1. Convert the binary number `11010100` into decimal.

(Total 1 mark)

2. Convert the binary number `10111001` into hexadecimal.

You should show your working.

(Total 2 marks)

3. State the largest decimal number that can be represented using 6 bits.

(Total 1 mark)

4. Add together the following three binary numbers and give your answer in binary.

	0	0	1	1	0	1	1	0
	1	0	0	1	0	0	1	0
+	0	0	1	0	0	0	0	1

(Total 2 marks)

5. Apply a binary shift three places to the right on the bit pattern `10101000`

Give the result using 8 bits.

(Total 1 mark)

6. The arithmetic effect of applying a left binary shift of two to a binary number is to multiply that number by four.

State the arithmetic effect of applying a left binary shift of four to a binary number.

(Total 1 mark)

7. The arithmetic effect of applying a left binary shift of two to a binary number is to multiply that number by four.

State the arithmetic effect of applying a left binary shift of three followed by a right binary shift of five to a binary number.

(Total 1 mark)

8. How many bits are there in two kilobytes?

Show your working.

(Total 2 marks)

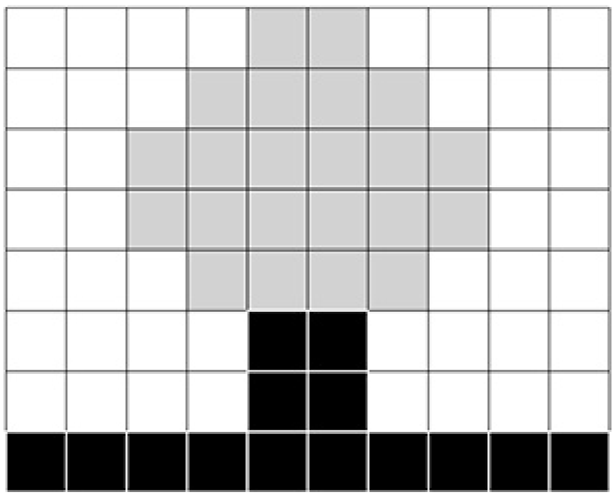
9. The ASCII value for the character x is the decimal number 120

Complete the table below with the missing ASCII and Unicode values.

Character	ASCII value	Unicode value
w		
x	120	
y		
z		

(Total 2 marks)

10. The figure below shows a 10 x 8 bitmap image that uses three colours.



Calculate the minimum file size that would be required to store the bitmap image in the figure above.

Give your answer in **bytes**.

Show your working.

(Total 3 marks)

11. Analogue sound must be converted to a digital form for storage and processing in a computer.

(a) Define the term **sample resolution**.

(1)

(b) State **one** disadvantage of a high sample resolution.

(1)

(c) A 50-second sound has been recorded at a sample rate of 40 000 Hz.
Two bytes have been used to store each sample of the sound.

Calculate the file size of the sound file in **megabytes**.

Show your working.

(2)
(Total 4 marks)

12. State **two** reasons why data are compressed.

(Total 2 marks)

13. The figure below shows a string.

`MISSISSIPPI`

One method for compressing data is run length encoding (RLE).

When using RLE, the data in the figure above becomes:

`1M 1I 2S 1I 2S 1I 2P 1I`

Explain why RLE is **not** a suitable method for compressing the data in the figure above.

(Total 2 marks)

14. **Figure 1** shows a string.

Figure 1

`MISSISSIPPI`

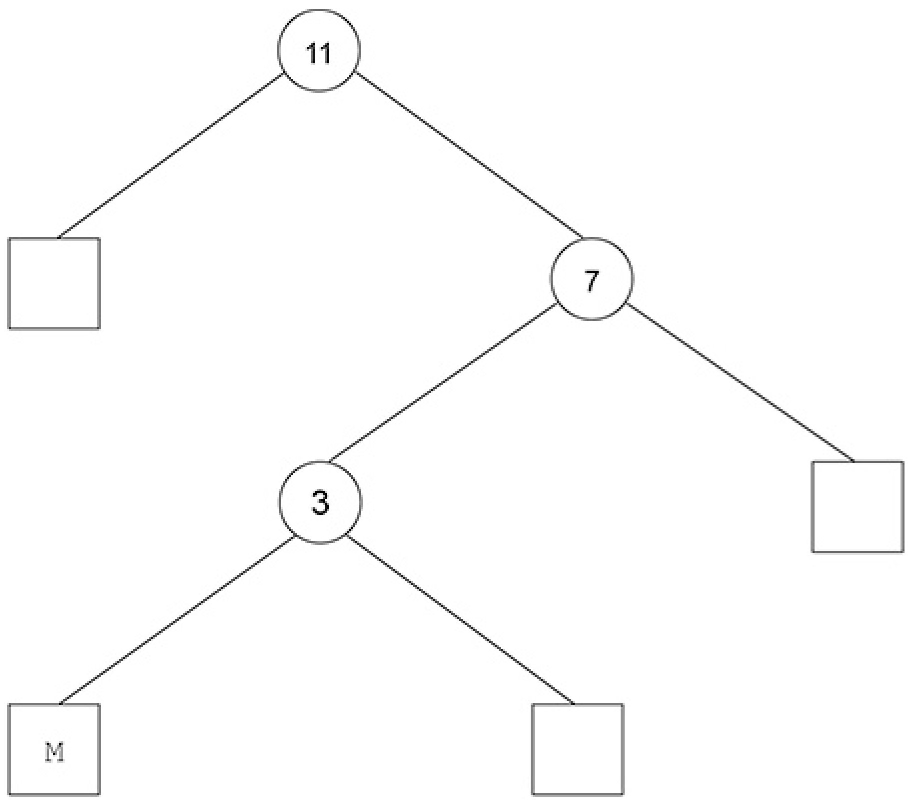
Another method for compressing data is Huffman coding. In Huffman coding, the codes for the characters can be created based on their position in a tree.

