AQA

AS COMPUTER SCIENCE

Paper 1

Tuesday 19 May 2020 Morning Time allowed: 1 hour 45 minutes

Materials
For this paper you must have:
- a computer
- a printer
- appropriate software
- the Electronic Answer Document
- an electronic version and a hard copy of the Skeleton Program
- an electronic version of the Data Files
- an electronic version and a hard copy of the Preliminary Material.

You must **not** use a calculator.

Instructions
- Type the information required on the front of your Electronic Answer Document.
- Before the start of the examination make sure your **Centre Number**, **Candidate Name** and **Candidate Number** are shown clearly **in the footer** of every page (not the front cover) of your Electronic Answer Document.
- Enter your answers into the Electronic Answer Document.
- Answer **all** questions.
- Save your work at regular intervals.

Information
- The marks for questions are shown in brackets.
- The maximum mark for this paper is 75.
- No extra time is allowed for printing and collating.
- The question paper is divided into **three** sections.

Advice
You are advised to allocate time to each section as follows:
Section A – 20 minutes; **Section B** – 25 minutes; **Section C** – 60 minutes.

At the end of the examination
Tie together all your printed Electronic Answer Document pages and hand them to the Invigilator.

Warning
It may not be possible to issue a result for this paper if your details are not on every page of your Electronic Answer Document.

Section A

You are advised to spend no more than **20 minutes** on this section.

Enter your answers to **Section A** in your Electronic Answer Document. You **must save** this document at regular intervals.

Question 3 in this section asks you to write program code **starting from a new program/project/file**.

You are advised to save your program at regular intervals.

0 1 State **two** examples of work that you would expect to undertake during the analysis stage of software development.

[2 marks]

0 2 A sentinel value is a special value **after** the end of a series of data values. It is a terminator for the series of data values but is not treated as part of the series. A sentinel value is used when you do not know how many data values are in the series.

The algorithm, represented using pseudo-code in **Figure 1**, is an attempt at a method to add numbers that are input as a series terminated by the sentinel value -1

Figure 1

```
X ← 0
Result ← 0
WHILE X ≠ -1
  INPUT X
  Result ← Result + X
ENDWHILE
OUTPUT Result
```

0 2 . 1 Complete **Table 1** by hand-tracing the algorithm in **Figure 1**. You may not need to use all the rows in **Table 1**.

The first row of **Table 1** has already been completed for you.

The sequence of numbers for input is: `4, 6, 3, 2, -1`

Table 1

X	Result	Output
0	0	–

Copy the contents of all the unshaded cells in **Table 1** into your Electronic Answer Document.

[3 marks]

0 2 . 2 Comment on the result of the trace and describe how the algorithm should be modified.

[2 marks]

Turn over for the next question

0 3 The algorithm, represented using pseudo-code in **Figure 2**, outputs a series of integers or the message `No result`. The output depends upon the value entered by the user.

Figure 2

```
OUTPUT "Enter an integer greater than 1: "
INPUT X
Product ← 1
Factor ← 0
WHILE Product < X
  Factor ← Factor + 1
  Product ← Product * Factor
ENDWHILE
IF X = Product THEN
  Product ← 1
  FOR N ← 1 TO Factor
    Product ← Product * N
    OUTPUT N
  ENDFOR
ELSE
  OUTPUT "No result"
ENDIF
```

What you need to do:

Task 1
Write a program to implement the algorithm in **Figure 2**.

Task 2
Test that your program works:
- run your program, then enter the number `720`
- run your program, then enter the number `600`

Evidence that you need to provide
Include the following evidence in your Electronic Answer Document.

0 3 . 1 Your PROGRAM SOURCE CODE for **Task 1**.

[9 marks]

0 3 . 2 SCREEN CAPTURE(S) showing the tests described in **Task 2**.

[1 mark]

0 3 . 3 What is true for all valid inputs for `X` that output a number of numbers which is not true for all other valid inputs that output `No result`?

[1 mark]

Section B

You are advised to spend no more than **25 minutes** on this section.

