Qualification Accredited

Oxford Cambridge and RSA

GCSE (9-1)

*Examiners' report*

# COMPUTER SCIENCE

**J276**
For first teaching in 2016

## J276/02 Summer 2019 series

Version 1

www.ocr.org.uk/computerscience

# Contents

Introduction .....3
Paper 2 series overview .....4
Question 1 (a) (i) .....5
Question 1 (a) (ii) .....5
Question 1 (b) (i) .....6
Question 1 (b) (ii) .....7
Question 1 (b) (iii) .....8
Question 2 (a) .....9
Question 2 (b) .....10
Question 2 (c) .....12
Question 3 (a) (i) .....13
Question 3 (a) (ii) .....13
Question 3 (b) (i) .....14
Question 3 (b) (ii) .....14
Question 4 (a) (i) .....15
Question 4 (a) (ii) .....15
Question 4 (b) .....16
Question 4 (c) .....17
Question 5 (a) .....18
Question 5 (b) .....18
Question 5 (c) .....19
Question 5 (e) .....19
Question 6 (a) (i) .....20
Question 6 (a) (ii) .....22
Question 6 (b) (i) .....23
Question 6 (c) (i) .....23
Question 6 (d) (i) .....24
Question 6 (d) (ii) .....26
Question 6 (d) (iii) .....27
Question 6 (d) (iv) .....27
Question 6 (e) .....28

# Introduction

Our examiners' reports are produced to offer constructive feedback on candidates' performance in the examinations. They provide useful guidance for future candidates. The reports will include a general commentary on candidates' performance, identify technical aspects examined in the questions and highlight good performance and where performance could be improved. The reports will also explain aspects which caused difficulty and why the difficulties arose, whether through a lack of knowledge, poor examination technique, or any other identifiable and explainable reason.

Where overall performance on a question/question part was considered good, with no particular areas to highlight, these questions have not been included in the report. A full copy of the question paper can be downloaded from OCR.

# Paper 2 series overview

J276/02 (Computational thinking, algorithms and programming) is one of two examination components for the GCSE Computer Science. This component focuses on:

- algorithms
- programming techniques
- producing robust programs
- computational logic
- translators and facilities of languages
- data representation

To do well on this paper, candidates need to be comfortable with writing, completing and using algorithms using pseudocode and/or flowcharts. This may involve applying their knowledge to unfamiliar contexts.

Where candidates had extensive practice of producing and completing algorithms using a high-level language in classroom situations, this clearly allowed them to answer questions on this paper more confidently. The mandatory Programming Project task is one chance for all candidates to spend a significant amount of time practising their programming skills.

Centres are encouraged to be aware of the appendices at the rear of the specification, particularly section 5f which shows the format that pseudocode and Boolean logic will be presented in examinations.

Candidates' answers will be accepted in any logical form; they are not required to present answers in a specific form. However, candidates should be aware of these conventions for them to successfully understand and access examination questions.

***Candidate performance overview***

Candidates who did well on this paper generally did the following:

- understood how sound and image data are represented in computer systems and were able to show precision in their explanations
- read and understood the requirements for questions relating to algorithms, attempting to answer the whole question and cover each requirement
- demonstrated that they had received significant opportunities to practise programming skills in the classroom and were therefore able to apply their knowledge to unfamiliar contexts.

Candidates who did less well on this paper generally did the following:

- covered only part of longer questions, such as giving an answer to 6a(i) without covering the requirement for the answer to be a function with parameters passed
- demonstrated only basic understanding of how a bubble sort works, giving superficial answers to 6d(i) and 6d(ii)
- showed a lack of understanding of computing-related mathematics (such as exponentiation) and comparison operators (such as <, <=, > and >=).

## Note

As you are no doubt aware, Ofqual announced that programming skills for all GCSEs in Computer Science will be assessed exclusively by exam from **summer 2022**. Since the announcement, we've been working on how your updated GCSE (9-1) Computer Science could look for first teaching in September 2020. Read Ceredig's blog which highlights what we've been working on so far. You can sign-up for email updates to ensure you are receiving the latest concise and informative newsletters and emails.

## Question 1 (a) (i)

**1** **(a)** A radio station records an interview with a computer scientist using a computer and audio recording software.

**(i)** Explain how sampling is used to store audio recordings.

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

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

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

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

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

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

Candidates generally understood that sampling involves taking measurements at regular or set intervals, and that these measurements are then converted to and stored as binary.