Figure 2 shows a Huffman code for each different character in the string in **Figure 1**.

Figure 2

Character	Binary code
M	100
I	0
S	11
P	101

Complete the Huffman tree below to show the position of the characters I, S and P using the codes from **Figure 2**.



(Total 1 mark)

15. Shade **one** lozenge that states why Unicode is now commonly used in preference to ASCII.

A	Unicode can be represented in hexadecimal.	⬭
B	Unicode includes characters from many different alphabets.	⬭
C	Unicode is a sequential character set.	⬭
D	Unicode is easier to remember than ASCII.	⬭
E	Unicode takes up less space in memory than ASCII.	⬭

(Total 1 mark)

16. A student has written the following statements about representing images. Two are correct and two are incorrect:

Statement 1
"Bitmap images are made up of pixels."

Statement 2
"A 2 pixel by 4 pixel bitmap image contains 16 pixels."

Statement 3
"A pixel is a single point in a graphical image."

Statement 4
"Black and white images have a minimum colour depth of two."

Write the correct versions of the **two** incorrect statements that the student has made.

(Total 2 marks)

17. Calculate the minimum file size in **bits** of a 10 pixel by 10 pixel image with a colour depth of 3 bits.

(Total 1 mark)

18. Calculate the minimum file size in **bytes** of a 10 pixel by 10 pixel image with 12 different colours.

You should show your working.

(Total 3 marks)

19. Number the following lines of code in order (**1**–**4**) so that they create an algorithm where the final value of the variable `n` is 13.

The `LEFTSHIFT` operator performs a binary left shift.
For example, `4 LEFTSHIFT 2` would left shift the value 4 twice.

Line of code	**Position (1–4 where 1 is the first line)**
`t ⟵ t - 1`	
`n ⟵ t – n`	
`n ⟵ 2`	
`t ⟵ n LEFTSHIFT 3`	

(Total 3 marks)

20. Convert the decimal number 220 into binary.

(Total 1 mark)

21. Convert the hexadecimal number AD into binary.

You should show your working.

(Total 2 marks)

22. Convert the hexadecimal number 1A into decimal.

(Total 1 mark)

23. What is the largest hexadecimal number that can be represented in binary using 8 bits?

(Total 1 mark)

24. Which of the following is a reason why hexadecimal is used instead of binary?

Shade **one** lozenge.

A Computers work in hexadecimal, not binary. ⬭

B Hexadecimal can be used to represent a wider range of numbers. ⬭

C Hexadecimal is a standard language and binary is not. ⬭

D Hexadecimal is more compact when displayed on screen. ⬭

(Total 1 mark)

25. The code below shows a value represented as a bit pattern.

```
1   0   1   1   0   0   0   0
```

A binary shift can be used to divide the value in the code above by 4.

What is the result of this shift?

Your answer **must** be in binary.

(Total 1 mark)

26. Add the following binary numbers and give your answer in binary.

```
  1 0 1 1 0 1 0 0
  0 0 1 0 0 0 1 0
+ 0 0 0 1 0 1 0 1
  _______________

  _______________
```

(Total 2 marks)

27. Eight minutes of sound has been digitally recorded. The sampling rate used was 25 000 Hertz and the sample resolution used was 4 bits.

(a) Calculate the minimum file size for the recording. Give your answer in **megabytes**.

You should show your working.

(4)

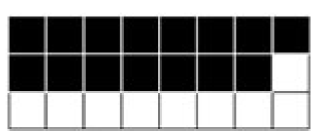
(b) Explain what effects increasing the sampling rate would have on the recording.

(2)

(Total 6 marks)

28. **Figure 1** shows a black and white image.

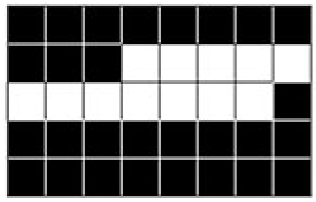
Figure 1



The image shown in **Figure 1** could be compressed using Run Length Encoding (RLE). The RLE for the image in **Figure 1** is B15 W9.

(a) **Figure 2** shows another black and white image.

Figure 2



Give the RLE for the image shown in **Figure 2**.

(1)

(b) The RLE will be represented using binary, with one bit representing the colour (W = 1, B = 0) followed by seven bits representing the frequency.

Give the binary representation of the RLE: B15 W9

(2)
(Total 3 marks)

29. The table below is a frequency table that contains the frequency of characters in a string.

A	6
B	2
C	3

Use the frequencies given in the table above to draw a Huffman tree that represents the string.

(Total 3 marks)

30. The table below shows the Huffman codes for the characters used in the string PIEDPIPER

Character	Character frequency	Huffman code
P	3	10
I	2	11
E	2	01
D	1	000
R	1	001

Calculate how many bits would be saved if the phrase PIEDPIPER was encoded using the Huffman codes shown in above table, rather than using ASCII.

You should show your working.

(Total 3 marks)

31. A bitmap image is represented as a grid of pixels.

State what is meant by the term pixel.

(Total 1 mark)

32. State the maximum number of different colours that can be used if a bitmap image has a colour depth of six bits.

(Total 1 mark)

33. What is the minimum file size for an 800 pixel by 1000 pixel bitmap image that uses 20 different colours? You should give your answer in **kilobytes**.

You should show your working.

(Total 3 marks)

34. The algorithms shown in **Figure 1** and **Figure 2** both have the same purpose.

The operator `LEFTSHIFT` performs a binary shift to the left by the number indicated.

