

.....

.....

..... **[2]**

© OCR 2025

2* A shopping website requires customers to create a username and password.

(a) The usernames and passwords are stored securely in a file. The file contains the username and passwords for 50 million users which are sorted in alphabetical order by username.

The file can be searched using a linear search or a binary search.

The big-O complexities for a linear search and a binary search are given in the table:

Search algorithm	**Worst-case time**	**Average-case time**	**Worst-case space**
Linear	O(n)	O(n)	O(1)
Binary	O(log n)	O(log n)	O(1)

Compare the use of a linear search and a binary search in this scenario.

You should include the following in your answer:

- the features and preconditions of a linear and binary search
- the benefits and drawbacks of each search in this scenario
- a conclusion justifying which search algorithm should be used in this scenario.

You should make reference to the big-O complexities in your answer. **[9]**

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

© OCR 2025

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

(b) The website collects data from customers' activities once they log on. For example, the items they view and purchase. The company that owns the website would like to use data mining to improve their website and customer experience.

(i) Describe what data mining means.

.....

.....

.....

.....

.....

..... **[3]**

(ii) Explain how the company can make use of data mining to improve their website and customer experience.

.....

.....

.....

..... **[2]**

© OCR 2025

BLANK PAGE

DO NOT WRITE ON THIS PAGE

© OCR 2025

3 A program is designed to use a stack data structure.

The current contents of part of the stack and its current pointer value are shown:

5	
4	
3	
2	"l"
1	"n"
0	"k"

`pointer = 2`

(a) Show the contents of the stack and its pointer value after the data items "t" and "s" are inserted in the order given.

5	
4	
3	
2	
1	
0	

`pointer =` .....

[2]

© OCR 2025

(b) The stack is stored using a 0-based 1-dimensional array called `theStack` with 20 elements.

The function `push()` stores its parameter into the next space in the stack if it is not full. The function returns `true` if it is successfully stored and returns `false` if the stack is full.

Complete the function `push()` using pseudocode or program code.

```
function push(data)

   if pointer == ............................................................ then

      return false

   else

      pointer = ............................................................

      theStack[pointer] = ..............................................................

      ............................................................

   endif

endfunction
```

[4]

© OCR 2025

(c) A `pointer` value of `-1` indicates that the stack is empty.

The function `pop()` returns the next element in the stack. If the stack is empty it returns `"false"`. The function updates the `pointer` value when appropriate.

Write the function `pop()` using pseudocode or program code.

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

..... **[6]**

(d) A program is designed to reverse a set of data using the stack.

The main program takes 10 characters separately as input from the user and uses the function `push()` to insert these into the stack.

Once all 10 characters have been inserted into the stack, the program then uses the function `pop()` to output all 10 characters from the stack.

Write the main program to achieve this using pseudocode or program code.

.....

.....

.....

.....

.....

.....

.....

.....

..... **[5]**

© OCR 2025

4* A processor is being designed to use pipelining and will need to be able to process millions of instructions.

For each instruction the following stages will be performed:

A the instruction will be fetched from memory
B the instruction will be decoded
C the instruction will be executed

The first four instructions to be processed will be instructions 1, 2, 3 and 4. The design allocates 1-time interval for each of the three stages A, B, C.

Discuss the use of pipelining by this processor.

You should include the following in your answer:

- the features of pipelining
- how pipelining can be used to manage the first four instructions
- a conclusion justifying the use of pipelining in this scenario. **[9]**

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

© OCR 2025

.....

.....

.....

.....

.....

.....

.....

.....

.....