Higher ability candidates were able to state that it is the height or amplitude of the waveform that is measured.

Some candidates were confused by this and incorrectly tried to explain sampling as a measurement of frequency.

## Question 1 (a) (ii)

A second interview with the computer scientist is recorded. Before this interview, the sampling frequency in the audio software is increased.

**(ii)** Define what is meant by the term **sampling frequency**.

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

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

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

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

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

Most candidates were able to identify sample frequency as how often a sample is taken. A slightly more precise answer was that it is the number of samples taken per time-period, such as 44,000 samples per second which can be given as 44KHz (1 Hertz = once per second).

## Question 1 (b) (i)

**(b)** The radio station uses a digital camera to take a photograph of the computer scientist for their website. The photograph is stored as a bitmap image.

**(i)** Describe how bitmap images are represented in binary.

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

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

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

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

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

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

Most candidates understood at least some of the mechanics behind the representation of bitmap images using binary, but many were unable to convey the precise details of this in their answer.

The basic idea of an image being represented by being split up into pixels was treated by many as a given, despite the word 'pixel' appearing nowhere in the question stem; candidates should be encouraged to state basic concepts such as this and not assume that they are obvious.

Answers relating to the data stored for each pixel were also relatively vague, with candidates mentioning colours but generally not hitting the required clarity that each pixel is given a binary code that represents a colour, and that each colour in the image is given a unique code.

A very common answer was that each pixel will be 0 for white and 1 for black, which shows a lack of understanding of coloured or even greyscale bitmap images.

## Question 1 (b) (ii)

**(ii)** Explain why computers represent data in binary form.

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

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

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

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

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

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

This question required candidates to explain why computers represent data of any form using binary.

Strong responses linked to the word 'why' in the question and explained that transistors (or equivalent) inside the computer have two states (on and off) which are represented by the 1 and 0 of binary.

Candidates who simply stated that binary is 1 and 0 did not therefore answer the question asked and received 0 marks.

Some candidates turned to colloquialisms such as 'because it is what the computer understands' which are not precise enough at GCSE Level.

## Question 1 (b) (iii)

The image is compressed using lossy compression before being uploaded to the radio station's web server. The image will be used on the radio station's website.

**(iii)** Describe **one** advantage and **one** disadvantage of using lossy compression on the image that will be used on the website.

Advantage ..........................................................................................................

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

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

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

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

Disadvantage ..........................................................................................................

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

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

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

..........................................................................................................
**[4]**

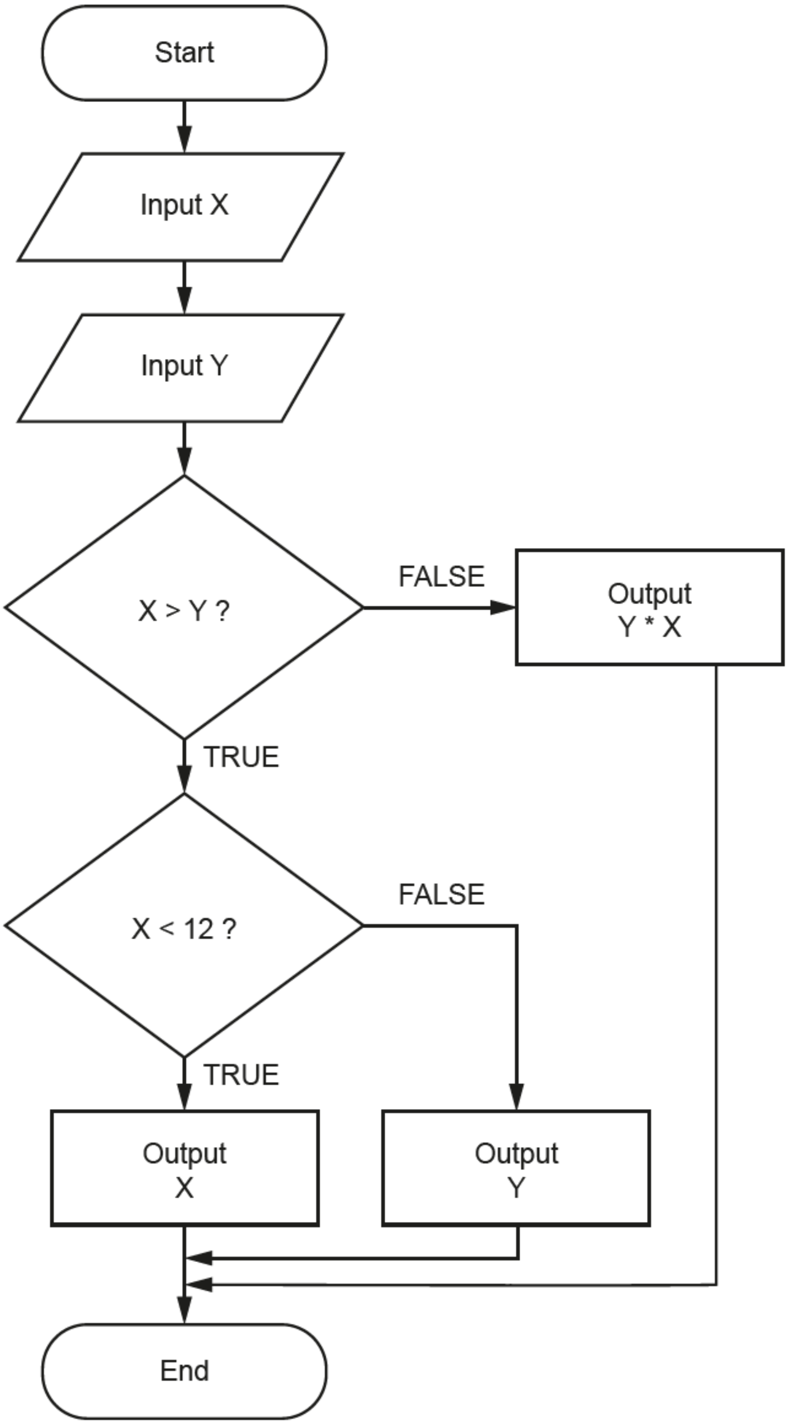
This question asked candidates to look at the advantages and disadvantages of using lossy compression when the image was used on a website.