For example, `6 LEFTSHIFT 1` will left shift the number `6` by one place, which has the effect of multiplying the number `6` by two giving a result of 12

Figure 1

```
result ← number LEFTSHIFT 2
result ← result – number
```

Figure 2

```
result ← 0
FOR x ← 1 TO 3
     result ← result + number
ENDFOR
```

(a) Complete the trace table for the algorithm shown in **Figure 1** when the initial value of `number` is 4

You may not need to use all rows of the trace table.

result

(2)

(b) Complete the trace table for the algorithm shown in **Figure 2** when the initial value of `number` is 4

You may not need to use all rows of the trace table.

`x`	`result`

(2)

(c) The algorithms in **Figure 1** and **Figure 2** have the same purpose.

State this purpose.

(1)

(d) Explain why the algorithm shown in **Figure 1** can be considered to be a more efficient algorithm than the algorithm shown in **Figure 2**.

(1)
(Total 6 marks)

35. State the **hexadecimal** representation of the binary number `10010100`

(Total 1 mark)

36. State the **decimal** representation of the binary number `10010100`

(Total 1 mark)

37. State the **hexadecimal** representation of the decimal number 143

You should show your working.

(Total 2 marks)

38. State the **binary** representation of the hexadecimal number BE

You should show your working.

(Total 2 marks)

39. Give **two** reasons why hexadecimal is often used instead of binary in computer science.

(Total 2 marks)

40. Add together the following three binary numbers and give your answer in binary.

```
  0 1 0 1 0 1 0 1
  0 1 1 0 1 1 0 0
+ 0 0 0 1 1 0 0 1
  _______________

  _______________
```

(Total 2 marks)

41. State the result, in binary, of performing a binary shift two places to the left on the binary value `00111001`

(Total 1 mark)

42. What is the largest decimal number that can be represented using 6 bits?

(Total 1 mark)

43. How many bits are there in 5 kB?

You should show your working.

(Total 2 marks)

44. Explain how a sound wave is converted so that it can be stored in a computer.

(Total 3 marks)

45. A student has recorded a 30 second digital sound track using a sample rate of 44 000Hz. 8 bits have been used to store each sample taken.

Calculate the file size **in kilobytes** of the digital sound track.

You should show your working.

(Total 2 marks)

46. Shade **one** lozenge to show which statement best describes data compression.

A The process of calculating the file size of a saved file. ⬭

B The process of encoding characters into more than one language. ⬭

C The process of encoding information to try and use fewer bits than the original. ⬭

D The process of removing necessary data from a file. ⬭

(Total 1 mark)

47. Give **two** reasons why data compression is often used.

(Total 2 marks)

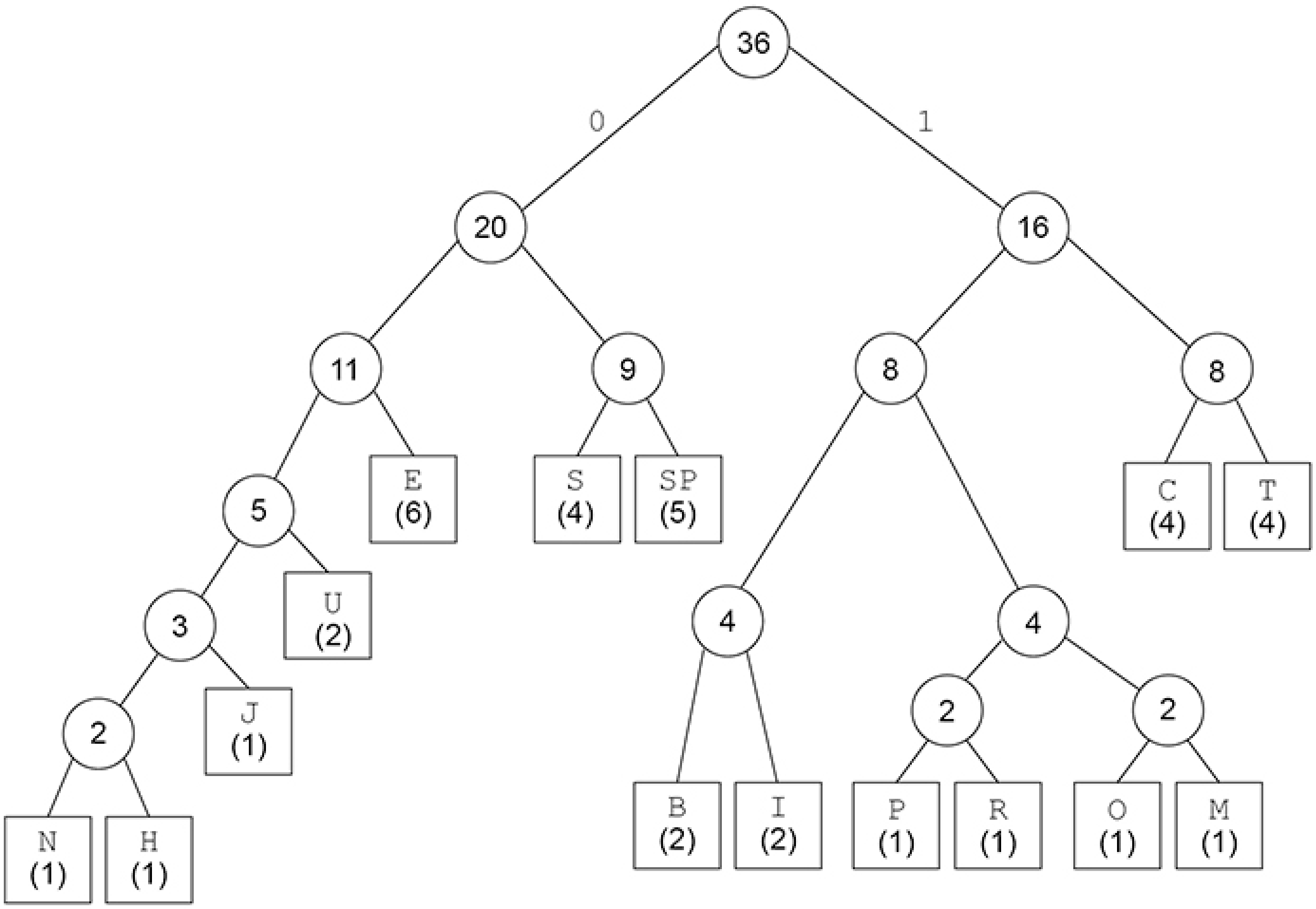
48. Run length encoding (RLE) is one method of compressing data.

