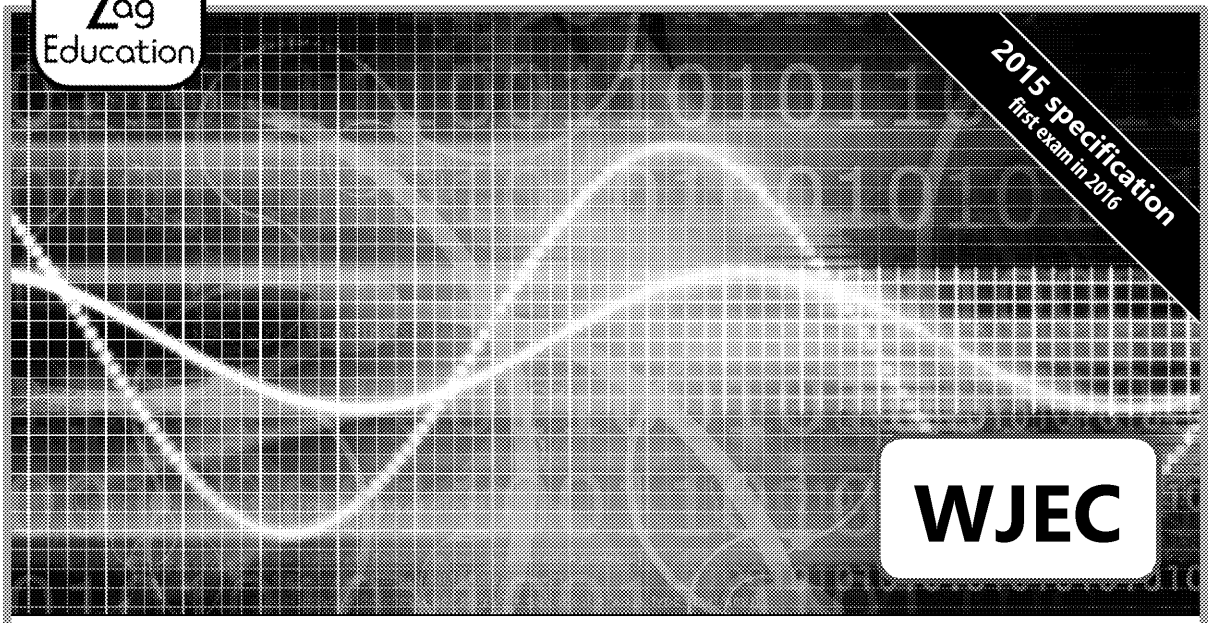
Zig Zag Education

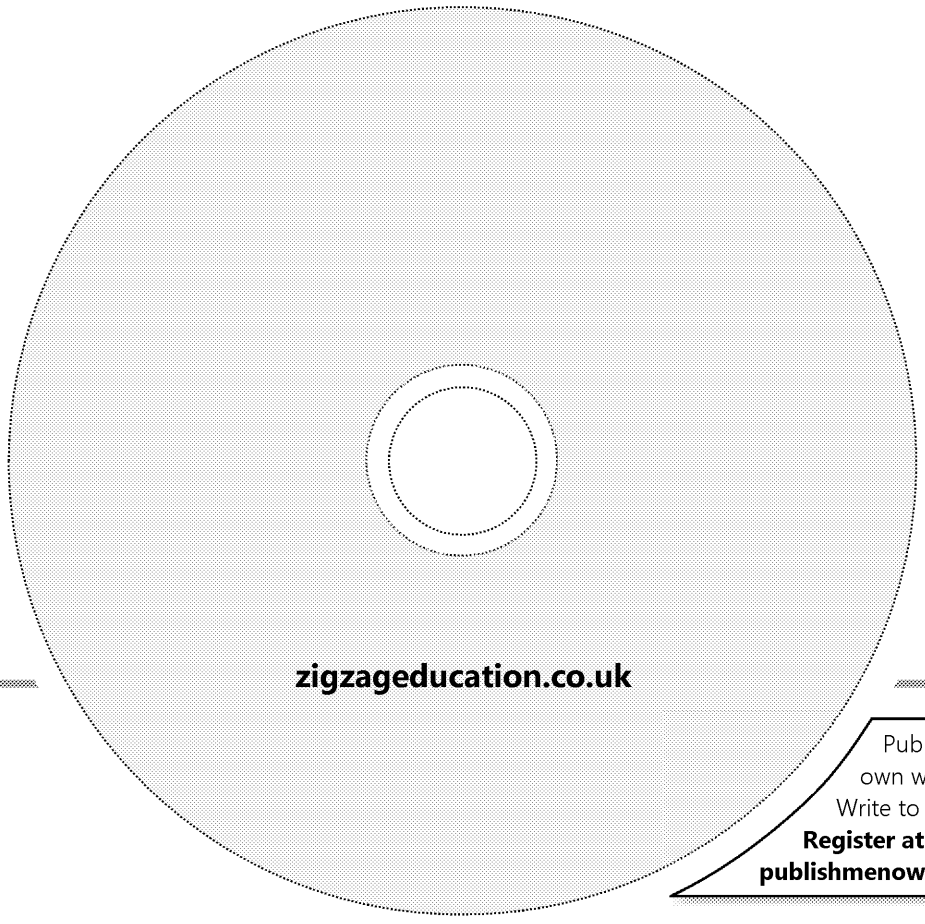
Computer Science AS | WJEC | 2500QS



TECHNICAL TOPICS

Presentations and Worksheets

for AS WJEC Computer Science



BI/ 7644

POD 7644

zigzageducation.co.uk

Publish your own work...
Write to a brief...
Register at publishmenow.co.uk

Contents

Teacher's Introduction

This resource is designed to support the delivery of the logical and mathematical topics from the AS WJEC specification (for first teaching in September 2015; first exams from June 2016).

The topics covered are as follows:

1. *Arrays*
2. *Logic Gates*
3. *Boolean Algebra* *
4. *Sorting Algorithms* *
5. *Searching Algorithms* *
6. *Tracing an Algorithm*
7. *Data Compression*
8. *Sequence and Selection*
9. *Repetition*
10. *Subroutines*
11. *Mathematical Operations*
12. *Binary and Hexadecimal*
13. *Data Types*
14. *Binary Arithmetic*
15. *Negative Numbers*
16. *Floating Point Numbers*
17. *Relational Databases*

* The *Boolean Algebra, Sorting Algorithms* and *Searching Algorithms* topics are taught at AS, but also contain additional requirements at A Level. Because we know that some teachers may want to cover these concepts to AS students anyway, we have included this content even though students are not required to know it for their exams.

This content is indicated using the **A Level Only** stamp, so that you can decide whether or not to use it with your students.

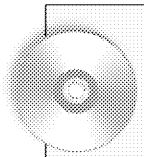
For each of the topics above, there is an animated presentation, providing a step-by-step walk-through of the key concept, plus a worksheet giving students the opportunity to demonstrate their understanding.

These presentations and accompanying worksheets can be used in a number of ways:

- ✓ The animated presentations and worksheets can be used in class to introduce topics.
- ✓ The worksheets can be used as homeworks to test understanding.
- ✓ The animated presentations make perfect revision aids.
- ✓ As part of a flipped classroom, where students watch the animated presentations as preparation for the lesson. The students could complete the worksheets in class to test their understanding prior to a more in-depth discussion of the topic.

The animated presentations are provided in PowerPoint (PPTX), HTML5 and PDF formats. The HTML5 versions are included so that students can use the presentations more easily on devices which lack PowerPoint support (such as tablet computers and even smartphones), making them great for revision. Hard copies of the PDF versions have been included at the back of this pack.

Answers are provided for each worksheet which facilitate self and peer assessment.



The CD-ROM contains the animated presentations in three formats (PPTX, HTML5 and PDF), which are linked together via a HTML frontend (**index.html**).

If using on a network, it is recommended that you provide a shortcut to the frontend to allow easy access for your students.

Alternatively, you can access the individual files directly (without using the frontend), simply by navigating to the relevant folder on the CD.

Free Updates!

Register your email address to receive any future free updates* made to this resource or other Computer Science resources your school has purchased, and details of any promotions for your subject.