Most candidates were able to give the advantage of this reducing the file size, but only higher ability candidates then went on to expand this in the context given and discussed the issues of space available on the web server or quicker upload/download speed.

Disadvantages were perhaps more confidently answered, with permanent loss of data and the associated loss of quality in the image not requiring the same level of contextualisation.

## Question 2 (a)

**2** A programmer creates an algorithm using a flow chart.



**(a)** Complete the table to give the output when each of the following set of values are input into the algorithm as X and Y.

| Input value of X | Input value of Y | Output |
|---|---|---|
| 15 | 10 | |
| 6 | 5 | |
| 2 | 3 | |
| 12 | 2 | |

**[4]**

This question required candidates to follow through the given algorithm and decide on the value that would be output given a set of input values.

Most candidates were able to complete this successfully for at least some values.

Where mistakes were made, these tended to be with the last set of values and the decision as to whether 12 is less than 12, which should be evaluated to be False.

Candidates gained a good understanding of this algorithm through this question which was then intended to lead onto the next question.

## Question 2 (b)

**(b)** Write this algorithm using pseudocode.

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

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

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

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

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

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

This question asked candidates to translate the given flowchart into pseudocode. It should be noted that no specific format for pseudocode is expected; candidates may use any code-like format that they choose as long as this conveys the logical intention of the solution.

Large numbers of candidates achieved full marks on this question and candidates who explored the whole of the algorithm were likely to do very well.

By comparison, candidates who attempted to convert each box of the flowchart into lines of pseudocode strictly from top to bottom often fell foul of the False outcome from the first decision being in the wrong place.

The most common issue here was a lack of understanding that the opposite of X>Y is not simply X<Y; this does not cater for X being equal to Y. Where candidates put X <=Y or equivalent, this was accepted.

This logical issue was also frequently repeated on the second decision for X<12.

## Exemplar 1

```
START
x = input ('Type a number')
y = input ('Type a number')
if x > y:
        if x < 12:
                output x
        else:
                output y
else:
    x = x ** y
    output x
```

In Exemplar 1, the candidate has logically hit every point from the flowchart and converted this successfully into pseudocode. This response achieved 6 marks.

## Question 2 (c)

The algorithm is written in a high-level language. The high level code must be translated into machine code before a computer processor can execute it.

**(c)** Describe **two** methods of translating high level code into machine code.

1 ..............................................................................................................................

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

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

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

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

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

2 ..............................................................................................................................

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

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

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

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

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

.............................................................................................................................. **[4]**

This question was answered well by large numbers of candidates who were able to list the two translators listed in the specification, compilers and interpreters.