State the feature of data that allows it to be compressed effectively using RLE.

(Total 1 mark)

49. **Figure 1** shows a Huffman tree that has been created to represent the string shown in **Figure 2**.

Figure 1



SP represents a space character

Figure 2

`COMPUTER SCIENCE IS THE BEST SUBJECT`

(a) Use the Huffman tree in **Figure 1** to state the Huffman encoding for the string `MOST`

M	**O**	**S**	**T**

(3)

(b) A student was asked to describe how a Huffman tree could be created for the string in **Figure 2**. Her response was:

"I would count the number of times each character appears in the string and create a frequency table sorted alphabetically. For example, the letter S has the highest frequency in **Figure 2**. Next I would take the two characters with the highest frequencies and combine them into a new node. The new node would be added to the end of the frequency table. The two characters with the lowest remaining frequencies are now combined into a new node and the process is repeated until all the characters have been added to nodes and the tree created."

State **four** mistakes the student has made in her response.

(4)

(c) When the Huffman tree in **Figure 1** is used, the string in **Figure 2** can be represented using 130 bits.

The 36-character string shown in **Figure 2** could also be encoded using ASCII.

How many bits are **saved** when Huffman coding is used rather than ASCII to represent the string shown in **Figure 2**?

You **must** show your working.

(2)
(Total 9 marks)

50. The pseudo-code below assigns two string values to two variables.

```
title ← 'computer science'
level ← 'gcse'
```

(a) Shade **one** lozenge that shows the length of the contents of the variable `level` in the pseudo-code.

A 1 ⬭

B 2 ⬭

C 3 ⬭

D 4 ⬭

(1)

(b) Shade **one** lozenge that shows the result of concatenating the variable `title` with the variable `level` in the pseudo-code.

A '`computer science gcse`'

B '`Computer Science GCSE`'

C '`computersciencegcse`'

D '`computer sciencegcse`'

(1)

(c) Shade **one** lozenge to show which of the following strings is a substring of the variable `title` in the pseudo-code.

A '`compsci`'

B '`puters`'

C '`sci`'

D '`tersci`'

(1)

(d) The Unicode character code of `title[0]`, which is '`c`', is 99.

Shade **one** lozenge to show the Unicode character code of the character `level[3]` in the pseudo-code.

A 95

B 99

C 101

D 103

(1)
(Total 4 marks)

51. Run length encoding (RLE) is a form of compression that creates frequency / data pairs to describe the original data.

For example, an RLE of the bit pattern `00000011101111` could be `6 0 3 1 1 0 4 1` because there are six `0`s followed by three `1`s followed by one `0` and finally four `1`s.

The algorithm below is designed to output an RLE for a bit pattern that has been entered by the user.

Five parts of the code labelled **L1**, **L2**, **L3**, **L4** and **L5** are missing.

- Note that indexing starts at zero.

```
pattern ⟵ L1
i ⟵ L2
count ⟵ 1
WHILE i < LEN(pattern)-1
     IF pattern[i] L3 pattern[i+1] THEN
          count ⟵ count + 1
     ELSE
          L4
          OUTPUT pattern[i]
          count ⟵ 1
     ENDIF
     L5
ENDWHILE
OUTPUT count
OUTPUT pattern[i]
```

(a) Shade **one** lozenge to show what code should be written at point **L1** of the algorithm.

A	`OUTPUT`	⬭
B	`'RLE'`	⬭
C	`True`	⬭
C	`USERINPUT`	⬭

(1)

(b) Shade **one** lozenge to show what value should be written at point **L2** of the algorithm.

A	`-1`	⬭
B	`0`	⬭
C	`1`	⬭
C	`2`	⬭

(1)

(c) Shade **one** lozenge to show what operator should be written at point **L3** of the algorithm.

A	`=`	⬭
B	`≤`	⬭
C	`<`	⬭
C	`≠`	⬭

(1)

(d) Shade **one** lozenge to show what code should be written at point **L4** of the algorithm.

A	`count`	⬭
B	`count ⟵ count – 1`	⬭
C	`count ⟵ USERINPUT`	⬭
C	`OUTPUT count`	⬭

(1)

(e) Shade **one** lozenge to show what code should be written at point **L5** of the algorithm.

A	`i ⟵ i * 2`	⬭
B	`i ⟵ i + 1`	⬭
C	`i ⟵ i + 2`	⬭
C	`i ⟵ i DIV 2`	⬭

(1)

(f) State a run length encoding of the series of characters `ttjjeeess`

(2)

(g) A developer implements the algorithm and tests their code to check that it is working correctly. The developer tests it only with the input bit pattern that consists of six zeros and it correctly outputs `6 0`.

Using example test data, state **three** further tests that the developer could use to improve the testing of their code.

(3)
(Total 10 marks)

52. Convert the decimal number 197 into binary.

(Total 1 mark)

53. Convert the hexadecimal number A4 into decimal.

Show your working.