Enter your answers to **Section B** in your Electronic Answer Document. You **must save** this document at regular intervals.

These questions refer to the **Preliminary Material** and the **Skeleton Program**, but do **not** require any additional programming.

Refer **either** to the **Preliminary Material** issued with this question paper **or** your electronic copy.

0 4 State the name of an identifier for:

0 4 . 1 a variable that is used to store a Boolean value.

[1 mark]

0 4 . 2 a user-defined subroutine that returns only **one** value that **must** be a string.

[1 mark]

0 4 . 3 a user-defined subroutine that uses nested **indefinite** iteration.

[1 mark]

0 5 The **Skeleton Program** uses a number of data structures.

0 5 . 1 State the identifier of the data structure that stores values of **more than one data type**.

[1 mark]

0 5 . 2 State the identifier of a data structure that stores values of **only one data type**.

[1 mark]

0 6 This question refers to the constants set at the beginning of the **Skeleton Program**.

It is a good programming technique to use a named constant rather than the value it represents.

State a reason why each of the following are set as constants. Your answers for Questions **06.1** and **06.2** must be different.

0 6 . 1 `EMPTY_STRING`

[1 mark]

0 6 . 2 `MAX_WIDTH`

[1 mark]

Turn over for the next question

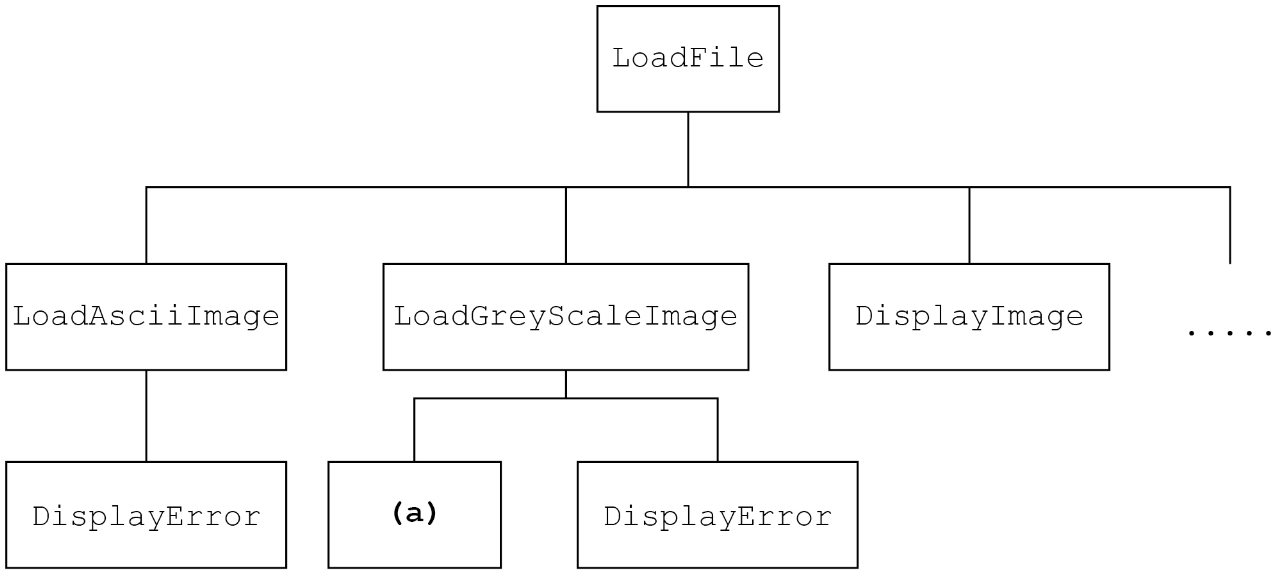
0 7 This question refers to the subroutine `LoadGreyScaleImage`

Describe **two** possible errors that could cause an exception in this subroutine.

[2 marks]

0 8 **Figure 3** shows an incomplete hierarchy chart for part of the **Skeleton Program**.

Figure 3



0 8 . 1 What is the purpose of a hierarchy chart?

[1 mark]

0 8 . 2 What does each box in a hierarchy chart represent?