Most candidates were then further able to describe the methods used by these translators, such as interpreters translating code line-by-line.

It is pleasing to see that this section of the specification is obviously well understood and generally answered with confidence.

## Question 3 (a) (i)

**3** Louise writes a program to work out if a number entered by the user is odd or even. Her first attempt at this program is shown.

```
01 num = input("enter a number")

02 if num MOD 2 >= 0 then

03      print("even")

04 else

05      pritn("odd")

06 endif
```

**(a)** The program contains a logic error on line **02**.

**(i)** State what is meant by a logic error.

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

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

This question asked candidates to state the meaning of the term 'logic error'.

To be successful, candidates needed to show examiners that they knew how a logic error differed from other types of errors and so answers such as 'when code does not work' were not precise enough.

Successful answers conveyed the idea of incorrect output without stopping the execution of the program.

## Question 3 (a) (ii)

**(ii)** Give a corrected version of line **02** that fixes the logic error.

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

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

The logic error present in the program was that any number entered would print out 'even' because line 02 checked if `num MOD 2` was greater than or equal to 0 rather than simply equal to 0. This could be fixed by replacing the greater than comparison with an = (or == ).

Few candidates achieved this, suggesting a lack of confidence either with the MOD operator or the program as a whole.

## Question 3 (b) (i)

**(b)** The program contains a syntax error on line **05**.

**(i)** State what is meant by a syntax error.

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

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

This question asked candidates to state the meaning of the term 'syntax error'.

To be successful, candidates needed to show examiners that they knew how a syntax error differed from other types of errors and so answers such as 'when code does not work' were not precise enough.

A very common incorrect answer here was to explain what caused the syntax error in this code or to give general examples of things that would cause syntax errors (e.g. 'a misspelling of a word') and not to give the wider answer of an error that breaks the grammatical rules of the programming language.

### Exemplar 2

**(i)** State what is meant by a syntax error.

when the programme is broken because

~~the~~ the coding is incorrect. **[1]**

In Exemplar 2, for example, the response given by the candidate above could equally apply to logic errors and is therefore assessed to be too vague. This was given 0 marks.

| | | |
|---|---|---|
| | **AfL** | Candidates should make sure that they read questions carefully to ascertain whether a contextual basis is required in their answer.<br>'What is meant by a syntax error' is a very different question than 'what caused the syntax error in this code'.<br>Examiners mark against a given mark scheme and even when candidates show understanding, this can only be credited if the question on the paper has been answered. |

## Question 3 (b) (ii)

**(ii)** Give a corrected version of line **05** that fixes the syntax error.

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

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

This question was answered extremely well as the majority of candidates were able to identify and fix the misspelling of the keyword 'print' in their answer.

The small number of candidates who did not manage to successfully answer this question tended to try to expand their answer beyond simply fixing the error. They then introduced new errors or unnecessary code. Where additional code was given, if it logically worked then this was accepted.

## Question 4 (a) (i)

**4** Elliott plays football for OCR FC. He wants to create a program to store the results of each football match they play and the names of the goal scorers. Elliott wants individual players from the team to be able to submit this information.

**(a) (i)** Define what is meant by **abstraction**.

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

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

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

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

## Question 4 (a) (ii)

**(ii)** Give **one** example of how abstraction could be used when developing this program.

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

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

As a topic new to the J276 specification, abstraction as a concept is obviously well understood by candidates who were generally able to give a definition for 4a(i) around removing unnecessary detail and focussing on the important details in a problem.

A small number then went on to state that this is done to simplify a problem or reduce complexity which shows a perhaps deeper understanding.

Answers for 4a(ii) were slightly less well completed as answers had to be linked to the scenario given; examiners were generous in their interpretation and accepted a wide range of answers that could potentially be removed or focused on.

## Question 4 (b)

**(b)** Describe **two** examples of defensive design that should be considered when developing this program.

1 ..........................................................................................................................

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

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

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

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

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

2 ..........................................................................................................................

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

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

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

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

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

.................................................................................................................... **[4]**

This question asked candidates to describe two examples of defensive design that should be considered when developing the program for OCR FC.

It was clear that large numbers of candidates did not understand the term 'defensive design' and gave wide ranging answers that linked topics as wide as decomposition to defensive tactics in football.