(Total 2 marks)

54. What is the largest decimal number that can be represented using 5 bits?

(Total 1 mark)

55. How many bits are there in 3 MB?

Show your working.

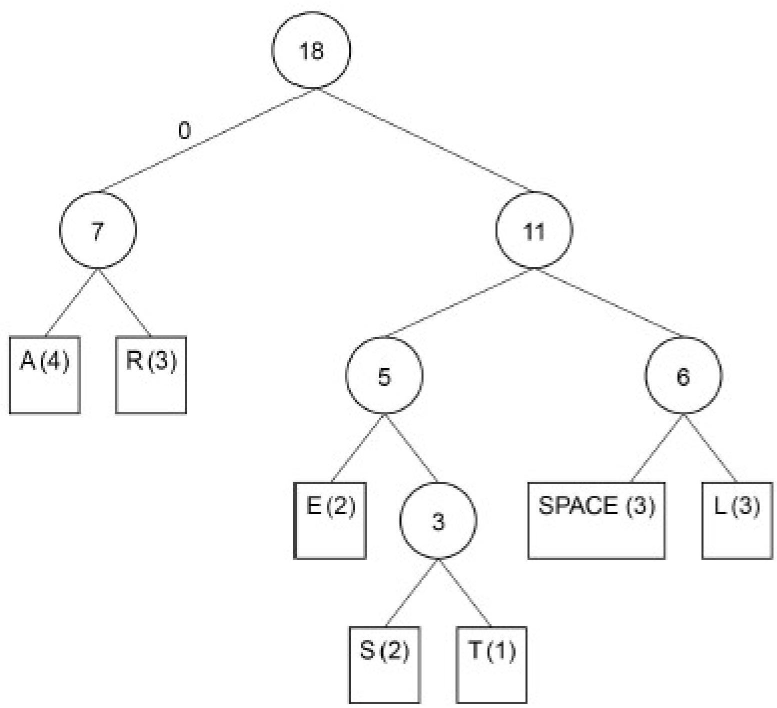
(Total 2 marks)

56. State **one** advantage of using Unicode instead of using ASCII.

(Total 1 mark)

57. The Huffman tree in **Figure 1** was generated for the string `ARE ALL STARS REAL`

Figure 1



(a) Part of the string `ARE ALL STARS REAL` was incorrectly encoded as in **Figure 2** below.

Figure 2

`1111000010101011`

What string does this encoding represent?

(1)

(b) What would be the correct binary encoding for the substring `STAR`?

Write the correct encoding below the letters in the table.

S	T	A	R

(2)
(Total 3 marks)

58. A bit pattern is shown below.

`01001110`

(a) Convert the bit pattern into decimal.

(1)

(b) Convert the bit pattern into hexadecimal.

(2)
(Total 3 marks)

59. A student's answer to the question "Why is hexadecimal often used instead of binary?" is shown below.

Because it uses fewer digits it will take up less space in a computer's memory.

Explain why the student's answer is incorrect.

(Total 2 marks)

60. Explain how a binary number can be multiplied by 8 by shifting bits.

(Total 2 marks)

Mark schemes

1. **Mark is for AO2 (apply)**

212;

[1]

2. **2 marks for AO2 (apply)**

B; (This must be the left digit to gain the mark)
9; (This must be the right digit to gain the mark)

[2]

3. **Mark is for AO2 (apply)**

63; // $2^6 - 1$;

[1]

4. **2 marks for AO2 (apply)**

`1110;   1001;`

[2]

5. **Mark is for AO2 (apply)**

`00010101;`

[1]

6. **Mark is for AO1 (understanding)**

The (binary) number is multiplied by 16 // 2^4 ;

Note for examiners: equivalent arithmetic methods should be credited.

[1]

7. **Mark is for AO1 (understanding)**

To divide by 4 // quarter the number // multiplies by 8 then divides by 32;

Note for examiners: equivalent arithmetic methods should be credited.

[1]

8. **2 marks for AO2 (apply)**

16 000;;

Maximum of **one** mark (if not fully correct) from:

- multiplying by 1000 **R.** if 1024 used in place of 1000
- multiplying by 8
- multiplying by 2

[2]

9. **2 marks for AO2 (apply)**

1 mark for each complete column with the correct values as below;

Character	ASCII value	Unicode value
`w`	119	119 // 77
`x`	120	120 // 78
`y`	121	121 // 79
`z`	122	122 // 7A

Note for examiner: award **one mark** for the Unicode column matching the ASCII column, even if incorrect values **R.** if any missing values

R. Binary values in the Unicode column

[2]

10. **3 marks for AO2 (apply)**

20;;;

Maximum of **two** marks (if not fully correct) from:

- multiplying 8 x 10 (even if result is incorrect) // 80 shown in working;
- multiplying by 2 // colour depth is 2;
- dividing by 8;

[3]

11. (a) **Mark is for AO1 (recall)**

(Sample resolution is the number of) bits per sample;

1

(b) **Mark is for AO1 (understanding)**

Maximum of **one** mark from:

- larger file size/takes up more storage space;
- sound files will take longer to download/transmit;
- uses more memory/processing power (when recording);

1

(c) **2 marks for AO2 (apply)**

4;;

Maximum of **one** mark (if not fully correct) from:

- multiplying the three values (50, 40 000 and 2) together (even if result is incorrect);
- division by 1000;
- division by 1 000 000;

2

[4]

12. **2 marks for AO1 (understanding)**

Maximum of **two** marks from:

- to reduce file size // to decrease the amount of storage needed; **NE.** Take up less space
- to make it faster to transmit/receive/read/write/access;
- (it could) save money if you use less bandwidth // (it could) save money if you use less storage capacity;
- to use less memory; **NE.** Take up less space

[2]

13. **2 marks for AO2 (apply)**

Maximum **two** marks from:

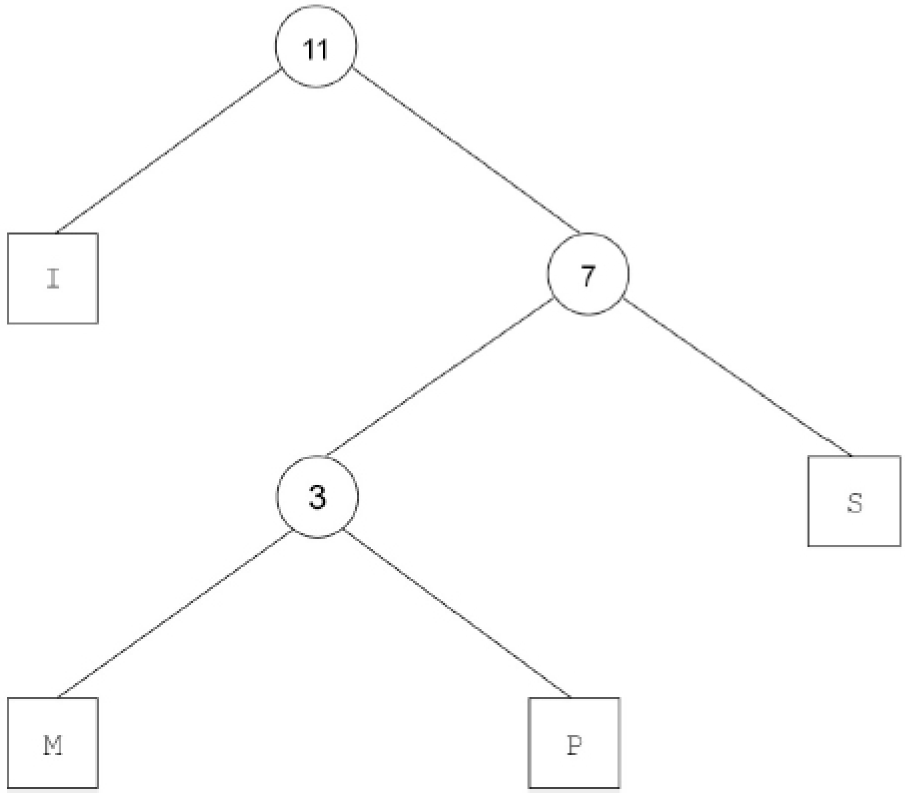
- The data do not have a high frequency of consecutive repeating characters;
- The (compressed) data will take up more storage space/be longer than the original/uncompressed data;

2 marks if a valid point is made along with a suitable valid expansion

[2]

14. **Mark is for AO2 (apply)**

One mark for **all characters** in the correct position.



[1]

15. **Mark is for AO1 (understanding)**

B Unicode includes characters from many different alphabets.;

R. if more than one lozenge is shaded.

[1]

16. **2 marks for AO2**

1 mark for each correction:

(Statement 2)
A 2 pixel by 4 pixel bitmap image contains 8 pixels //

A *n* pixel by *m* pixel bitmap image contains 16 pixels *[where n*m=16]*;

(Statement 4)
Black and white images have a minimum colour depth of one //
Three/Four-colour images have a minimum colour depth of two;

A. Explanation of error that makes it clear what should have been written instead of the corrected statement.

[2]

17. **Mark is for AO2**

300 (b // bits);

[1]

18. **3 marks for AO2**

50 (B // bytes);;;

If incorrect answer then award a **max of two** marks for the following working:

identifying the colour depth is 4;
correctly multiplying 10 x 10 x (possibly incorrect) colour depth;
attempt at dividing the calculated size in bits by 8;

[3]

19. **3 marks for AO3 (programming)**

1 mark for 1 correct position;
2 marks for 2 correct positions;
3 marks for 4 correct positions;

R. Any position which is used more than once

Line of code	Position (1–4 where 1 is the first line)
`t ← t - 1`	3
`n ← t - n`	4
`n ← 2`	1
`t ← n LEFTSHIFT 3`	2

[3]

20. **Mark is for AO2 (apply)**

11011100;

[1]

21. **2 marks for AO2 (apply)**

10101101;;

If the binary answer given is incorrect then award a maximum of **one** working mark as follows:

- A converted to 1010
- D converted to 1101
- both hexadecimal digits converted correctly to decimal, ie A=10, D=13

[2]

22. **Mark is for AO2 (apply)**

26;

[1]

23. **Mark is for AO2 (apply)**

FF;

[1]

24. **Mark is for AO1 (understanding)**

D Hexadecimal is more compact when displayed on screen;

R. If more than one lozenge shaded.

[1]

25. **Mark is for AO2 (apply)**

00101100;

A. Any number of leading zeros including none, eg 101100.

[1]

26. **2 marks for AO2 (apply)**

1110 1011;;

If the binary answer given is incorrect then award a maximum of **one** working mark as follows:

- left-hand side 4 bits are correct, ie 1110
- right-hand side 4 bits are correct, ie 1011

[2]

27. (a) **4 marks for AO2 (apply)**

6;;;;

If the answer given is incorrect then award a maximum of **three** working marks as follows:

- converting 8 minutes to 480 (seconds);
- 100 000 // 25 000 x 4;
- divide by 8;
- divide by 1 million;

4

(b) **2 marks for AO1 (understanding)**

One mark for each of the following points:

- the file size would increase;
- (the more samples per second) the more accurate / (the higher the sampling rate) the truer (higher quality) the (recorded) sound;

2

[6]

28. (a) **Mark is for AO2 (apply)**

B11 W12 B17;

1

(b) **2 marks is for AO2 (apply)**

00001111 10001001;;

If answer is not correct, **one** working mark may be given as follows:

- the first bit values (for colour) are both correct (0 and 1);
- the frequency values are both correct (0001111 and 0001001);

2

[3]

29.