** resulting from minor specification changes, suggestions from teachers and peer reviews, or occasional errors reported by customers*

Go to **zzed.uk/freeupdates**

Arrays and Records

Use the names array shown below to help you to answer the following ques

Index	0	1	2	3	4
Value	Susan	Ian	Barbar	Steven	Ben

1. State the value of n[0]. (1)

2. State the value of names[1]. (1)

3. How can the value 'Ben' be accessed in the names array? (1)

4. How can the value 'Sarah' be accessed in the names array? (1)

5. Create the names array using pseudocode. (2)

6. Write a FOR loop that will cycle through the names array, outputting ea

Use the scores array shown below to help you to answer the following ques

	0	1	2	
0	45	71	34	5
1	23	82	57	3
2	18	[illegible]	53	5
3	[illegible]	58	97	4

7. State th[illegible] of scores[3][1]. (1)

8. State the value of scores[1][2]. (1)

9. How can the value 71 be accessed in the scores array? (1)

10. How can the value 97 be accessed in the scores arr[illegible] (1)

11. Create the scor[illegible] using pseudocode. (3)

12 Describe one difference between arrays and records. (2)

Logic Gates

1. Draw the following logic gates: (3)

AND	OR

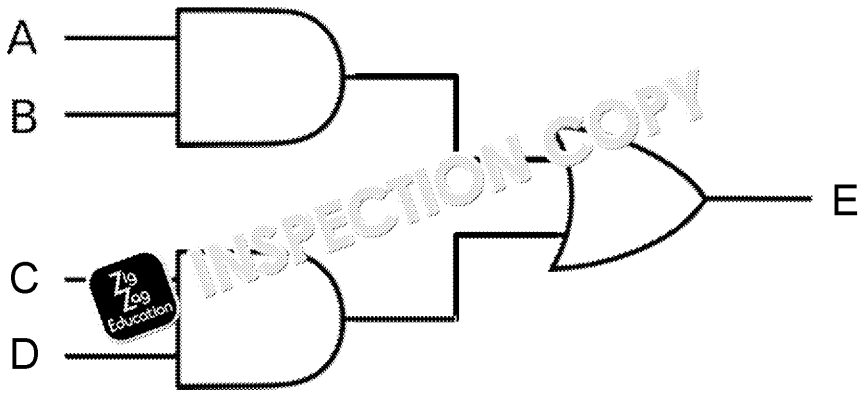
2. Complete this truth table for an OR gate. (4)

Input A	Input B	Input Q

3. Draw the logic diagram for this expression. (3)

$\overline{(A+B).C}$

4. Write a logic statement to represent this logic diagram. (3)



Boolean Algebra

Simplify the following showing each step:

1. $(A + B) . (A + \overline{B})$ (4)

2. $\overline{P}.\overline{Q}.R + P.\overline{Q}.R + P.R$ (3)

3. Which of following statements will simplify to the same expression? (2)

A.1 $A.\overline{A}$ A.A A+1

4. Which of the following is not a statement of De Morgan's Theory? (1)

$\overline{A+B} = \overline{A}.\overline{B}$ $\overline{A.B} = \overline{A} + \overline{B}$ $\overline{A.B} = \overline{A} + \overline{B}$

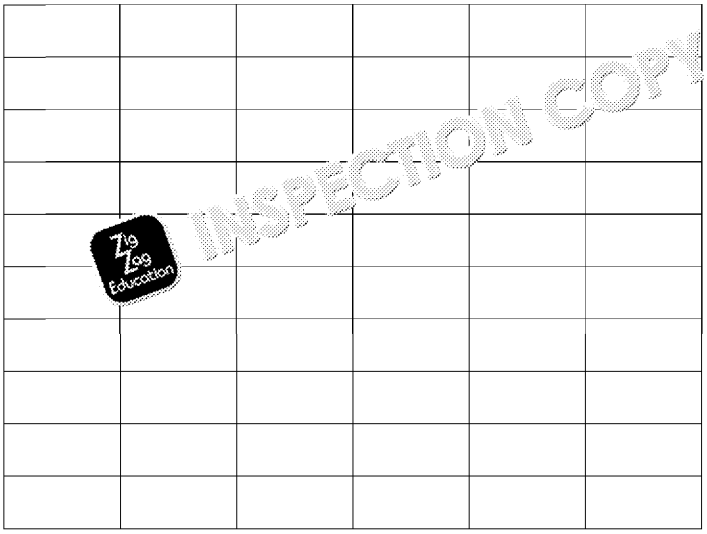
5. Simplify the following using De Morgan's Theory. (3)