© OCR 2025

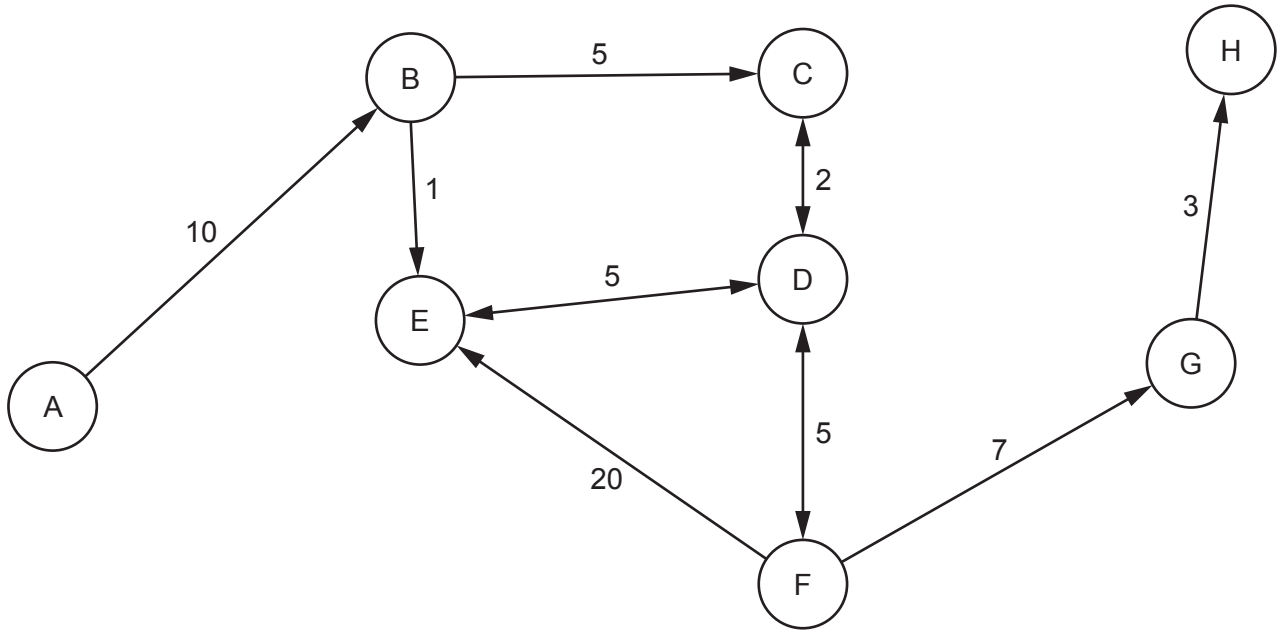
5 A program is being written that allows users to travel between different planets in a virtual world. Spaceships are used to travel between planets and there are only set routes that they use.

Fig. 5 shows a directed graph. The nodes represent the different planets and the edges represent the routes the spaceships can travel.

The arrows on the edges show which way the spaceships can travel. For example, a spaceship can travel from planet B to planet C, but **not** from planet C to planet B.

The value on each edge is the cost in space dollars to travel on that route.

Fig. 5



(a) Explain why this graph is a visualisation of the problem.

.....

.....

.....

..... **[2]**

© OCR 2025

(b) A user needs to travel from planet A to planet H. They need to travel the least expensive route.

Show how Dijkstra's algorithm can be used on the directed graph shown in **Fig. 5** to find the shortest path from start planet A to the end planet H.

You must state the nodes on the final path and the distance of this path. Show your working.

You may use the table to give your answer.

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

Node	Distance travelled	Previous node

Final path: .....

Distance: .....

[7]

© OCR 2025

(c) The A* algorithm can also be used to find the shortest path in a graph.

Describe how an A* algorithm uses heuristics to be an efficient algorithm.

.....

.....

.....

..... **[2]**

6 A student is writing a program in a high-level programming language.

(a) The student writes the program using an Integrated Development Environment (IDE).

Complete the table by identifying **and** describing **three** features of an IDE that the student can use to **debug** their program.

Debugging feature	Description
....	....
....	....
....	....

[6]

© OCR 2025

(b) The program the student is developing includes both global and local variables.

(i) Explain the difference between a global variable and a local variable.

.....

.....

.....

