Delegate Booklet and Student Responses

Course Title:

Edexcel GCSE (9-1) Computer Science: Preparing for First Assessment/Mocks Marking Training

About this event

Course Title: Edexcel GCSE (9-1) Computer Science: Preparing for first assessment/mocks marking training

Course Code: <TBD>

Aims and Objectives of the event

- Understand the structure of the new qualification
- Look at different questions types and understand the requirements and how to mark questions.
- Mark student responses and discuss the exemplars
- Address common issues and frequently asked questions

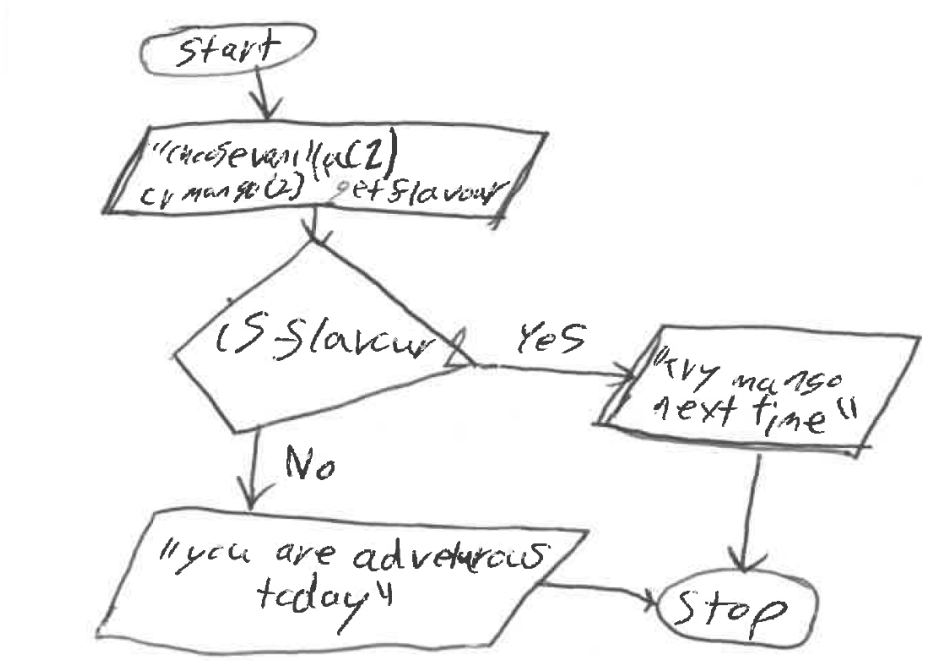
Agenda

Time	Item
16:00	Welcome
16:05	Introduction to the specification
16:10	Paper 1 - Introduction
16:20	Paper 1 - Marking
16:50	Paper 2 - Introduction
17:00	Paper 2 - Marking (Q01-Q03)
17:20	Paper 2 - Marking (Q04-Q06)
17:50	Support
17:55	Final questions
18:00	Finish