$$\overline{(X + Y) . (Y + Z)}$$

Sorting Algorithms

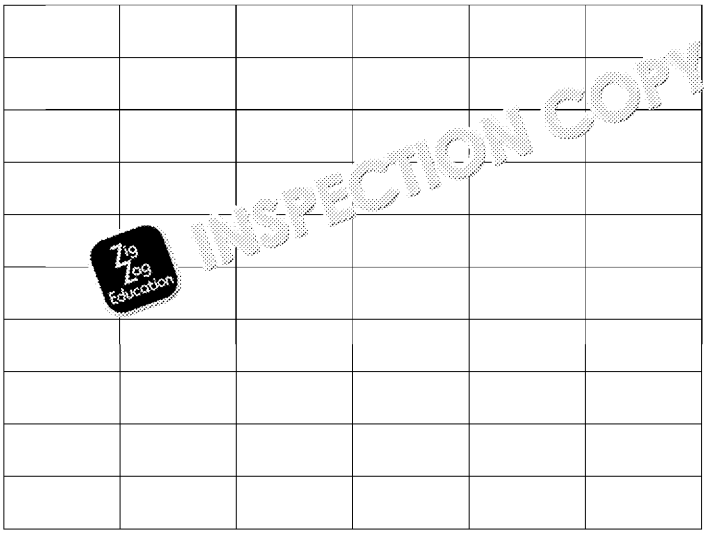
1. Show the list below at each stage of sorting using the bubble sort algor

   45, 32, 2, 78, 5, 9



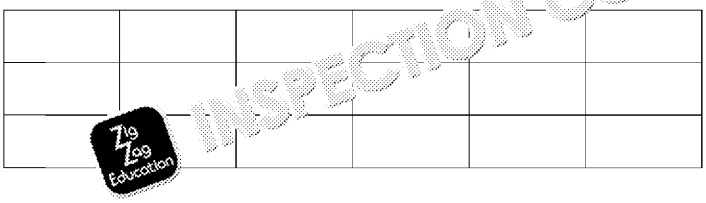
2. Show the list below at each stage of sorting using the insertion sort alg

   45, 32, 2, 78, 5, 9



3. Show the list below at each stage of sorting using the quick sort algorit

   45, 32, 2, 78, 5, 9



Searching Algorithms

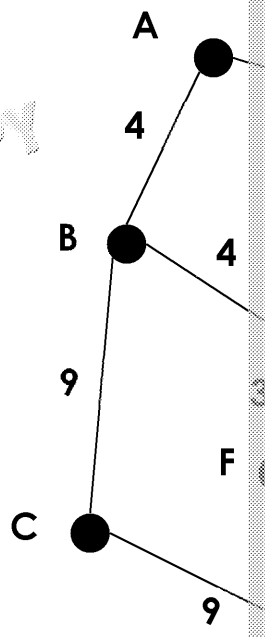
1. Explain how the linear search algorithm works. (2)

2. How many comparisons would be needed to find the value 23 in the list

   5, 9, 11, 15, 23, 47, 58, 81

3. Use Dijksta's shortest path algorithm to find the shortest path between A and the other vertices on the graph below. (8)

Node	Shortest Distance from Vertex A	Previous Node



4. What is the shortest path between vertex A and vertex G? (1)

5. What is the shortest path between vertex A and vertex E? (1)

Tracing an Algorithm

Program A		Program B	
01	A = 1	01	C = 0
02	B = 5	02	WHILE C < 3
03	IF A > B THEN		OUTPUT
04	OUTPUT A		C = C
05	ELSE	05	END WHILE
06	OUTPUT B		
07	END IF		

1. Trace Program A by completing the trace table below. (6)

Line	A	B	OUTPUT	C

2. Trace Program B by completing the trace table below. (14)

Line	C	OUTPUT	Comm

Data Compression

1. Below is an image encoded in binary. Apply run-length encoding to eac

1	1	0	1	1	→	
1	1	0	1	1	→	
0	0	0	0	0	→	
1	1	0	1	1	→	
1	1	0	1	1	→	

2. Apply run-length encoding (RLE) to this string of text: (1)

   CCCCCLLLLLQQQPPPP

3. Suggest possible codes that could be used to represent each line of the dictionary-based compression. (5)

1	1	0	1	1	→	
1	1	0	1	1	→	
0	0	0	0	0	→	
1	1	0	1	1	→	
1	1	0	1	1	→	

4. State one advantage and one disadvantage of lossy compression comp

Advantage:	Disadvantage:

Sequence and Selection

Program A

```
01  INPUT A, B
02  Total = A + B
03  Average = Total / 2
04  OUTPUT Average
```

Program B

```
01  INPUT A, B
02  IF A == B THE
03        OUTPUT
04  ELSE IF A > B
05        OUTPUT
06  ELSE
07        OUTPUT
08  END IF
```

1. Which one of the two programs shown above is an example of a seque

2. Give the line number of a statement from the program you identified in question. (1)

3. Identify the different variables that are used in Program A. (3)

4. Which one of the two programs shown above is an example of selectio

5. Give the line numbers of the conditions from the program you identifie question. (2)

6. Write the meaning of each of the comparison operators shown the in ta

Operator	Meaning
>	
<	
>=	
<=	
=	
<>	

Repetition

Program A

```
01  INPUT Num
02  FOR i = 1 to Num
03        OUTPUT i
04  NEXT i
```

Program B

```
01  Password = "
02  WHILE Passwo
?.        INPUT
14  END WHILE
05  OUTPUT "Pass
```

1. Which one of the t... programs shown above features an example of a

2. Give the line number of the condition from the program you identified i question. (1)

3. Which one of the two programs shown above features an example of a

4. Describe the purpose of Program A. (3)

5. Describe the purpose of Program B. (3)

6. Explain the difference between REPEAT UNTIL and WHILE loops. (3)

7. Rewrite Program A using a WH... loop. (3)

Subroutines

Program A		Program B		P
01	_____ areaCalc(W, H)	01	_____ average(A, B, C)	0
02	Area = W * H	02	Total = A + B + C	0
03	OUTPUT Area	03	Averag = Total / 3	0
04	END _____	04	R.t.n .verage	0
05	areaCalc(10, 8)	05	EN_ _____	0
		06	verage(4, 3, 4)	0
				0

1. Identify the program that contains a function from the three shown above

2. Identify the parameters that are defined in Program B. (3)

3. Identify the arguments that are passed to the areaCalc subroutine in Pro

4. Identify the arguments that are passed to the compare subroutine in Pro

5. What would be the output of the areaCalc subroutine based on the call

6. What would be the output of the compare subroutine based on the call

7. Describe the difference between a procedure and a function. (2)

Mathematical Operations

1. Which symbol is used to represent the exponent operator? (1)

2. Describe the function of the DIV operator. (2)

3. Describe the function of the MOD operator. (2)

4. State the output that will be produced by the following algorithm in each case:

```
01  IF Num MOD 2 > 0 THEN
02        OUTPUT "Valid"
03  ELSE
04        OUTPUT "Invalid"
05  END IF
```

Num = 15	
Num = 16	
Num = 27	

5. State the output that will be produced by the following algorithm when:

```
01  INPUT Num1
02  INPUT Num2
03  Result = Num1 DIV Num2
04  OUTPUT Result
```

Num1 = 7 Num2 = 2	
Num1 = 20 Num2 = 3	
Num1 = 3 Num2 = 3	

Binary and Hexadecimal

1. Convert the decimal number 45 into binary. (2)

128	64	32	16	8	4	2	1

2. Convert the binary number 01101001 into decimal. (2)

3. Convert the decimal number 165 into hexadecimal. (2)

4. Convert the hexadecimal number B6 into decimal. (2)

5. Convert the hexadecimal number 9B into binary. (2)

6. Convert the binary number 10010010 into hexadecimal. (1)

Data Types

1. Select a suitable data type for each of the variables shown in the table.

Variable	Example Data	Data Type
ProductID	64	
ProductName	"Wholemeal Bread"	
InStock	TRUE	
Price	1.50	

2. Calculate the storage requirements for the example data shown below.

Variable	Data	Storage Requirement
ProductID	64	
ProductName	"Wholemeal Bread"	
InStock	TRUE	
Price	1.50	
ProductID	19	
ProductName	"Milk"	

3. An array is used to store the average Computer Science test scores over
Select a suitable data type for the data stored in the array. Justify your c

4. The array mentioned in the previous question stores the average scores
Calculate the storage requirements for the entire array and give your an

Binary Arithmetic

Complete the following calculations (give your answers using 5 bits). (24)

