



Please write clearly in block capitals.

Centre number

--	--	--	--	--

Candidate number

--	--	--	--

Surname

---

Forename(s)

---

Candidate signature

---

I declare this is my own work.

# AS COMPUTER SCIENCE

## Paper 2

Time allowed: 1 hour 30 minutes

### Materials

For this paper you must have:

- a calculator.

### Instructions

- Use black ink or black ball-point pen.
- Fill in the boxes at the top of this page.
- Answer **all** questions.
- You must answer the questions in the spaces provided. Do not write outside the box around each page or on blank pages.
- If you need extra space for your answer(s), use the lined pages at the end of this book. Write the question number against your answer(s).
- Do all rough work in this book. Cross through any work you do not want to be marked.

### Information

- The marks for questions are shown in brackets.
- The maximum mark for this paper is 75.

### Advice

- In some questions you are required to indicate your answer by completely shading a lozenge alongside the appropriate answer as shown. 
- If you want to change your answer you must cross out your original answer as shown. 
- If you wish to return to an answer previously crossed out, ring the answer you now wish to select as shown. 

For Examiner's Use	
Question	Mark
1	
2	
3	
4	
5	
6	
7	
8	
9	
10	
11	
<b>TOTAL</b>	



J U N 2 2 7 5 1 6 2 0 1

IB/G/Jun22/E11

**7516/2**

Answer **all** questions in the spaces provided.

**0 1 . 1**

Describe the difference between natural numbers and integers.

In your answer, give **one** example of a number that is an integer but not a natural number.

**[2 marks]**

---

---

---

---

---

---

---

**0 1 . 2**

Describe what it means for a number to be irrational.

In your answer, give **one** example of an irrational number.

**[2 marks]**

---

---

---

---

---

---

---



0 1 . 3

Shade **one** lozenge in the **Counting** column to indicate which set of numbers is most suitable for counting and **one** lozenge in the **Measuring** column to indicate which set of numbers is most suitable for measuring real-world quantities.

[2 marks]

		Counting	Measuring
<b>A</b>	Integer	<input type="checkbox"/>	<input type="checkbox"/>
<b>B</b>	Natural	<input type="checkbox"/>	<input type="checkbox"/>
<b>C</b>	Rational	<input type="checkbox"/>	<input type="checkbox"/>
<b>D</b>	Real	<input type="checkbox"/>	<input type="checkbox"/>

6

0 2 . 1

Convert the decimal number 177 to unsigned binary using 8 bits.

[1 mark]

---



---



---

0 2 . 2

Convert the decimal number 193 to hexadecimal.

[1 mark]

---



---



---

2

Turn over for the next question

Turn over ►



0 3 . 1

State, **in decimal**, the lowest and highest values that could be represented in unsigned binary when using 16 bits.

[2 marks]

Lowest \_\_\_\_\_

---



---

Highest \_\_\_\_\_

---



---

0 3 . 2

Figure 1 and Figure 2 show the bit patterns of two unsigned binary integers.

Figure 1

0	0	0	1	0	1	0	1
---	---	---	---	---	---	---	---

Figure 2

0	0	0	0	0	1	1	1
---	---	---	---	---	---	---	---

Calculate the result of multiplying these two numbers together using **binary multiplication**.

You **must** show your working in binary.

[2 marks]

---



---



---



---



---



---

Answer \_\_\_\_\_

4
---



**0 4 . 1** ASCII is one character coding system.

Explain the term 'character code'.

**[1 mark]**

---

---

---

**0 4 . 2** Explain why Unicode was introduced as an alternative to ASCII.

**[2 marks]**

---

---

---

---

---

---

---

**Question 4 continues on the next page**

**Turn over ►**



**Figure 3** shows a 7-bit ASCII character code. The character code is to be sent across a network using a parity system.

**Figure 3**

0	0	1	0	1	1	1
---	---	---	---	---	---	---

0 4 . 3

Describe how the parity bit would be generated for the character code in **Figure 3** using even parity.

**[2 marks]**

---

---

---

---

---

---

---

---

0 4 . 4

Write the parity bit below to complete the byte that will be sent using even parity.

**[1 mark]**

	0	0	1	0	1	1	1
--	---	---	---	---	---	---	---







**0 6 . 2** Translators are one type of system software.

Give **two** other types of system software.

**[2 marks]**

Type 1 \_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

Type 2 \_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

**0 6 . 3** Some compilers translate source code into an intermediate language rather than producing an executable file. Bytecode is one example of an intermediate language.

Explain how intermediate language code is used after it has been generated.

**[2 marks]**

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

**0 6 . 4** Give **one** reason why some compilers produce their final output in an intermediate language instead of machine code.