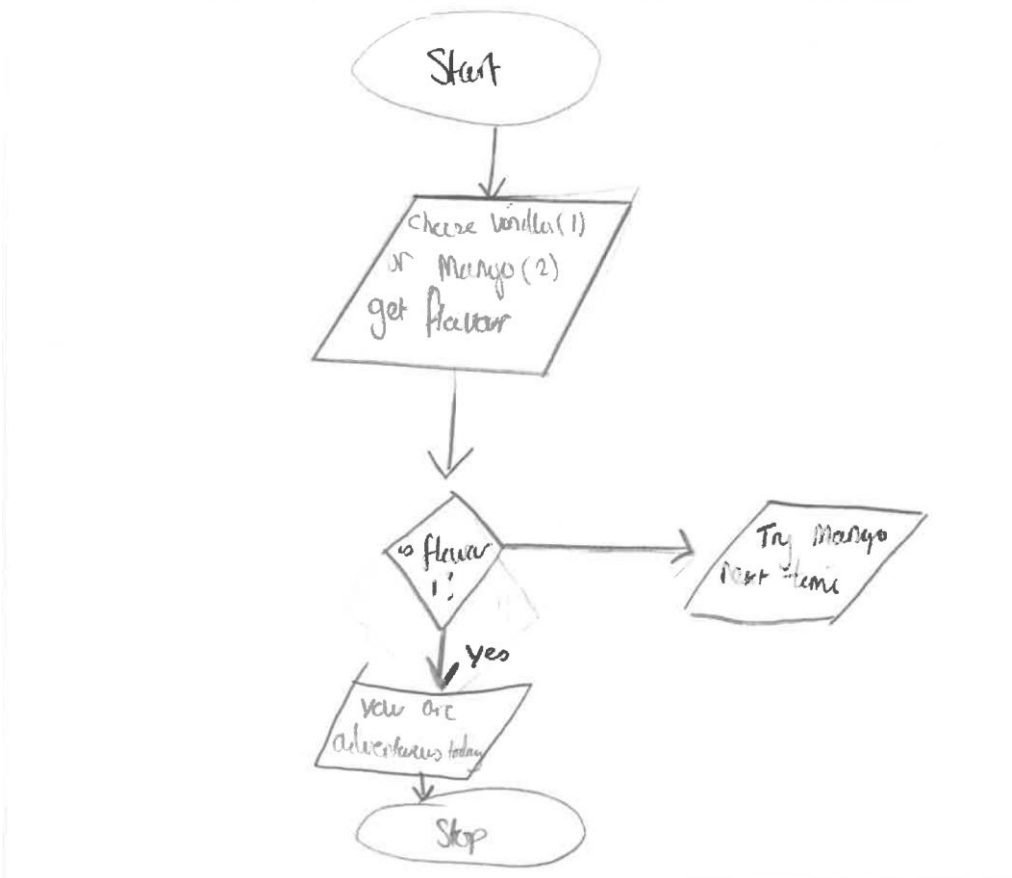
Paper 1 - student responses

Q01d

Response A



Response B



Q03g

Response A

1 High level languages are programmer friendly whereas low level languages are difficult and time consuming to work out.

2 High level languages will run on any form device with different CPUs but low level languages are Machine Specific so Change depending on the Computer.

Response B

(g) Describe **two** ways that a high-level language differs from a low-level language. **(4)**

1 High-level languages are closer to human languages and easier to understand whereas low-level languages ~~and~~ are difficult and require knowledge of it to understand.

2 ~~Ha~~ High-level languages have tools to debug the code whereas low-level languages do not.

Q04d

Response A

(6)

Access doors should be installs Access doors should be in place, in order to prevent unauthorised access to server rooms (could be fitted with facial recognition software).

Software should only be made accessible to authorised personal by installing a multifactor authentication. Access control can help decide who logs onto the network of the movie streaming company, and limit who has access to their files and who has permission to alter them.

A firewall could also be installed, in order to prevent communications from entering the ~~[illegible]~~ network without permission and also (to prevent) programs and users from accessing the internet from within the network without permission.

Response B

A way a streaming company could protect its network is first by using physical security. They can do this by using keycards to access the server room, locks, one way entrances etc. This is important as it makes sure no one can destroy or steal their networks.

Access control also does this as it stops certain people from the work place entering the server room. This allows for specific people to access these rooms e.g tech support, engineers etc.

Firewalls will for digital protection as it filters anything from outside the internal network to see if it has access.

Q05a

Response A

(a) Describe **one** reason that lossless compression is preferred to lossy compression for transmitting word-processed documents over a network.

(2)

lossy compression removes unwanted data from the file whereas lossless doesn't meaning no words or letters will be removed from the document.

Response B

5 Data

(a) Describe **one** reason that lossless compression is preferred to lossy compression for transmitting word-processed documents over a network.

(2)

Lossless compression reduces the filesize without any data being deleted whereas lossy compression deletes data when reducing the file size.

Q05c

Response A

(c) Explain the reason why at least nine bits are needed to store 300 different binary patterns.

(2)

number of patterns stored = 2^{bits}

$2^9 = 512$

$2^8 = 250$ 8 bits is too small

Response B

(c) Explain the reason why at least nine bits are needed to store 300 different binary patterns.

(2)

because the number of bits is the power from two e.g. $2^8 = 256$ so $2^9 = 300$.