1. 0111
   0101 +

2. 0101
   0111 +

3. 1011
   0111 +

4. 1011
   0011 +

5. 1011
   1011 +

6. 0111
   1010 +

7. 0111
   0111 ×

8. 1101
   0111 ×

9. 1011
   1111 ×

10. 1011
    0101 ×

11. 1011
    1001 ×

12. 0111
    1000 ×

Negative Numbers

1. Convert this sign and magnitude binary integer to decimal. (2)
   `10110110`

2. Convert this sign and magnitude binary integer to decimal. (2)
   `00110010`

3. Convert this two's complement binary integer to decimal. (2)
   `10110110`

4. Convert this two's complement binary integer to decimal. (2)
   `00110010`

5. Convert this decimal number to an 8-bit two's complement binary integer
   `-122`

6. Convert this decimal number to an 8-bit two's complement binary integer
   `72`

7. Convert this decimal number to an 8-bit two's complement binary integer
   `-98`

Complete the following calculations. (6)

8.
```
  0111
- 0101
```

9.
```
  1101
- 0111
```

10.
```
  0011
- 0111
```

Floating Point Numbers

1. Convert this floating-point binary number to decimal: (2)

0	●	1	0	1	0

Mantissa

0	0	1	0

Exponent

2. Convert this floating-point binary number to decimal: (2)

1	●	1	1	1	1

Mantissa

0	0	1	1

Exponent

3. Represent this decimal number using floating-point binary, using 8 bits for the mantissa and 4 bits for the exponent: (2)

-4.5

4. The decimal value 17.27 has been represented as 17.25. Calculate the absolute and relative error: (2)

Absolute:	Relative:

5. Normalise this floating-point binary number: (2)

1	●	1	0	1	1	0

Mantissa

0	1	0

Exponent

6. Normalise this floating-point binary number: (2)

0	●	0	1	1	0

Mantissa

0	0	1

Exponent

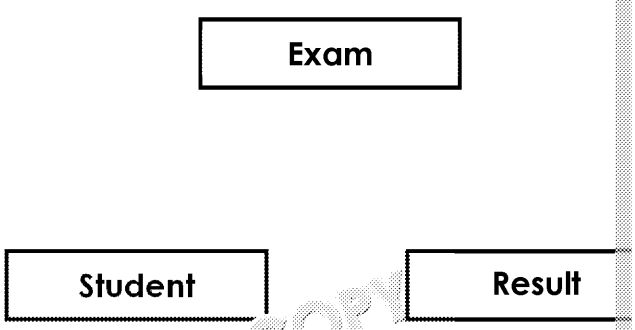
Relational Databases

An exam board uses a database to store the results of each exam for each s
tables: Student, Exam and Result.

1. Details of the three tables are displayed below. Un[illegible]ine the correct fie
for each table. (3)

Student (StudentID, Fo[illegible], Surname, School, Gender, Da
Exam (ExamID, [illegible], Subject, Level)
Result ([illegible]mID, StudentID, RawMark, Grade)

2. Draw the correct relationships between each of the tables. (2)



3. Which part of a da[illegible]e [illegible]bles all the data about one object or person

4. Underline the foreign keys used in the database. (2)

Student (StudentID, Forename, Surname, School, Gen
Exam (ExamID, ExamName, Subject, Level)
Result (ExamID, StudentID, RawMark, Grade)

Answers

Arrays and Records

1. Steven
2. Ian
3. `names[4]`
4. `names[6]`
5. *1 mark initialising the names [illegible] correctly, 1 mark for using " " around the*
   `names ← ["Susan", "Ian", "Barbara", "Steven", "Ben", "Polly`
6. *1 mark for [illegible] correct range, 1 mark for using the counter variable (i) [illegible] value [illegible] the names array, 1 mark for correct indentation*

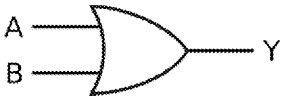
```
FOR i ← 0 to 6
    OUTPUT names[i]
NEXT i
```

7. 58
8. 57
9. scores[0][1]
10. Scores[3][2]
11. *1 mark for initialising the scores array, 1 mark for initialising it as a 2D arra*
    `scores ← [[45, 71, 34, 55],[23, 82, 57, 37],[18, 31, 53, 5`
12. All elements in an array must be of the same data type (1) whereas each f[illegible] data type (1).