**[1 mark]**

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_  
**6**

**Turn over for the next question**

**Turn over ►**



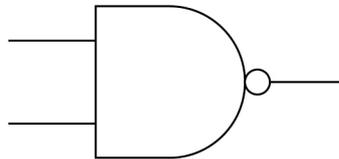
07.1

State which logic gate has the truth table shown in **Figure 4**.**[1 mark]****Figure 4**

A	B	Q
0	0	1
0	1	0
1	0	0
1	1	0

Answer \_\_\_\_\_

07.2

State the logic gate that is represented by the symbol shown in **Figure 5**.**[1 mark]****Figure 5**

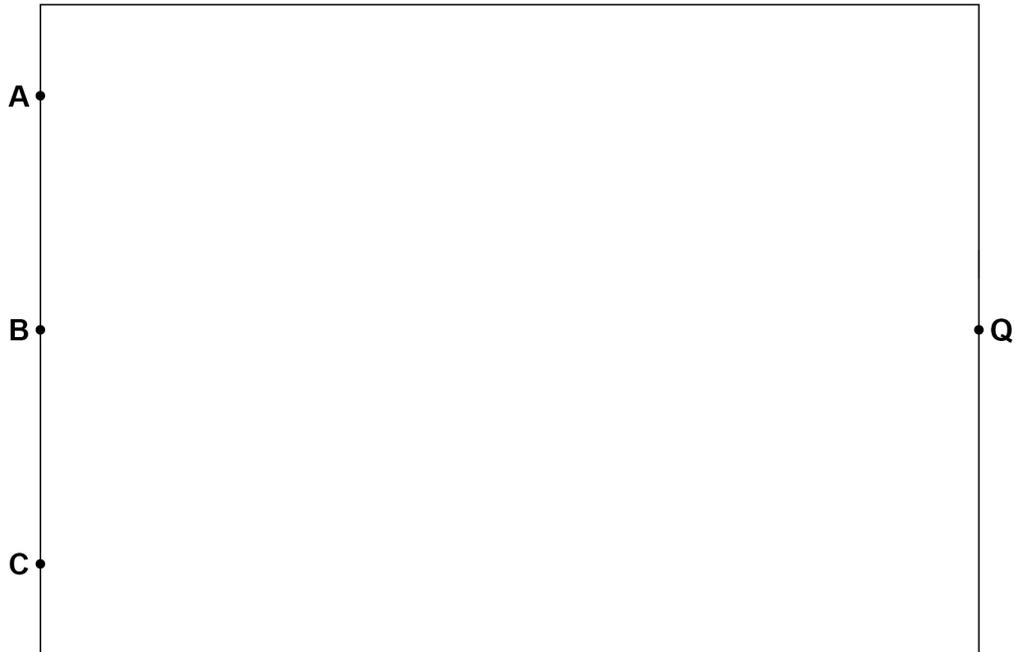
Answer \_\_\_\_\_



**0 7 . 3** Draw the logic circuit for the following Boolean expression.

$$Q = \overline{\overline{A} \cdot \overline{B}} + C$$

**[2 marks]**



**0 7 . 4** Complete the truth table below.

A	B	$\overline{B}$	$(A + \overline{B})$	$(A + \overline{B}) \cdot B$
0	0			
0	1			
1	0			
1	1			

Using the final column, give a simplified Boolean expression for

$$(A + \overline{B}) \cdot B$$

**[3 marks]**

Answer \_\_\_\_\_

**Question 7 continues on the next page**

**Turn over ►**





0 8 . 1

The fetch-execute cycle involves the Current Instruction Register (CIR), Control Unit, Memory Address Register (MAR), Memory Buffer Register (MBR) and Program Counter (PC).

**Figure 6** lists four events that can take place during one cycle of the fetch-execute cycle. The events are labelled **A** to **D**.

Some events that take place during the fetch-execute cycle are not listed.

Put these events in the order they would occur in the fetch-execute cycle when an ADD instruction is executed.

Write the numbers 1 to 4 beside each description in **Figure 6** to indicate the order in which the events occur. The number 1 should be used to indicate the event that would happen first.

**[3 marks]****Figure 6**

	Description	Order (1 to 4)
<b>A</b>	The contents of the MBR are copied to the CIR.	
<b>B</b>	The contents of the PC are copied to the MAR.	
<b>C</b>	The Control Unit decodes the contents of the CIR.	
<b>D</b>	The result of the calculation is stored.	

0 8 . 2

Describe the role of main memory in the execution of computer programs.

**[2 marks]**


---



---



---



---



---



---

**Question 8 continues on the next page**

**Turn over ►**

0 8 . 3

State the name of the processor component that is responsible for performing mathematical operations such as addition and multiplication.

[1 mark]

---

---

---

0 8 . 4

Explain why increasing the data bus width can lead to improvements in processor performance.