Paper 2 – student responses

Q01

Supplied student file

```
# ---------------------------------------------------------------------
# Import libraries
# ---------------------------------------------------------------------
# =====> Import the math library


# ---------------------------------------------------------------------
# Global variables
# ---------------------------------------------------------------------
# =====> Create an integer variable named radius and set it to 0


# =====> Create a real variable named circumference and set it to 0.0


# ---------------------------------------------------------------------
# Main program
# ---------------------------------------------------------------------
# =====> Complete the line to assign an integer, input by
#        the user, to the variable radius
radius =

# =====> Complete the code in the brackets to check for
#        an invalid radius of zero or less input by the user
if (radius           ):
    # =====> Add a line to tell the user the entry is invalid

else:
    # =====> Complete the calculation of the circumference
    circumference =

    # =====> Complete the line to round circumference to three
    #        decimal places using the round() function
    circumference =

    print ("The circumference is", circumference)
```

Student response

```
# ---------------------------------------------------------------------
# Import libraries
# ---------------------------------------------------------------------
# =====> Import the math library
import math
# ---------------------------------------------------------------------
# Global variables
# ---------------------------------------------------------------------
# =====> Create an integer variable named radius and set it to 0
radius= 0

# =====> Create a real variable named circumference and set it to 0.0
circumference= 0.0

# ---------------------------------------------------------------------
# Main program
# ---------------------------------------------------------------------
# =====> Complete the line to assign an integer, input by
#        the user, to the variable radius
radius = input("what would you like the radius to be?")

# =====> Complete the code in the brackets to check for
#        an invalid radius of zero or less input by the user
if (radius < circumference):
    print ("invalid entry")
    # =====> Add a line to tell the user the entry is invalid

else: print ("Calculating logical operation")
    # =====> Complete the calculation of the circumference
circumference = (math.pi * (radius*2))##0

    # =====> Complete the line to round circumference to three
    #        decimal places using the round() function
    circumference =

    print ("The circumference is", circumference)
```

Q02

Supplied student file

```
# -----------------------------------------------------------------------
# Global variables
# -----------------------------------------------------------------------
a = "Y"
initials =

# -----------------------------------------------------------------------
# Main program
# -----------------------------------------------------------------------

while ((a == "Y") and (a == "y")):
    initials = input ("Enter your initials, without spaces: ")
    if (initials.isalpha ()):
        print ("Must be alphabetic characters")
    elif (len (initials) <= 2):
        print ("Not long enough")
    else if (len (initials) > 3):
        print ("Too long")
    else
        initials = initials.lower ()
        print ("Your initials are:", initials)

    a == input ("Would you like to go again? ")
```

Student response

```
# -----------------------------------------------------------------------
# Global variables
# -----------------------------------------------------------------------
a = "Y"
nickname = 0

# -----------------------------------------------------------------------
# Main program
# -----------------------------------------------------------------------

while ((a == "Y") or (a == "y")):
    nickname = input("Enter your initials, without spaces: ")
    if nickname.isalpha ():
        print("Must be alphabetic characters")
    elif (len(nickname) <= 2):
        print("Not long enough")
    if (len(nickname) > 3):
        print("Too long")
    else:
        nickname = nickname.upper()
        print("Your nickname is:", nickname)  # this prints the nickname

    a = input("Would you like to go again? ")
```

Q03

Supplied student file

```
# ------------------------------------------------------------------
# Global variables
# ------------------------------------------------------------------
data = ""

# =====> Create a one-dimensional data structure holding the
#        five vowels in upper case
theVowels =

# =====> Create a one-dimensional data structure to hold the count for each vowel
theCounts =

# ------------------------------------------------------------------
# Subprograms
# ------------------------------------------------------------------
def displayHistogram (inSymbols, inNumbers):
    # =====> Complete the call to range (), using len ()
    for index in range (                    ):
        # Repeat the symbol for the number of times required
        # =====> Complete the print statement to print the vowel
        #        for the number of times it was counted
        print (                           )

# ------------------------------------------------------------------
# Main program
# ------------------------------------------------------------------
# User types in a string
data = input ("Enter a string: ")

# =====> Complete the line to convert data to uppercase
data =

# Count each vowel in the input
for letter in data:
    # =====> Use selection to check for each vowel and increment the corresponding count


# Print a horizontal histogram
# =====> Complete the call to the user-devised subprogram
displayHistogram (                    )
```