Logic Gates

1. *AND gate:*

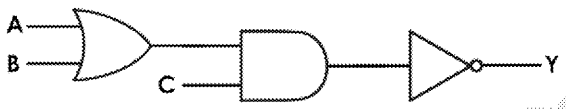
OR gate:



2. *1 mark for each correct line:*

Input A	Input B	Output Q
0	0	0
0	1	1
1	0	1
1	1	1

3. *1 mark for each correctly placed symbol:*



4. E = (A.B) + (C.D)
   One mark for (A.B).
   One mark for (C.D)
   One [illegible] or

Boolean Algebra

1. $A.A + A.\overline{B} + B.A + B.\overline{B}$ *(1 mark for each step)*
   $A + A.B + \overline{B}.A$
   $A + A$
   A

2. $P.R + P.R$ *(1 mark for each step)*
   $R+R$
   R

3. $A.1$ *(1 mark)* $A.A$ *(1 mark)*

4. $\overline{A.B} = \overline{A} + \overline{B}$ *(1 mark)*

5. $\overline{(X + Y).(Y + Z)}$ *(1 mark for initial use of De Morgan)*
   $(\overline{X} . \overline{Y}) + (\overline{Y} . \overline{Z})$ *(1 mark for repeated use of De Morgan)*
   $\overline{Y}.(\overline{X} + \overline{Z})$ *(1 mark for final answer)*

Sorting Algorithms

1. *1 mark per correct row:*

45	32	2	78	5	9
32	45	2	78	5	9
32	2	45	78	5	9
32	2	45	5	78	9
32	2	45	5	9	78
2	32	45	5	9	78
2	32	5	45	9	78
2	32	5	9	45	78
2	5	32	9	45	78
2	5	9	32	45	78

2. *1 mark per correct row:*

45	32	2	78	5	9
32	45	2	78	5	9
32	2	45	78	5	9
2	32	45	78	5	9
2	32	45	5	78	9
2	32	5	45	78	9
2	5	32	45	78	9
2	5	32	45	9	78
2	5	32	9	45	78
2	5	9	32	45	78

3. *1 mark per correct row:*

45	32	2	78	5	9
2	5	9	45	32	78
2	5	9	32	45	78

Searching Algorithms

1. Searches each element starting at the start of the list (1) until it finds a ma
2. 1
3. *One mark per correct row*

Node	Shortest Distance from vertex A	Previous Node
A	0	
B	4	
C	13	B
D	15	F
E	7	H
F	11	G
	8	B
H	5	A

4. A – B – G
5. A – H – G

Tracing an Algorithm

1. *Award 1 mark for each correct line:*

Line	A	B	OUTPUT	Comment
01	1			
02		5		
03				FALSE, go to line 05
05				
06			5	
07				

2. *Award 1 mark for each cor[illegible]*

Line	C	[illegible]	Comment
[illegible]			
0[illegible]			TRUE, go to line 03
03		0	
04	1		
05			Return to line 02
02			TRUE, go to line 03
03		1	
04	2		
05			Return to line 02
02			TRUE, got to line 03
03		2	
04	3		
05			Return to line 02
02			FALSE, end [illegible]

Data Compression

1. *1 m[illegible] e[illegible]*

1		0	1	1	→	2(1), 1(0), 2(1)
1	1	0	1	1	→	2(1), 1(0), 2(1)
0	0	0	0	0	→	5(0)
1	1	0	1	1	→	2(1), 1(0), 2(1)
1	1	0	1	1	→	2(1), 1(0), 2(1)

2. 5(C), 5(L), 3(Q), 4(P)
3. *1 mark for each row; accept any two suitable values:*

1	1	0	1	1	→	00
1	1	0	1	1	→	00
0	0	0	0	0	→	01
1	1	0	1	1	→	00
1	1	0	1	1	→	00

4. Advantage: lossy compression usually results in significantly smaller files.
   Disadvantage: permanently discards some of the data.