Where this topic was understood, the most common answers given were those listed in the specification of input sanitisation, validation and authentication.

| | | |
|---|---|---|
| ? | **Misconception** | Defensive design is listed in the specification under section 2.3 'Producing robust programs'.<br>It is the deliberate act of anticipating how users could misuse a system, either unintentionally or maliciously and then attempting to reduce these opportunities.<br>Examples of defensive design in Computer Science include authentication (e.g. by asking for a username and password before allowing access to a system and input sanitisation (removing unexpected characters from user input). |

## Question 4 (c)

The number of goals scored in each football match is held in an array called `goals`. An example of this array is shown.

```
goals = [0, 1, 3, 0, 4, 5, 2, 0, 2, 1]
```

Elliott wants to count how many matches end with 0 goals.

**(c)** Complete the following pseudocode for an algorithm to count up how many matches with 0 goals are stored in the array and then print out this value.

```
nogoalscount = 0
for count = 0 to (goals.length-1)
    if goals[..............................] == 0 then
        nogoalscount ...........................................................
     endif
next count
print(....................................................)
```

**[3]**

This question assessed whether candidates were able to complete an algorithm to count the number of times that the value 0 appeared in a given array.

The first blank to be filled in tested candidates' understanding of the use of a for loop counter to access each index in an array sequentially; this was not answered successfully by many candidates and perhaps shows a lack of understanding.

The next blank was more successfully answered and required candidates to increment the variable by 1 which could have been done in several ways, all of which were accepted.

Finally, the simplest and most commonly correct answer simply required candidates to print out the value of the variable.

## Question 5 (a)

5 (a) Convert the hexadecimal number **A3** to denary. Show your working.

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

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

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

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

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

This question asked candidates to convert a two-digit hexadecimal number to denary.

Many answers were fully correct. Where mistakes were made, it was very common to see A being converted to 10 and then this added to the 3, giving 13; this obviously misses out the crucial step of multiplying 10 by 16.

Where other sensible methods were used, such as converting to binary first, this was credited although candidates should be able to complete conversions directly between hexadecimal and denary (and vice versa) without the need for the intermediary step.

## Question 5 (b)

(b) Convert the binary number **1011011** to denary. Show your working.

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

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

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

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

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

As in previous years, conversions to and from binary are now done extremely well by the majority of candidates.

This question used a 7-bit number. Candidates are expected to understand that this can be dealt with in many ways, including filling the unused $8^{th}$ bit on the left with 0 or counting each digit from the right as increasing powers of 2.

It was pleasing to see that so many candidates understood this and were able to give the correct answer of 91.

Where mistakes were made, this involved candidates either adding a 0 to the right of the least significant bit or assigning bit values from 128 on the left which means that the answer is doubled.

## Question 5 (c)

**(c)** The symbol **^** is used for exponentiation.

Give the result of `a^b` when `a = 3` and `b = 2`.

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

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

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

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

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

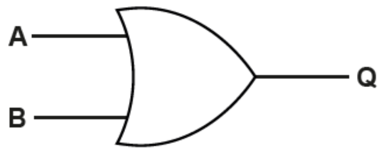
Exponentiation is covered in the specification in section 2.4, Computational logic under the subheading 'applying computing-related mathematics'.

Approximately half of all candidates did not understand the use of this symbol, even when the meaning was given in the question.

The most common incorrect answer was to treat exponentiation as synonymous to multiplication.

## Question 5 (e)

**(e)** Complete the truth table for the following logic gate.




| A | B | Q |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| | 0 | |
| 1 | | |

**[4]**

This question asked candidates to complete a truth table for an OR logic gate, with the correct answer being all 1s.

This relatively simple question firstly assessed whether candidates understood the use of truth tables in general, with half the marks being given for the inputs, regardless of the gate.

Secondly, this question assessed whether candidates understood the outputs given by an OR gate with these inputs.

As expected, the majority of candidates were able to gain full marks for both of these areas.

## Question 6 (a) (i)

**6** OCR Land is a theme park aimed at children and adults. Entrance tickets are sold online. An adult ticket to OCR Land costs £19.99, with a child ticket costing £8.99. A booking fee of £2.50 is added to all orders.

**(a)** A function, `ticketprice()`, takes the number of adult tickets and the number of child tickets as parameters. It calculates and returns the total price to be paid.

**(i)** Use pseudocode to create an algorithm for the function `ticketprice()`.

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

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

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

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

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

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

This question assessed not only whether candidates could create an algorithm to calculate a ticket price using the information given, but also whether this could be done as a function.

Many candidates were able to achieve the first part of this well, but only a small number even attempted to create this in the form of a function with parameters passed in.