Student response

```
# -----------------------------------------------------------------
# Global variables
# -----------------------------------------------------------------
data = ""

# =====> Create a one-dimensional data structure holding the
#        five vowels in upper case

theVowels = ["A", "E", "I", "O", "U"]

# =====> Create a one-dimensional data structure to hold the count for each vowel

theCounts = [0,0,0,0,0]

# -----------------------------------------------------------------
# Subprograms
# -----------------------------------------------------------------

def displayHistogram (inSymbols, inNumbers):

    # =====> Complete the call to range (), using len ()

    for index in range (len(inSymbols[0])):

        # Repeat the symbol for the number of times required
        # =====> Complete the print statement to print the vowel
        #        for the number of times it was counted

        print (inNumbers, inSymbols)

# -----------------------------------------------------------------
# Main program
# -----------------------------------------------------------------
# User types in a string

data = input ("Enter a string: ")

# =====> Complete the line to convert data to uppercase

data = data.upper()

# Count each vowel in the input
for letter in data:
    if letter == theVowels[0]:#if the letter A is found it adds 1 to A
        theCounts[0] + 1

    elif letter == theVowels[0]:#if the letter E is found it adds 1 to E
        theCounts[1] + 1

    elif letter == theVowels[0]:#if the letter I is found it adds 1 to I
        theCounts[2] + 1

    elif letter == theVowels[0]:#if the letter O is found it adds 1 to O
        theCounts[3] + 1

    elif letter == theVowels[0]:#if the letter U is found it adds 1 to U
        theCounts[4] + 1

    else:
        print("Not a vowel")
    # =====> Use selection to check for each vowel and increment the corresponding count


# Print a horizontal histogram
# =====> Complete the call to the user-devised subprogram

displayHistogram (theVowels,theCounts)

print (data)
```

Q04

Supplied student file

```
# ----------------------------------------------------------------
# Global variables
# ----------------------------------------------------------------
# ----------------------------------------------------------------
# Main program
# ----------------------------------------------------------------
# Get the user's input
# Multiply each digit by its position in the number
# Find the remainder from integer division by 11
# Calculate the check digit
# Test for boundaries


elif (checkDigit == 10):
isbn = input ("Enter a 9-digit ISBN number: ")
if (checkDigit == 11):
checkDigit = 11 - modulus
modulus = total % 11
print ("Check digit = " + strCheckDigit + " ISBN = " + isbn + strCheckDigit)
while (index < len (isbn)):
else:

    total = product + total
    strCheckDigit = "0"
    product = int (isbn[index]) * multiplier
    strCheckDigit = "X"
    strCheckDigit = str (checkDigit)
    multiplier = multiplier - 1
    index = index + 1

product = 0
checkDigit = 0
strCheckDigit = ""
total = 0
multiplier = 10
modulus = 0
isbn = ""
index = 0
```

Student response

```
# ----------------------------------------------------------------
# Global variables
# ----------------------------------------------------------------
isbn = ""
index = 0
product = 0
total = 0
multiplier = 10
modulus = 0
checkDigit = 0
strCheckDigit = ""

# ----------------------------------------------------------------
# Main program
# ----------------------------------------------------------------
# Get the user's input
Box_ID = int(input("Enter a 6-digit Box ID number: "))

# Multiply each digit by its position in the number
while (index < len (Box_ID)):
    product = Box_ID[index] * multiplier
    total = product + total
    index = index + 1
    multiplier = multiplier - 1

# Find the remainder from integer division by 11
modulus = total % 11

# Calculate the check digit
checkDigit = 11 - modulus

# Test for boundaries
if (checkDigit == 11):
    strCheckDigit = "0"
elif (checkDigit == 10):
    strCheckDigit = "X"
else:
    strCheckDigit = str (checkDigit)

print ("Check digit = " + strCheckDigit + " ISBN = " + isbn + strCheckDigit)
```

Q05

Supplied student file

```
# ------------------------------------------------------------------
# Constants
# ------------------------------------------------------------------
FILE_NAME = "Q05_Data.txt"      # The input data file
NUMS_PER_LINE = 10              # Number of items per line in the file

# ------------------------------------------------------------------
# Subprograms
# ------------------------------------------------------------------
def processLines (inFile):
    # =====> Write your code here


    # Get the line of numbers from the file
    # =====> Write your code here


        # Calculate the mean for the items and adjust the row counter
        # =====> Write your code here


        # Display the information in columnar format
        # =====> Write your code here


    # =====> Write your code here
    # Close the file


def displayTableHeaders ():
    # =====> Write your code here


# ------------------------------------------------------------------
# Main program
# ------------------------------------------------------------------
# Do the processing and the display
displayTableHeaders ()
processLines (FILE_NAME)
```