..... **[2]**

(ii) Describe **one** drawback of using global variables in a program.

.....

.....

.....

..... **[2]**

(iii) Describe **one** drawback of using local variables in a program.

.....

.....

.....

..... **[2]**

(c) A variable can be passed as a parameter into a function by value or by reference.

Describe what parameter passing by value means.

.....

.....

.....

..... **[2]**

© OCR 2025

(d) An algorithm is needed to count how many times each of the numbers from 1 to 9 appear in an array called `myArray`. Another array called `checked` is used to store each number found in `myArray` and the number of times it appears. For example, the contents of `myArray` are:

7	3	6	7	5	3	2	3	6

Based on these contents in `myArray`, the contents in `checked` would be:

7	2
3	3
6	2
5	1
2	1
–1	0
–1	0
–1	0
–1	0

–1 indicates an unused space in the array as the numbers 1, 4, 8 and 9 have **not** been found.

The student has written the following pseudocode for this program.

The pseudocode contains **several** errors.

```
array myArray[9]
myArray = [7,3,6,7,5,3,2,3,6]
array checked[9,2]
checked = [[-1,0],[-1,0],[-1,0],[-1,0],[-1,0],[-1,0],[-1,0],
           [-1,0],[-1,0]]
numbersFound = 1
for x = 0 to myArray.length
   found = false
   for y = 0 to checked.length
      if checked[y,0] == myArray[y] then
         checked[y,1] = checked[y,1] + 1
         found = true
      endif
   next y
   if found == true then
      checked[numbersFound,0] = myArray[x]
      checked[numbersFound,1] = 0
      numbersFound = numbersFound + 1
   endif
next y
```

© OCR 2025

Identify the line number of any **four** errors **and** write the corrected line for each error.

Error 1

Line number .....

Corrected line .....

Error 2

Line number .....

Corrected line .....

Error 3

Line number .....

Corrected line .....

Error 4

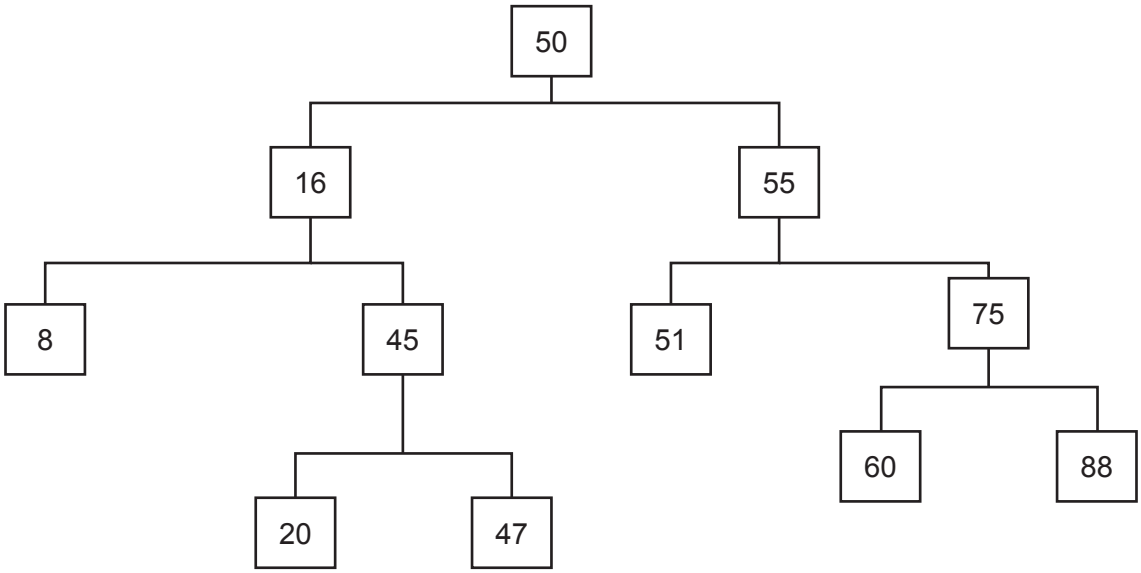
Line number .....