Most candidates chose to ask for inputs during the algorithm instead of passing parameters and were therefore only able to achieve a maximum of 3 marks out of 6.

| | | |
|---|---|---|
| ? | **Misconception** | A function should not take input directly from the user; instead, values required should be passed in as parameters.<br>Equally, when the required calculations are completed the value should not be printed out but returned.<br>Using parameters and returned values in this way makes a function truly reusable. |

## Exemplar 3

```
FUNCTION ticketprice(adulttickets, childtickets)
    ~~adultprice =~~
    ~~fee = 2.50~~
    FLOAT fee = 2.50
    adultprice = adulttickets * 19.99
    childprice = childtickets * 8.99
    total = adultprice + childprice + fee
    RETURN total
ENDFUNCTION
```

Exemplar 3 shows a response that successfully covers all of the points required by the question and therefore achieves 6 marks.

## Exemplar 4

```
Input Number of Adult ()
Input Number Of children ()
Price = (number of adult * 19.99) + (Number of children *
8.99)
Booking Fee = 2.50
(Price + Booking fee) = Overall price
Print Overall price ( )
```

The response above shows a response that only managed to gain 2 marks, for the calculation of the total price for adults and the total price for children.

There is no attempt to use a function and the input/outputs are done through inputs and print statements rather than parameters passed and a value returned.

In addition, the booking fee was not correctly added to the overall price as the assignment statement is the wrong way around.

## Question 6 (a) (ii)

**(ii)** Tick (✓) **one** box to identify the data type of the value returned from the function `ticketprice()`, justifying your choice.

| **Data type of returned value** | Tick (✓) **one** box |
|---|---|
| Integer | |
| Real | |
| Boolean | |
| String | |

Justification ..........................................................................................................

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

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

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

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

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

The majority of candidates answered this well, with Real being chosen because of the potential need to use decimal places in the value returned as the main correct answer.

Where candidates had justified the use of a string (because the returned data would include currency symbols or other text) this was also credited.

## Question 6 (b) (i)

**(b)** OCR Land regularly emails discount codes to customers. Each discount code includes a check digit as the last character.

**(i)** Give **one** benefit of using a check digit for the discount code.

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

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

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

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

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

This question asked candidates to give a benefit of using a check digit for a discount code that had been emailed out to customers.

A check digit is used to make sure that data has been transmitted, received or entered correctly into a computer system and so answers around this point were expected.

Where candidates went further and said that this therefore meant that discount codes could not be falsified or made up, this was accepted as the same main idea.

A very common wrong answer was so that the discount code could not be used again or was somehow unique to each user, showing a lack of understanding as to the use of check digits in computer systems.

## Question 6 (c) (i)

**(c)** A list of valid discount codes is shown below.

[NIC12B, LOR11S, STU12M, VIC08E, KEI99M, WES56O, DAN34S]

**(i)** State **one** reason why a binary search would not be able to be used with this data.

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

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

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

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

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

This question required candidates to understand that one of the pre-requisites of a binary search is that the data must be in some form of order. Where this was understood, candidates were able to identify that the data presented was not in order and so could be searched using this algorithm.

Extremely common wrong answers included that only numeric data could be used or that data had to be in binary.

| | | |
|---|---|---|
| | **Misconception** | A binary search can be carried out on non-numeric data.<br>If the data is ordered in some way (e.g. alphabetically) then this algorithm will be able to efficiently find data. |

## Question 6 (d) (i)

**(d)** OCR Land keeps track of the size of queues on its rides by storing them in an array with the identifier `queuesize`. It uses the following bubble sort algorithm to put these queue sizes into ascending numerical order.

```
swaps = True
while swaps
      swaps = False
      for p = 0 to queuesize.length-2
            if queuesize[p] > queuesize[p+1] then
                  temp = queuesize[p]
                  queuesize[p] = queuesize[p+1]
                  queuesize[p+1] = temp
                  swaps = True
            endif
      next p
endwhile
```

**(i)** Explain the purpose of the Boolean variable `swaps` in this bubble sort algorithm.

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

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

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

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

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

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

This question presented candidates with an algorithm for a bubble sort and asked them to explain the purpose of the Boolean value `swaps`.

This proved to be a very tough question, with the majority unable to explain its use as a flag to store whether a swap had taken place during that pass, and then further as a condition to check whether to repeat lines 03 to 11.