Student response

```
# ------------------------------------------------------------------
# Constants
# ------------------------------------------------------------------
FILE_NAME = "Q05_Data.txt"      # The input data file
NUMS_PER_LINE = 10              # Number of items per line in the file

# ------------------------------------------------------------------
# Subprograms
# ------------------------------------------------------------------
def processLines (inFile):
    # =====> Write your code here
    open_file = open("Q05_Data.txt","r")
    meancalc=0
    meancalc2=0
    row=1
    layout = "{:<5},{:>6}"#sets layout for table
    # Get the line of numbers from the file
    # =====> Write your code here
    for line in open_file:

        for index in line:
            if index != "," and index != "\n":
                index=int(index)

                # Calculate the mean for the items and adjust the row counter
                # =====> Write your code here
                meancalc=index+meancalc#adds everything together for the mean
                # Display the information in columnar format
                # =====> Write your code here
                index=index+1
                index=str(index)
            else:
                i=1+1
        meancalc2=meancalc/10
        print(layout.format(row,meancalc2)) #prints off table
        row=row+1
    open_file.close()



    # =====> Write your code here
    # Close the file


def displayTableHeaders ():
    # =====> Write your code here
    layout = "{:<5},{:>6}"#sets format for table
    print(layout.format("Row","Mean"))
    print("_ "*10)

# ------------------------------------------------------------------
# Main program
# ------------------------------------------------------------------
# Do the processing and the display
displayTableHeaders ()
processLines (FILE_NAME)
```

Q06

Supplied student file

```
# ---------------------------------------------------------------------
# Global variables
# ---------------------------------------------------------------------
theArtists = [["Andy", "Warhol", 1928],
              ["Pablo", "Picasso", 1881],
              ["Salvador", "Dali", 1904],
              ["Lavinia", "Fontana", 1552],
              ["Jackson", "Pollock", 1912],
              ["Henri", "Matisse", 1869],
              ["Frida", "Kahlo", 1907],
              ["Georgia", "O'Keeffe", 1887],
              ["Kara", "Walker", 1969],
              ["Yayoi", "Kusama", 1929]]

theLabels = []    # Put the new user labels into this structure
# ====> Write your code here

# ---------------------------------------------------------------------
# Main program
# ---------------------------------------------------------------------
# ====> Write your code here
```

Student response

```
# -----------------------------------------------------------------
# Global variables
# -----------------------------------------------------------------
theArtists = [["Andy", "Warhol", 1928],
              ["Pablo", "Picasso", 1881],
              ["Salvador", "Dali", 1904],
              ["Lavinia", "Fontana", 1552],
              ["Jackson", "Pollock", 1912],
              ["Henri", "Matisse", 1869],
              ["Frida", "Kahlo", 1907],
              ["Georgia", "O'Keeffe", 1887],
              ["Kara", "Walker", 1969],
              ["Yayoi", "Kusama", 1929]]

theLabels = []   # Put the new user labels into this structure
# ====> Write your code here

youngYear = 0

# -----------------------------------------------------------------
# Main program
# -----------------------------------------------------------------
# ====> Write your code here

for artist in theArtists:      # Iterates each artist's record in the array th
    first = artist[1][0]       # Finds the first letter of the first name, sec
    second = artist[0][0]
    year = artist[2]

    if year > youngYear:       # Compares the year of the current record to th
        youngYear = year
        youngName = artist[0] + ' ' + artist[1]

    label = str(first) + str(second) + str(year)     # Formats the label

    theLabels.append(label)    # Adds the label to the array theLabels

for label in theLabels:        # Iterates through the labels in theLabels and
    print(label)

print("The youngest artist is:", youngName, "with year of birth", youngYear)
```

Paper 1 – commentaries

Q01d

Response A

In this response, the start and stop symbols are in the correct place.

The decision box clearly has only two outputs, each directed to an output message.

Yes/No labels have been included. Careful inspection, however, shows that the test in the decision box checks if the user choice is mango (number 2). Therefore, the message, based on 'yes', is not accurate.

The flowchart is fully connected, although it will produce an incorrect response, which is accounted for in the previous bullet. This bullet is still awarded even though there are two input arrows to the stop symbol.

Bullets 1, 2, and 4 are awarded for 3 out of the available 4 marks.

Response B

In this response, the start and stop symbols are in the correct place.

The decision box clearly has only two outputs, each directed to an output message.

Yes/No labels have been not been included correctly, as only the 'yes' route is provided.

The flowchart is not fully connected, because the mango output is left hanging.

Bullets 1 and 2 are awarded for 2 of the 4 marks.