Sequence & Selection

1. Program A
2. *Any line from Program A*
3. A, B and Average
4. Program B
5. 02 and 04 (from program B)
6. *1 mark per correct row:*

Operator	Meaning
>	Greater than
<	Less than
>=	Greater than or equal to
<=	Less than or equal to
=	Equal to
<>	Not equal to

Repetition

1. Program B
2. Line 03
3. Program A
4. It asks the user to input a number (1) and uses a FOR loop (1) to output the numbers the user inputted (1).
5. The value of the password variable is set to 'none' (1). A WHILE loop is used to input their password (1) until they enter the value 'turing' at which point it
6. In a WHILE loop the condition is tested at the start, whereas in a REPEAT UNTIL loop the condition is tested at the end (1). A WHILE loop repeats until the condition is TRUE, whereas a REPEAT UNTIL loop repeats until the condition is FALSE (1). In a WHILE loop the statements may never be executed (1), whereas in a REPEAT UNTIL loop the statements will always be executed at least once (1).
7. *Award 1 mark for each of the following (also refer to the example below):*
   - The user is asked to input a number.
   - The counter (i) has been initialised correctly (value = 1) before the loop.
   - The condition is correct.

   Example:

```
01 INPUT Num
02
03 WHILE i <= Num
04     OUTPUT i
05     i = i + 1
06 END WHILE
```

Subroutines

1. Program B
2. A, B and C
3. 10 and 8
4. 6 and 4
5. 80
6. 6
7. A procedure is a subroutine that does not normally return a value (1), whe returns values to the calling routine (1).

Mathematical Operations

1. ^
2. To perform a division operation (1) and return an integer (whole number)
3. To perform a division operation (1) and return the remainder (1)
4. "Invalid" / "Valid" / "Valid"
5. 3 / 6 / 11

Binary and Hexadecimal

1. 00101101 *(1 mark for the correct answer and 1 mark for working)*
2. 101 *(1 mark for the correct answer and 1 mark for working)*
3. A5 *(1 mark for the correct answer and 1 mark for working)*
4. 182 *(1 mark for the correct answer and 1 mark for working)*
5. 10011011 *(1 mark for the correct answer and 1 mark for working)*
6. 92 *(1 mark for the correct answer and 1 mark for working)*

Data Types

1. *One mark per row.*

Variable	Example Data	Data Type
ProductID	64	Integer
ProductName	"Wholemeal Bread"	String
InStock	TRUE	Boolean
Price	1.50	Real

2. *One mark per row:*

Variable	Data	Storage Requirements
ProductID	64	2 or 4 bytes
ProductName	"Wholemeal Bread"	15 bytes
InStock	TRUE	1 bit
Price	1.50	4 or 8 bytes
ProductID	19	2 or 4 bytes
ProductName	"Milk"	4 bytes

3. Real (1) because the average scores may have a fractional part/decimal pl
4. 10 x 4 = 40 bytes / 10 x 8 = 80 b

Binary Arithmetic

(1 mark for the correct answer and 1 mark for working)

1. 011
2. 01100
3. 010010
4. 01110
5. 010110
6. 010001
7. 110001
8. 1011011
9. 10100101
10. 110111
11. 1100011
12. 111000

Negative Numbers

(1 mark for the correct answer and 1 mark for working)

1. -54
2. 50
3. -74
4. 50
5. 10000110
6. 01001000
7. 10011110
8. 0010
9. 0110
10. 1100

Floating Point Numbers

1. -2.5 *(1 mark for the correct answer and 1 mark for working)*
2. 13.[illegible] *(1 mark for the correct answer and 1 mark for working)*
3. Mantissa: [illegible].0111000 Exponent: 0011
4. Absolute: 0.02 Relative: 0.001158
5. Mantissa: 0.110110 Exponent: 0101
6. Mantissa: 0.110 Exponent: 0001

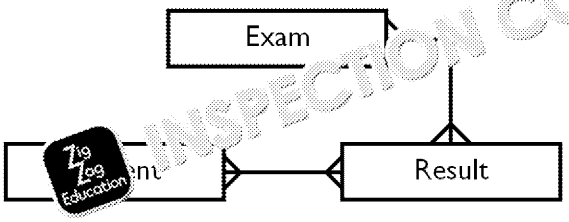
Relational Databases

1. *One mark for each primary key*

   Student (StudentID, Forename, Surname, School, Gender, Date
   Exam (ExamID, ExamName, Subject, Level)
   Result (ExamID, StudentID, RawMark, Grade)

   Note: ExamID and StudentID combined form the unique identifier for the result tabl

2. *One mark for each correct relationship:*



3. Record
4. *1 mark for each correctly identified foreign key*

   Student (StudentID, Forename, Surname, School, Gender, Date
   Exam (ExamID, ExamName, Subject, Level)
   Result (ExamID, StudentID, RawMark, Grade)