[1 mark]

0 8 . 3 What should be written in box **(a)** in **Figure 3**?

[1 mark]

0 9 This question refers to the subroutines `Graphics` and `LoadFile`

0 9 . 1 State what values are passed in the parameter `Header` to the subroutine `LoadFile` when `LoadFile` is called for the **first** time in the subroutine `Graphics`

[1 mark]

0 9 . 2 Explain how the content of `HeaderLine` is processed after it has been assigned a value.

[3 marks]

1 0 This question refers to the subroutine `ConvertChar`

With reference to the fact that 0 represents the darkest greyscale (black) and 255 represents the lightest greyscale (white), explain the purpose of `ConvertChar`

[2 marks]

1 1 This question refers to the text file `image3.txt` and the subroutine `LoadAsciiImage`

The first line of the text file `image3.txt` contains:

```
Cat,59,25,A
```

1 1 . 1 Explain the effect of changing the first line to `Cat,25,59,A` and then calling `LoadAsciiImage`

[2 marks]

1 1 . 2 Explain the effect of changing the first line to `Cat,59,59,A` and then calling `LoadAsciiImage`

[2 marks]

1 1 . 3 Explain the effect of changing the first line to `Cat,59,10,A` and then calling `LoadAsciiImage`

[2 marks]

1 2 This question refers to the subroutine `DisplayImage`

Explain what effect swapping around the nested iteration structure in this subroutine would have on the image output, assuming that the image width and height are equal.

Written in pseudo-code the **altered** iteration structure would be:

```
FOR ThisColumn ← 0 TO Header.Width - 1
  FOR ThisRow ← 0 TO Header.Height - 1
    OUTPUT Grid[ThisRow, ThisColumn]
  ENDFOR
  OUTPUT newline
ENDFOR
```

[3 marks]

Turn over for the next section

Section C

You are advised to spend no more than **60 minutes** on this section.

Enter your answers to **Section C** in your Electronic Answer Document. You **must save** this document at regular intervals.

These questions require you to load the **Skeleton Program** and to make programming changes to it.

---

1 3 This question extends the functionality of the **Skeleton Program**. A mirror image is to be produced from the loaded image. For example, **Figure 4** shows the image loaded from the `ascii.txt` data file. **Figure 5** shows the mirror image.

Figure 4

	0	1	2	3	4
0	#	#	#	@	@
1	@	@	@	A	A
2	B	B	!	!	!

Figure 5

	0	1	2	3	4
0	@	@	#	#	#
1	A	A	@	@	@
2	!	!	!	B	B

What you need to do:

Task 1
Write a new subroutine `MirrorImage` that produces a mirror image of the image loaded into `Grid` and stores it in a new data structure. The subroutine then displays the mirror image.

Task 2
Amend subroutine `DisplayMenu` to include the new option:

```
M – Mirror image
```

Task 3
Amend subroutine `Graphics` to call `MirrorImage` when the user chooses option `M`

Task 4
Test that the changes you have made work by conducting the following test:
- run the program
- enter `L`
- load **`image1`**
- enter `M`

Evidence that you need to provide
Include the following evidence in your Electronic Answer Document.

1 3 . 1 Your PROGRAM SOURCE CODE for the entire subroutines `MirrorImage`, `DisplayMenu` and `Graphics`

[6 marks]

1 3 . 2 SCREEN CAPTURE(S) showing the requested test, including the menu and the display of the mirror image.

[1 mark]

Turn over for the next question

1 4 This question refers to the subroutine `LoadGreyScaleImage`

A greyscale image can contain a hidden message, which has been encrypted using a key. The message can be revealed by decrypting the image using the method described below.

If an image contains a hidden message, the key to use is the integer value of the digit at the end of the image title in the first line of the file. For example, if the image title is `image4` then the key is 4

When decrypting a message from an image file, each greyscale value is used in turn with the key to produce a result. The key is subtracted from the greyscale value and the result is interpreted according to **Table 2**.

Table 2 describes the rules that should be followed to determine what secret character, if any, is hidden in the greyscale image.