A few tips to help learners with this type of question are:

- Put arrows on your connecting lines to show the algorithm flow
- Have exactly one input to each symbol
- Have exactly one output from each symbol, except the decision box, which has two

Q03g

Response A

The first description makes a clear comparison, using the linking word 'whereas'. However, the two sides of the comparison contain shallow descriptions where interpretation is left to the examiner.

The second description again makes a clear comparison, using the linking word 'but'. The first part of this description expresses the idea that high-level languages are usable across different CPUs. That earns one mark. It may have been improved by acknowledging that a translator needs to exist for that CPU. The second part of this description addresses portability for low-level languages. It identifies them as machine specific. That also earns a mark.

This response matches both parts of bullet three. Therefore, two marks are awarded.

Response B

The first response uses the 'whereas' connecting word to set up the required comparison. The first part of this response is the equivalent of stating that high-level languages are closer to English than low-level languages. That earns one mark. The second part of the response is less clear and effectively just states the opposite of the first part, so earns no marks.

The second response again uses the 'whereas' connection word to set up a comparison. In this case, both sides of the connector are opposites. This response is about the tools available for a language, rather than the language itself. According to the additional guidance, this would not be awarded, even if it were correct. The description implies that low-level languages do not have tools to help programmers, but they do, although they may be more limited than those available to a high-level language programmer.

Bullet 1, subpart 1 awarded for 1 mark.

A few tips to help learners with this type of question are:

- Set up comparisons with a connecting word
- Address both sides of the comparison with the same aspect from each side

Q04d

Response A

Physical methods are accurately identified as doors and facial recognition.

Access control to the network is addressed with multi-factor authentication, with further exemplification as limiting access to files with permission.

The description of the firewall is accurate, reflecting controlling the flow of communications in both directions.

There is some reference to the context.

All items from the stem have been accurately addressed.

This response most closely matches Level 2 of the levels-based mark scheme, therefore, 4 marks are awarded.

Response B

Physical security has been identified as using keycard locks to access the server room.

Access control, in the second paragraph, has been incorrectly interpreted as physically accessing. Therefore, leaving access control unaddressed.

The last paragraph provides a partial description of a firewall, but does not explain how it would be used in the context.

There is a little inclusion of the context.

This response most closely matches Level 1 of the levels-based mark scheme, therefore, 2 marks are awarded.

A few tips to help learners with this type of question are:

- Ensure each bullet in the stem, if given is addressed
- Use linking words (whereas, so, because) to provide explanations and descriptions
- Set the responses in the context given in the stem

Q05a

Response A

This response chooses to compare the aspects of data loss over the two compression methods.

The first part of the response identifies that data is removed. There is an assumption that this means permanently. That earns one mark.

There is use of a linking word, which is always a good strategy for ensuring a well-formed response.

In the second part of the response, it would not be sufficient to say only that 'lossless doesn't remove data', as that would be the opposite of the first statement. However, the response goes further to indicate how this would be evidenced in the result of the compression, as 'no words or letters will be removed'. That earns the second mark.

The exemplar response in the mark scheme is the reverse of the student response here, but as long as two aspects are compared, with clear descriptions, marks are awarded.

This response earns both of the available marks.

Response B

The first part of the responses identifies that with lossless compression, data is not lost. The second identifies that with lossy compression, data is lost. Between the parts, definitions for the words in the stem are provided. However, there is no reference to a description of why that makes lossless compression preferable for word-processed documents.

One mark is awarded for either the first or second part of the response, but both marks cannot be awarded.

A few tips to help learners with this type of question are:

- Use linking words (so, such as, in order to)
- Bring in the context from the stem, if one is given, to clarify your thinking

Q05c

Response A

This response identifies the number of available binary patterns with 9 bits as 512. That earns the first mark.

The second mark is awarded for identifying that the number of available binary patterns with 8 bits is 256.

The response could have stopped there. However, the addition of the first statement, giving the expression for calculating the number of bit patterns based on the number of bits, and the last statement, giving the relationship between the two numbers, makes it very clear that the student understands the need for at least 9 bits.

This response earns both of the available marks.

Response B

This response identifies the number of available binary patterns with 8 bits as 256. That earns one mark.

The second part of the response indicates that 2 to the power of 9 is 300, which is inaccurate, so cannot earn a mark.

The response earns 1 mark.