Corrected line .....

[4]

© OCR 2025

7 An ordered binary search tree contains the following data:



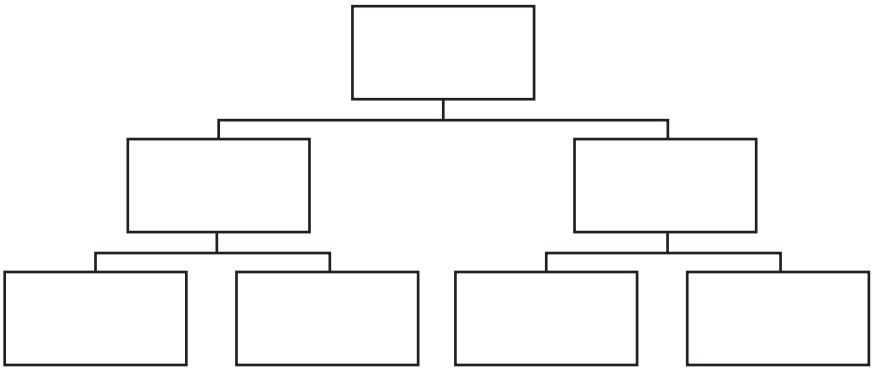
(a) Add the following data items in the given order to the binary search tree above.

5 30 53 57 90

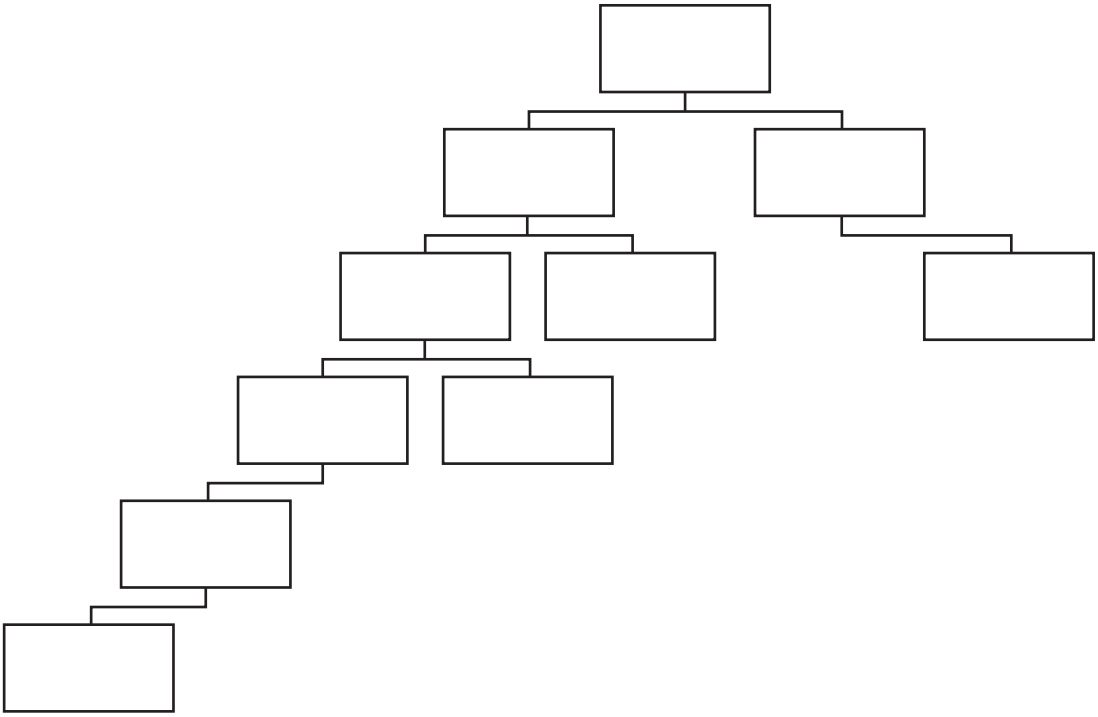
[3]

© OCR 2025

(b) A balanced binary search tree has data consistently at each level, for example:



An unbalanced binary search tree is uneven, for example:



Describe the drawback of an unbalanced binary search tree when searching for a value.