Table 2

Value of result	Interpretation
0	The decrypted character is the space character.
> 0 AND ≤ 26	The result is the position in the alphabet of the decrypted character.
> 26	The result means that the character was not part of the message that was encrypted, and should be represented by an underscore character, ie the _ character.

Example 1
Greyscale value is 1 and the key is 0
1 – 0 = 1
The decrypted character is A as this is the first letter in the alphabet. This should be appended to the hidden message.

Example 2
Greyscale value is 5 and the key is 3
5 – 3 = 2
The decrypted character is B as this is the second letter in the alphabet. This should be appended to the hidden message.

Example 3
Greyscale value is 7 and the key is 7
7 – 7 = 0
The decrypted character is a space as the result was 0. This should be appended to the hidden message.

Example 4
Greyscale value is 52 and the key is 1
52 – 1 = 51
The result is greater than 26 so the greyscale value was not an encrypted character and an underscore character should be added to the hidden message.

The hidden message is built by concatenating each of the decrypted characters.

What you need to do:

Task 1
Write a new subroutine, `FindSecretChar`, which takes as parameters the `PixelValue` and `Key`, where `Key` is the key used to decrypt the hidden character. The subroutine should return the decrypted character if the pixel value was a character from the message, otherwise it should return the underscore character.

Task 2
Amend the subroutine `LoadGreyScaleImage` to:
- get the key from the image title
- call the subroutine `FindSecretChar` in the appropriate place and build up the hidden message from the returned characters
- output the complete hidden message.

Task 3
Test that the changes you have made work by conducting the following test:
- run the program
- enter `L`
- load `greyscale`

Evidence that you need to provide
Include the following evidence in your Electronic Answer Document.

1 4 . 1 Your PROGRAM SOURCE CODE for the entire subroutines `FindSecretChar` and `LoadGreyScaleImage`

[9 marks]

1 4 . 2 SCREEN CAPTURE(S) showing the requested test, including the display of the message and the image.

[1 mark]

Turn over for the next question

1 5 This question extends the functionality of the **Skeleton Program**. A new option, `C`, is to be added to compress an existing ASCII art image file using run-length encoding and store it in a new file.

The number of symbols in a run of symbols is to be counted and this count, separated by a comma from the symbol, is stored in the new file. Each count and symbol pair should be put on a new line.

For example, **Figure 6** shows the contents of the `ascii.txt` data file.

Figure 6

```
TestImage1,5,3,A
###@@@@@AABB!!!
```

The compressed file will contain the file header followed by several value pairs and the file type in the header should be changed from `A` to `C`, as shown in **Figure 7**.

Figure 7

```
TestImage1,5,3,C
3,#
5,@
2,A
2,B
3,!
```

You can assume that there are no newline characters in the ASCII art image file.

What you need to do:

Task 1
Write a new subroutine, `CompressFile`

This subroutine is to:
- ask the user for the name of the file to be compressed
- read the existing file header and change the file type to `C` to indicate that this is a compressed file
- create a new file with the existing filename preceded by `CMP`, for example, if the file read in was `image2` the new filename should be `CMPimage2`
- save the edited file header to the new file
- save each pair of symbol count and symbol separated by a comma on a new line to the new file.

Task 2
Amend subroutine `DisplayMenu` to include the new option:

```
C – Compress file
```

Task 3
Amend subroutine `Graphics` to call `CompressFile` when the user chooses option `C`

Task 4
Test that the changes you have made work by conducting the following test:
- run the program
- enter `C`
- enter the file name `image2`
- load the file `CMPimage2` into a text editor.

Evidence that you need to provide
Include the following evidence in your Electronic Answer Document.

15.1 Your PROGRAM SOURCE CODE for the entire subroutine `CompressFile`
[12 marks]

15.2 SCREEN CAPTURE(S) showing the requested test, including the menu and all of the contents of the file `CMPimage2` in the text editor.
[1 mark]

END OF QUESTIONS

There are no questions printed on this page

There are no questions printed on this page

There are no questions printed on this page