A tip to help learners with this type of question is:

- Use linking words (because, so that)

Paper 2 – commentaries

Q01

Marks, for this response, in the same order as the mark scheme are:

- 1 mark for importing the math library on line 5
- 1 mark for creating an integer variable, with the correct name, on line 10
- 1 mark for creating a real variable, with the correct name, on line 13
- 1 mark for using the input() function on line 20
- 0 marks for lack of conversion from string input to integer, on line 20
- 0 marks for negative input validation, line 24, as it will not work because radius is a string. Use of the variable 'circumference', because it is set to 0.0, would validate negative integers, but this demonstrates confusion about the use of consistent data types to suit the problem.
- 1 mark for the appropriate error message for the user on line 25
- 1 mark for the correct translation of the formula given in the QP, on line 30
- 0 marks for attempting rounding, on line 34
- 0 marks for parameters to round, on line 34

This response earns 6 of the 10 marks available.

A few tips to help learners with this type of question are:

- Comment out any lines you don't want to deal with yet, so you can check the functionality of what you're currently working on. Don't forget to come back to the commented lines, though.
- Attempt to fix the lines of code as shown in the comments. Even if you try, but don't get it perfect, you may do enough to get partial marks.

Q02

Marks, for this response, in the same order as the mark scheme are:

- 1 mark for fixing the syntax error by initialising a variable to an integer, rather than a string. This works because of the dynamic typing in Python, but may cause problems later in the code.
- 0 marks for removing the 'else' rather than changing the 'else if' to 'elif'. This changes the behaviour of the program to produce inaccurate output because the validation has been disabled.
- 1 mark for adding the colon to the end of the line.
- 1 mark for changing the logical operator 'and' to 'or'
- 0 marks as the length check (line 15) is not corrected
- 0 marks as the logical operator 'not' is not to correct the error
- 1 mark for replacing the relational operator with assignment.
- 1 mark as conversion to upper found (line 20)
- 0 marks as the variable name 'a' is not changed to more meaningful one
- 1 mark as a comment is added to line 21 to explain print

This response earns 6 of the 10 marks available.

A few tips to help learners with this type of question are:

- If you add lines of code, the line numbers will change, so you may need to look near the line number given in the question paper. The original line of code is also given, so you can check against it.
- Syntax errors should be shown by your editor or you can find them by simply running the code.

Q03

Points-based marks, for this response, in the same order as the mark scheme are:

- 1 mark as theVowels array is created correctly (line 9)
- 1 mark as theCounts array is initialised correctly to 0 (line 13)
- 1 mark as the length function indexes the array of symbols in the selection statement (line 49)
- 0 marks as the print statement (line 29) does not choose a symbol from the vowels.
- 0 marks as the print statement (line 29) does not repeat any symbols.
- 1 mark as a conversion to upper is found (line 40)
- 1 mark as selection is used (lines 44-60) to check if letter is a vowel
- 1 mark as the correct relational operator is used in selection
- 1 mark as the call to the subprogram has two arguments (line 67)
- 1 mark as the call to the subprogram has arguments in the correct order (line 67)

Levels-based marks for Functionality are based on these observations.

- The program translates and runs without crashing, given the input from the question paper. The program does not crash when presented with an empty string or any string terminated by a single carriage return and line feed. Although the counts appear to be updated, there is no reassignment, so the original values in theCounts are never updated.
- The component parts of the solution are provided and some of the requirements are met, particularly the internal behaviour. The program outputs are not as required. The program does not crash. It is robust within the constraints of the problem.
- 2 marks for Functionality

This response earns 9 of the 13 marks available.

A few tips to help learners with this type of question are:

- Read all of the code before starting to make amendments. Sometimes, you'll see the names of variables or logic that you need to know about to make some of the earlier amendments.
- Make one or two amendments at a time, then check your code translates and runs. That reduces the places you have to look for errors.

Q04

Marks, for this response, in the same order as the mark scheme are:

- 1 mark as white space has been used to separate the blocks of logic.
- 0 marks as input (line 17) is converted to integer, which will cause a crash; Assignment to a variable not used in the rest of the processing will cause a cascade of errors.
- 1 mark as modulus is calculated before checkDigit calculation (line 27 before line 30)
- 1 mark as repetition encompasses product calculation (line 21 indented under line 20)
- 1 mark awarded for order, although the original line of code has been changed (line 21 before line 23)
- 1 mark awarded for order, although the original line of code has been changed (line 21 before line 24)
- 1 mark as total (line 22) is calculated after product (line 21)
- 1 mark as multiplier is initialised before use (line 8)
- 1 mark as index is initialised before use (line 5)
- 1 mark as total is initialised before use (line 7)
- 0 marks as no output is produced as the program crashes with type error (line 20)
- 0 marks as no output is produced as the program crashes with type error (line 20)
- 0 marks as no output is produced as the program crashes with type error (line 20)
- 1 mark as comments match the code blocks.
- 1 mark as the last line (line 40) of the program displays a result, although inaccurate.