Many candidates understood, perhaps through the name given to the variable, that it was involved in swapping something, but many said that the variable itself made the swap, which is logically incorrect.

Some candidates seemed to not understand the while loop on line 02.

| | | |
|---|---|---|
| ? | **Misconception** | A while loop, as its name suggests, repeats while a condition is True.<br>For example, `while x>5` will repeat while the condition `x>5` is evaluated to be True.<br>However, if the condition given is itself a Boolean value, then no evaluation is necessary, and the True of False nature of this condition is used to decide whether to repeat or not.<br>In this question, `swaps` **is** a Boolean value.<br>The line `while swaps` is logically equivalent to the more verbose and unnecessary `while swaps == True` |

## Exemplar 5

To decide is there are values in the array need to be swapped or not. is it is 'True' a swap is required, is 'False' no swap is required

The above response shows a typical answer where the candidate has not quite understood the process undertaken by the bubble sort algorithm; swaps is set to True when a swap has occurred and not, as this candidate suggests, that the swap occurs because of the variable being set.

## Question 6 (d) (ii)

**(ii)** Explain the purpose of lines **06** to **08** in this bubble sort algorithm.

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

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

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

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

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

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

This was another question where candidates struggled to explain clearly the purpose of a specific part of the given algorithm.

Lines 06 to 08 dealt with swapping two values over. Simply identifying that swap takes place and which numbers are swapped was sufficient to achieve both marks here, but the majority were unable to successfully do this.

A mark could also have been given for explaining that the two numbers were not directly swapped over, but that a temporary value was used as an intermediary.

Many candidates provided answers that were entirely different from this. In some cases, candidates attempted to simply translate each line into structured English.

***Sorting and searching algorithms***

Candidates are required to understand the step by step process taken for three sorting algorithms (bubble sort, insertion sort, merge sort) and two searching algorithms (binary search, linear search).

Candidates are **not** expected to be able to write these algorithms fully in pseudocode in an exam situation, but it is expected that they are aware of the steps needed and the sequence that these steps should be taken in.

Question 6(d) attempted to assess this understanding in a slightly more technical way than in previous papers and students generally struggled to do this.

Centres should be encouraged to investigate sorting and searching algorithms in a manner that is perhaps more closely linked to their implementation in a high-level language in the classroom.

## Question 6 (d) (iii)

**(iii)** Describe **one** way that the maintainability of this algorithm could be improved.

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

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

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

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

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

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

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

Maintainability is obviously well understood by centres and students, with many students giving very pleasing answers to this question.

Commenting and the need for comments was perhaps the most popular answer.

It was nice to see modularisation discussed also, although a number of students were unable to explain what they meant by this and how it could be applied to the code given.

## Question 6 (d) (iv)

**(iv)** Give the names of **two** other sorting algorithms that could be used instead of bubble sort.

1 .......................................................................................................................................

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

2 .......................................................................................................................................

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

**[2]**

This question was answered extremely well by candidates and shows that the prescribed sorting algorithms are covered in centres and their names retained well by candidates.

Merge sort and insertion sort are the two other algorithms covered in the specification and these were the two most popular answers given by candidates. However, credit was also given to many other valid sorting algorithms and it is pleasing to see that centres or candidates are interested in going beyond the confines of the specification to investigate algorithms such as Quick sort (which appears in the A Level specification) and even Bogo sort. This additional learning is to be applauded.

## Question 6 (e)

**(e)** One ride in OCR Land has a minimum height of 140 cm to ride alone or 120 cm to ride with an adult.

Create an algorithm that:

- asks the user to input the height of the rider, in centimetres
- if needed, asks if they are riding with an adult
- outputs whether or not they are allowed to ride
- repeats this process until 8 people have been allowed to ride.

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

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

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

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

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

...................................................................................................................................... **[8]**

This question asked candidates to create an algorithm for checking the heights of riders on a theme park ride.

The algorithm was decomposed for candidates using bullet points and many candidates had a solid attempt at completing this.

Most candidates were able to ask for the input of a rider's height, although some candidates struggled to use (or missed out) either the assignment operator or the INPUT (or equivalent) command. A line of pseudocode that simply printed out 'enter a height' is not the same logically as an input and was not credited.

Where an input was taken and comparisons made, these were generally well attempted.

Examiners worked logically through the code and credit was given for checking for the three types of riders (over 140 and so can ride alone, between 120-140 and so need to ride with an adult, under 120 and so cannot ride) and giving the right output to each rider.