.....

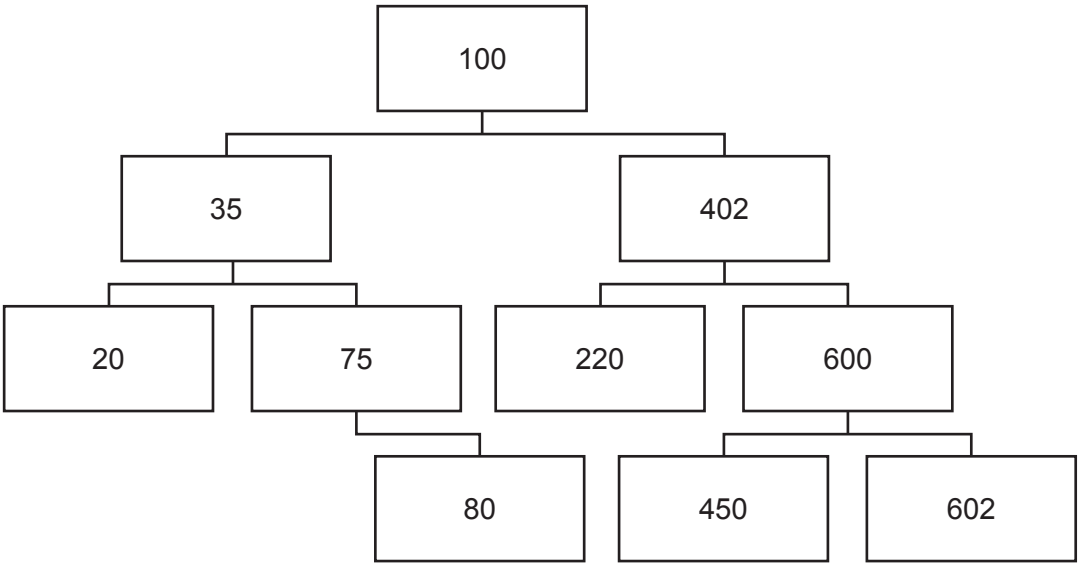
.....

.....

..... **[2]**

© OCR 2025

(c) A different binary search tree stores the following data:



Give the output that will be produced from a depth-first (post-order) traversal and a breadth-first traversal of this binary search tree.

Depth-first (post-order) .....

.....

Breadth-first .....

.....

[4]

© OCR 2025

BLANK PAGE

DO NOT WRITE ON THIS PAGE

© OCR 2025

Section B

8 A linked list data structure is designed using object-oriented programming.

The design includes two classes called `node` and `linkedList`.

(a) Describe the features of a linked list data structure.

.....

.....

.....

.....

.....

..... **[3]**

(b) The class design for `node` is shown here.

class: `node`

attributes:
`private value : integer`
`private nextNode : node`

methods:
`new()`
`getValue()`
`getNextNode()`
`setValue()`
`setNextNode()`

`new()` is the constructor method. The constructor method takes a parameter and assigns this to the `value` attribute.

(i) The method `getNextNode()` returns the attribute `nextNode`.

Write the method `getNextNode()` using pseudocode or program code.

.....

.....

.....

..... **[2]**

© OCR 2025

(ii) The method `setValue()` takes an integer as a parameter and assigns this to the attribute `value`.

Write the method `setValue()` using pseudocode or program code.

.....

.....

.....

.....

.....

..... **[3]**

(c) The class design for `linkedList` is shown here.

class: `linkedList`
attributes: `private headNode : node`

methods:
`new()`
`insertNode()`
`outputList()`

`new()` is the constructor method. The constructor initialises the attribute `headNode` to a `null` value.