This response earns 11 of the 15 marks available.

A few tips to help learners with this type of question are:

- Comment out all the lines and work with just a few at a time, checking to make sure the code translates as you go.
- Do not change any of the lines of code, unless instructed to do so, otherwise the solution may not work at all.

Q05

Points-based marks, for this response, in the same order as the mark scheme are:

- 0 marks as the mean of each row is not calculated correctly,. The loop on line 22 does not move the value of index along in the way that is required, accounting for numbers greater than single digits.
- 1 mark for dealing with the carriage return by using a novel approach of looping through the line, ignoring commas and CR.
- 1 mark for dealing with separating the line values. Although not used split the approach deals with the commas.
- 1 mark as the file is correctly opened for read only on line 12.
- 0 marks as the file is not closed.
- 1 mark as only local variables are used.

Levels-based marks for Solution Design are based on these observations.

- A variable has been used to track the row number; For loops have been used throughout to ensure all numbers could be processed. Lines 32 and 33 serve no purpose in the logic; <string>.format() used for controlling column headers, but not the decimal places;
- The component parts are visible in the solution. Most of the logic is clear ana appropriate to the problem. The use of variables is mostly appropriate. The choice of programming constructs is mostly appropriate.
- 2 marks for Solution Design

Levels-based marks for Functionality are based on these observations.

- Translates and runs without error; Headers and separator fit for purpose; Each row has the correct sequence number and the columns of values are aligned to make reading easier; The calculations are not correct, so not fully functional.
- The parts are complete enough to provide a functional program that meets most of the stated requirements. The outputs accurately reflect the logic of the code, even though they are not accurate. They are informative. No outside input is needed. The solution is robust within the constraints of the problem.
- 2 marks for Functionality

This response earns 8 of the 12 marks available.

A few tips to help learners with this type of question are:

- Read all of the code and comments before starting to edit the code file, so that you understand the structure of the solution.
- If you see something you don't know how to do, you can always look in the Programming Language Subset for help.
- Be sure to address each line that instructs you to write code, because there will usually be marks associated with it.

Q06

Points-based marks, for this response, in the same order as the mark scheme are:

- 1 mark for use of two-dimension indexing on lines 26-27.
- 1 mark for conversion of integer to string on line 34.
- 1 mark for string concatenation used (line 34) to create artist's code
- 1 mark for appending new label to the array of labels (line 36)
- 1 mark for date (youngYear) initialised to 0, so that the first real year encountered will be used (line 18)
- 1 mark for tracking the name and year of the youngest artist (lines 31-32)

Levels-based marks for Solution Design are based on these observations.

- The problem has been decomposed; Solution uses for loop rather than while to ensure all artists are processed; Correct data type conversions are found throughout;
- The component parts can be seen clearly in the solution. The logic is clear and appropriate. Variables and data structures are used appropriately; Choice of language constructs fit the problem;
- 3 marks for Solution Design

Levels-based marks for Good Programming Practices are based on these observations.

- The constructs available in the Programming Language Subset have been used effectively. The code is laid out to aid readability; Variable names are meaningful; Comments have been provided to explain the logic; An alternative, efficient, approach to finding the youngest artist is used;
- All the rules of layout, variable names, and comments have been followed. The code is clear, readable, and the logic is documented.
- 3 marks for Good Programming Practices

Levels-based marks for Functionality are based on these observations.

- Outputs match requirements set in the question paper; Functions correctly for any number of artists;
- The program translates and executes without errors. The component parts are complete giving a functional program that fully meets the given requirements. Outputs are accurate, informative, and suitable for the audience. There is no outside input required. The solution is robust within the constraints of the problem.
- 3 marks for Functionality

This response earns 15 of the 15 marks available.

A few tips to help learners with this type of question are:

- Read all of the requirements in the question paper carefully to note any constraints, such as using an existing data structure, code, or algorithm approach.

- Also, just focus on the requirements and do not add any extra functionality.
- Test your program against the requirements from the question paper.
- Always use comments, white space, and layout to help readability, even if it's not listed in the requirements.