3 marks for AO2 (apply)

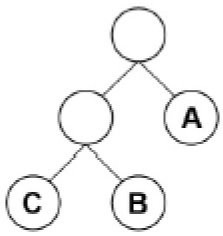
3 marks if solution is fully correct;;;

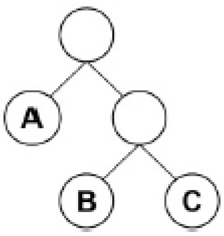
If solution is not fully correct then mark as follows:

- If all three letters shown then award 1 mark for each letter in correct relative location (to a maximum of **two** marks)
- Award 1 mark if only two letters shown but they are in the correct relative locations

I. Frequency totals written inside nodes.

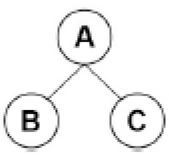
Correct solutions

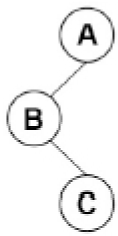




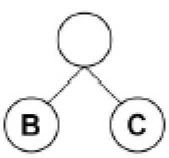
3 marks;;;

Partially correct solutions

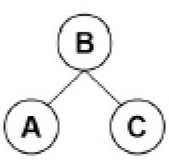




2 marks (all three letters shown with two letters in correct relative location);;



1 mark (only two letters shown in correct relative location);



1 mark (all three letters shown but only 1 letter (C) in correct relative location);

[3]

30. **3 marks for AO2 (apply)**

43;;;

If incorrect answer then a maximum of **two** working out marks may be awarded as follows:

- calculate ASCII bits:
  9 characters x 7 bits // 63 (bits);
- Huffman bits:
  (3 x 2) + (2 x 2) + (2 x 2) + (1 x 3) + (1 x 3) /
  (7 x 2) + (2 x 3) //
  20 (bits);

- correctly subtracting Huffman bit total from ASCII bit total;
- Huffman code written in full (10 11 01 000 10 11 10 01 001);

[3]

31. **Mark is for AO1 (recall)**

(A pixel is a) single point (of colour) in an image/smallest (addressable) part of an image;

A. picture element

A. alternatives to the word point eg dot, element 1

[1]

32. **Mark is for AO2 (apply)**

64 // 2^6;

[1]

33. **3 marks for AO2 (apply)**

500/500kB/500 kilobytes;;;

If incorrect answer is given then award a maximum of 2 marks for working as follows:

indicating the colour depth is 5;
multiplying (800x1000) by the colour depth (even if colour depth is incorrect);
correct conversion from bits to kilobytes;

[3]

34. (a) **2 marks for AO2 (apply)**

The first value of result 16;
The last value of result 12;

Max 1 mark if more than two values are given for result.

The correct table is as follows:

result
16
12

2

(b) **2 marks for AO2 (apply)**

The `x` column fully correct;
The `result` column fully correct;

If more values are given in any column then max 1 mark.

The correct table is as follows: x result

x	result
	0
1	4
2	8
3	12

I. horizontal alignment of values as long as the vertical order of values is correct.

2

(c) **Mark is for AO2 (apply)**

(The purpose of the algorithms is) to multiply the value in `number` by 3;

A. the value 4 instead of `number`.
NE. multiply two numbers.

1

(d) **Mark is for AO2 (apply)**

The algorithm in **Figure 1** uses fewer steps/instructions;

A. the algorithm in **Figure 1** uses fewer variables;
A. the algorithm in **Figure 1** has fewer instructions so will take up less memory;
A. the algorithm in **Figure 1** will execute in less time;
A. opposite statements for **Figure 2**.
NE. reference to number of lines.

1

[6]

35. **Mark is for AO2 (apply)**

94;

[1]

36. **Mark is for AO2 (apply)**

148;

[1]

37. **2 marks for AO2 (apply)**

8F;;

If the answer given is not 8F then award a **maximum of 1 working mark** for any of the following:

- converted 143 to `10001111;`
- converted 143 to an incorrect 8-bit binary number but converted this correctly to hexadecimal;
- attempted division of 143 by 16 to get a quotient of 8 and a remainder of 15 but incorrectly represented this in hexadecimal;
- either the 8 or the F are present anywhere within the answer;

[2]

38. **2 marks for AO2 (apply)**

`10111110;;`

If the answer given is not `10111110` then award a **maximum of 1 working mark** for any of the following:

- converted B to 1011;
- converted E to 1110;
- converted BE to 190 and then incorrectly converted this value to binary;

[2]

39. **2 marks for AO1 (understanding)**

A **maximum of 2 marks** can be awarded.

Examples include:

- hexadecimal is easier (for humans) to read (than binary); **A.** easier to understand
- numbers are displayed in a more compact way (in hexadecimal than in binary);
- it is quicker to type in (hexadecimal numbers than binary numbers);
- it reduces the risk of typing errors (hexadecimal numbers than binary numbers);

R. individual points that imply less memory is used.

[2]

40.

2 marks for AO2 (apply)

Marks should be awarded as follows:

- the right hand side 2 bits are correct (10);
- the answer is fully correct (11011010);

[2]

41.

Mark is for AO2 (apply)

One mark for the correct answer only;

1	1	1	0	0	1	0	0

[1]

42.

Mark is for AO2 (apply)

63 // 2^6 -1;

[1]

43.

2 marks for AO2 (apply)

40 000;;

If incorrect answer is given then **maximum of 1 mark** for working:

- 5000;
- multiplying by 8;
- multiplying by 1000;

[2]

44.

3 marks for AO1 (understanding)

A **maximum of 3 marks** can be awarded.