(i) Write the constructor method using pseudocode or program code.

.....

.....

.....

..... **[2]**

© OCR 2025

The class design for `node` and `linkedList` are shown again here for your reference.

class: node
attributes: private value : integer private nextNode : node
methods: new() getValue() getNextNode() setValue() setNextNode()

class: linkedList
attributes: private headNode : node
methods: new() insertNode() outputList()

(ii) The method `getValue()` will return the value of a node. The method `getNextNode()` will return the next node to be accessed in the linked list.

The method `outputList()` traverses the linked list starting at the node `headNode`. The method joins the data in each node and returns it.

Complete the method `outputList()` using pseudocode or program code.

```
function outputList()

   returnValue = ""

   start = ..............................................

   while start ..............................................  null

      returnValue = returnValue + start. ..............................................() + " "

      start = start. .............................................. ()

   endwhile

   return ..............................................

endfunction
```

[5]

© OCR 2025

(d) The method `insertNode()` will insert a new node into the linked list.

The method accepts an integer as a parameter, creates a new node with that data and inserts it into the linked list, e.g. `insertNode(8)` creates a node object with the data 8 and adds it to the linked list.

The program needs to store the following data items in the linked list in the order they are given:

10 25 39

The program then needs to output all of the data in the list using the appropriate method.

Write an algorithm using pseudocode or program code to:

- create a new `linkedList` object with the identifier `createList`
- insert each of the data items into the linked list (10, 25, 39)
- output the content of the linked list.

.....

.....

.....

.....

.....

.....

.....

.....

.....

..... **[5]**

© OCR 2025

(e) A new method is needed to remove a node from the linked list. The method needs to take the data to remove as a parameter and then find this data in the linked list before removing the node.

Describe the steps that this method will need to follow **and** the decisions that need to be made in the method to remove the correct node.

.....

.....

.....

.....

.....

.....

.....

.....

.....

..... **[5]**

(f) The program developer made use of computational methods when designing the object-oriented programming solution. One computational method is divide-and-conquer.

Identify **three** other computational methods that the program developer could have used.

1 .....

2 .....

3 .....

[3]

© OCR 2025

BLANK PAGE

DO NOT WRITE ON THIS PAGE

© OCR 2025

(g)* This program uses classes and objects to create a linked list.

An alternative method is creating a 2-dimensional array. In this design the first index stores the data and the second element stores the index of the next data item. Array elements not currently used are also linked together to form a chain of empty values to identify where the next data item can be inserted.

Compare the two different methods of creating a linked list that have been described.

You should include the following in your answer:

- the features of both methods including the reusability of program components
- benefits and drawbacks of both methods in this program
- a conclusion evaluating the efficiency of both methods when designing a program that needs a linked list. **[12]**

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

© OCR 2025

END OF QUESTION PAPER

© OCR 2025

EXTRA ANSWER SPACE

If you need extra space use this lined page. You must write the question numbers clearly in the margin.

OCR
Oxford Cambridge and RSA

Copyright Information

OCR is committed to seeking permission to reproduce all third-party content that it uses in its assessment materials. OCR has attempted to identify and contact all copyright holders whose work is used in this paper. To avoid the issue of disclosure of answer-related information to candidates, all copyright acknowledgements are reproduced in the OCR Copyright Acknowledgements Booklet. This is produced for each series of examinations and is freely available to download from our public website (www.ocr.org.uk) after the live examination series.

If OCR has unwittingly failed to correctly acknowledge or clear any third-party content in this assessment material, OCR will be happy to correct its mistake at the earliest possible opportunity.

For queries or further information please contact The OCR Copyright Team, The Triangle Building, Shaftesbury Road, Cambridge CB2 8EA.

OCR is part of Cambridge University Press & Assessment, which is itself a department of the University of Cambridge.

© OCR 2025