There were many different valid and invalid attempts at this but logically, if each type of rider would have produced the right output then this should have been credited. The mark scheme provides an exemplar answer, but a wide range of responses were accepted.

Many candidates did not attempt to repeat the algorithm until 8 riders had been allowed to ride and so were unable to access these marks. However, several higher ability candidates thought even more deeply about this and wrote algorithms that added two riders if someone was riding with an adult; this was a valid thought and so was credited as well.

| ***Acceptable methods of asking for input in pseudocode*** | ***Unacceptable methods of asking for input in pseudocode*** |
|---|---|
| • `INPUT height`<br>• `height = input`<br>• `height = input('enter your height')` | • `height = 'input'` (this assigns the string value 'input' to the variable')<br>• `input = height` (this has the assignment the wrong way around)<br>• `print 'enter your height'` (this is not input) |

## Exemplar 6

```
allowed = 0
~~Wh~~
~~DO~~
WHILE allowed != 8
    ~~user = INP~~
    height = INPUT("Input height in centimetres")
    IF height > 140 THEN
        PRINT("Allowed")
        allowed += 1
    ~~ELSE~~
    ELIF height >120 THEN
        adult = INPUT("Are you with an adult? Y/N")
        IF adult == "Y" THEN
            PRINT ("Allowed")
            allowed += 1
        ELIF adult == "N" THEN
            PRINT("Not allowed")
        ENDIF
    ELSE
        PRINT ("Not allowed")
    ENDIF
ENDWHILE
```

Exemplar 6 shows a well organised response that gained full marks. Each check is completed in order, with ELIFs and ELSE used efficiently to make sure that logically correct comparisons are made. The loop works correctly and repeats until 8 riders have been allowed to ride. This is a very good example of the sort of response needed to achieve highly.

# Supporting you

For further details of this qualification please visit the subject webpage.

## Review of results

If any of your students' results are not as expected, you may wish to consider one of our review of results services. For full information about the options available visit the OCR website. If university places are at stake you may wish to consider priority service 2 reviews of marking which have an earlier deadline to ensure your reviews are processed in time for university applications.

activeresults

***Review students' exam performance with our free online results analysis tool. Available for GCSE, A Level and Cambridge Nationals.***

It allows you to:

- review and run analysis reports on exam performance
- analyse results at question and/or topic level*
- compare your centre with OCR national averages
- identify trends across the centre
- facilitate effective planning and delivery of courses
- identify areas of the curriculum where students excel or struggle
- help pinpoint strengths and weaknesses of students and teaching departments.

*To find out which reports are available for a specific subject, please visit ocr.org.uk/administration/support-and-tools/active-results/

Find out more at ocr.org.uk/activeresults

# CPD Training

Attend one of our popular CPD courses to hear exam feedback directly from a senior assessor or drop in to an online Q&A session.

Please find details for all our courses on the relevant subject page on our website.

www.ocr.org.uk

The small print

**OCR Resources:** *the small print*

OCR's resources are provided to support the delivery of OCR qualifications, but in no way constitute an endorsed teaching method that is required by OCR. Whilst every effort is made to ensure the accuracy of the content, OCR cannot be held responsible for any errors or omissions within these resources. We update our resources on a regular basis, so please check the OCR website to ensure you have the most up to date version.

This resource may be freely copied and distributed, as long as the OCR logo and this small print remain intact and OCR is acknowledged as the originator of this work.

Our documents are updated over time. Whilst every effort is made to check all documents, there may be contradictions between published support and the specification, therefore please use the information on the latest specification at all times. Where changes are made to specifications these will be indicated within the document, there will be a new version number indicated, and a summary of the changes. If you do notice a discrepancy between the specification and a resource please contact us at: resources.feedback@ocr.org.uk.

Whether you already offer OCR qualifications, are new to OCR, or are considering switching from your current provider/awarding organisation, you can request more information by completing the Expression of Interest form which can be found here: www.ocr.org.uk/expression-of-interest

Please get in touch if you want to discuss the accessibility of resources we offer to support delivery of our qualifications: resources.feedback@ocr.org.uk

## Looking for a resource?

There is now a quick and easy search tool to help find **free** resources for your qualification:

www.ocr.org.uk/i-want-to/find-resources/


**www.ocr.org.uk**

OCR Customer Support Centre

**General qualifications**

Telephone 01223 553998
Facsimile 01223 552627
Email general.qualifications@ocr.org.uk