[1 mark]

---

---

---

0 8 . 5

Identify the bus that would need to be changed **and** state the change needed so that the maximum amount of memory addressable by the processor would be doubled.

[2 marks]

Bus to change \_\_\_\_\_

Change needed \_\_\_\_\_

---

---

9



**Turn over for the next question**

*Do not write  
outside the  
box*

**DO NOT WRITE ON THIS PAGE  
ANSWER IN THE SPACES PROVIDED**

**Turn over ►**



1 5

**Table 1** shows the standard AQA assembly language instruction set that should be used to answer question **0 9 . 1** and question **0 9 . 2**

**Table 1 – standard AQA assembly language instruction set**

LDR Rd, <memory ref>	Load the value stored in the memory location specified by <memory ref> into register d.
STR Rd, <memory ref>	Store the value that is in register d into the memory location specified by <memory ref>.
ADD Rd, Rn, <operand2>	Add the value specified in <operand2> to the value in register n and store the result in register d.
SUB Rd, Rn, <operand2>	Subtract the value specified by <operand2> from the value in register n and store the result in register d.
MOV Rd, <operand2>	Copy the value specified by <operand2> into register d.
CMP Rn, <operand2>	Compare the value stored in register n with the value specified by <operand2>.
B <label>	Always branch to the instruction at position <label> in the program.
B<condition> <label>	Branch to the instruction at position <label> if the last comparison met the criterion specified by <condition>. Possible values for <condition> and their meanings are: EQ: equal to                      NE: not equal to GT: greater than                LT: less than
AND Rd, Rn, <operand2>	Perform a bitwise logical AND operation between the value in register n and the value specified by <operand2> and store the result in register d.
ORR Rd, Rn, <operand2>	Perform a bitwise logical OR operation between the value in register n and the value specified by <operand2> and store the result in register d.
EOR Rd, Rn, <operand2>	Perform a bitwise logical XOR (exclusive or) operation between the value in register n and the value specified by <operand2> and store the result in register d.
MVN Rd, <operand2>	Perform a bitwise logical NOT operation on the value specified by <operand2> and store the result in register d.
LSL Rd, Rn, <operand2>	Logically shift left the value stored in register n by the number of bits specified by <operand2> and store the result in register d.
LSR Rd, Rn, <operand2>	Logically shift right the value stored in register n by the number of bits specified by <operand2> and store the result in register d.
HALT	Stops the execution of the program.

**Labels:** A label is placed in the code by writing an identifier followed by a colon (:). To refer to a label the identifier of the label is placed after the branch instruction.

#### Interpretation of <operand2>

<operand2> can be interpreted in two different ways, depending on whether the first character is a # or an R:

- # – use the decimal value specified after the #, eg #25 means use the decimal value 25
- Rm – use the value stored in register m, eg R6 means use the value stored in register 6

The available general purpose registers that the programmer can use are numbered 0–12



- 0 9 . 1** Shade **one** lozenge to show which of the assembly instructions in **Figure 7** uses immediate addressing.

[1 mark]

Figure 7

	Instruction	Immediate Addressing
<b>A</b>	LDR R3, 42	<input type="checkbox"/>
<b>B</b>	MOV R3, #42	<input type="checkbox"/>
<b>C</b>	STR R3, 101	<input type="checkbox"/>
<b>D</b>	SUB R3, R2, R1	<input type="checkbox"/>

- 0 9 . 2** A computer program is required that will multiply the value stored in X by 2 if it is less than 50 and leave it unchanged if it is 50 or more.

The algorithm for this task can be written in pseudocode as:

```
IF X < 50 THEN
  X ← X * 2
ENDIF
```

Write an assembly language program using the AQA assembly language instruction set shown in **Table 1** to carry out this task.

At the start, the value of X is stored in memory location 101

[4 marks]

---



---



---



---



---



---



---



---



---



---



---



---

5

Turn over ►



1 0 . 1

Describe the purpose of start and stop bits in asynchronous data transfer.

[2 marks]

Purpose of start bit \_\_\_\_\_

---

---

Purpose of stop bit \_\_\_\_\_

---

---

1 0 . 2

Protocols are used in computer networking.

Define the term 'protocol'.

[1 mark]

---

---

1 0 . 3

Users of a computer network will experience latency.

Define the term 'latency'.

[1 mark]

---

---

1 0 . 4

Explain how a physical star topology can behave logically as a bus network.

[2 marks]

---

---

---

---

---

---

---

6



**Turn over for the next question**

*Do not write  
outside the  
box*

**DO NOT WRITE ON THIS PAGE  
ANSWER IN THE SPACES PROVIDED**

**Turn over ►**







**There are no questions printed on this page**

*Do not write  
outside the  
box*

**DO NOT WRITE ON THIS PAGE  
ANSWER IN THE SPACES PROVIDED**