Example mark points include:

- a microphone/sound sensor picks up/detects the sound wave;
- this wave is converted to an (electrical) analogue signal;
- the amplitude/height of the wave is measured;
- the sampling takes place at regular intervals;
- these samples are stored as binary values;

Note: award one mark for “analogue (signal) converted to digital” if none of the last three example mark points awarded

[3]

45. **2 marks for AO2 (apply)**

1320 (kB);;

If the answer given is not 1320 kilobytes, a **maximum of 1 working mark** should be awarded as follows:

1. Multiplying the correct 3 values together (44 000 * 30 * 8 = 10,560,000 bits) even if the result is incorrect;

2. Dividing the result of a multiplication by 8 (even if the result is incorrect);

3. Correctly dividing the result of a calculation by 1000;

[2]

46. **Mark is for AO1 (recall)**

C The process of encoding information to try and use fewer bits than the original;

R. if more than one lozenge shaded.

[1]

47. **2 marks for AO1 (understanding)**

A **maximum of 2 marks** can be awarded.

Example mark points include:

- To speed up file transfer;
- To use less storage (capacity);
- (It could) save money if you use less bandwidth;
- To use less memory;
- (It could) save money if you use less storage capacity;

[2]

48. **Mark is for AO1 (understanding)**

(The data) contains lots of runs;

A. repeating values

[1]

49. (a) **3 marks for AO2 (apply)**

1 mark for any one character correctly encoded
2 marks for any three characters correctly encoded
3 marks for all characters correctly encoded

M	O	S	T
10111	10110	010	111

3

(b) **4 marks for AO1 (understanding)**

1 mark for each of the following:

- the characters with the highest frequencies should not be combined // the characters with the lowest frequencies should be combined;
- the frequency table should not be sorted alphabetically // the frequency table should be sorted in order of frequency;
- the letter S does not have the highest frequency in Figure 2 // E has the highest frequency;
- the new node should not be added to the end of the frequency table // the new node should be inserted in the correct place based on the combined frequencies;

4

(c) **2 marks for AO2 (apply)**

122 (bits);;

If the answer given is not 122 bits, a **maximum of 1 working out mark** should be awarded as follows:

<u>ASCII Calculation</u>
36 characters * 7 bits // 252 (bits);

<u>Calculate Saving</u>
Subtracting 130 from any other value;

2

[9]

50. (a) **Mark is for AO2 (apply)**

D 4;
If more than one lozenge shaded then mark is not awarded

1

(b) **Mark is for AO2 (apply)**

D '`computer sciencegcse`';
If more than one lozenge shaded then mark is not awarded

1

(c) **Mark is for AO2 (apply)**

C '`sci`';
If more than one lozenge shaded then mark is not awarded

1

(d) **Mark is for AO2 (apply)**

C 101;
If more than one lozenge shaded then mark is not awarded

1

[4]

51. (a) **Mark is for AO2 (apply)**

D `USERINPUT`;
If more than one lozenge shaded then mark is not awarded

1

(b) **Mark is for AO2 (apply)**

B `0`;
If more than one lozenge shaded then mark is not awarded

1

(c) **Mark is for AO2 (apply)**

A `=` ;
If more than one lozenge shaded then mark is not awarded

1

(d) **Mark is for AO2 (apply)**

D `OUTPUT count`;
If more than one lozenge shaded then mark is not awarded

1

(e) **Mark is for AO2 (apply)**

B `i ⟵ i + 1`;
If more than one lozenge shaded then mark is not awarded

1

(f) **2 marks for AO2 (apply)**

Maximum of 1 mark if Upper Case Characters given

- 1 mark for a series of more than one correct frequency/value or value/frequency pairs (ignore order of pairs);
- 1 mark for all correct pairs in the correct order;

Correct answer is:
2 t 2 j 3 e 2 s

Other, clear ways to show frequency/value or value/frequency pairs such as ‘(2, t), (2, j),…’ or ‘t2 j2…’.

2

(g) **3 marks for AO2 (apply)**

Maximum three marks from:

- It could be tested with only 1s;
- It could be tested with different lengths of input;
- It could be tested with an input where the 1s and 0s vary;
- It could be tested with an input where the last two numbers are different;
- It could be tested with the empty string;
- It could be tested with a string of length one;
- It could be tested with two runs of 0s separated by a run of 1s / two runs of 1s separated by a run of 0s;
- It could be tested with invalid data (such as 1010abc);

Any other correct reasoning as long as clearly distinct from other mark points.

R. not enough tests are carried out.

3

[10]

52. **1 mark for AO2 (apply)**

1100 0101;

[1]

53. **2 marks for AO2 (apply)**

164;

If incorrect answer is given then maximum of 1 mark for working.

- determining that A is worth 10 irrespective of it being in the correct column (place value);
- multiplying an incorrect conversion of A by 16;
- converting to binary to give 1010 0100;

[2]

54. **1 mark for AO2 (apply)**

31 // 2^5-1;

[1]

55. **2 marks for AO2 (apply)**

24 000 000;

If incorrect answer is given then maximum of 1 mark for working.

- 3 000 000//3*1000*1000 to calculate the correct number of bytes;
- Multiplying an incorrect number of bytes by 8;
- 3 000 000 * 8 with incorrect result;

[2]

56. **1 mark for AO1 (understanding)**

- To be able to represent additional / more characters / more languages (and symbols not available in the ASCII character set);
- ASCII only allows 128 characters whereas Unicode can represent more;
- To represent characters from other alphabets;

A. A response that says a specific single character can be shown i.e. “a playing card character can be shown”;
A. To represent non-English languages;

[1]

57. (a) **1 mark for AO2 (apply)**

`LEAST`;

A. Any text sentence such as ‘the string represents the word LEAST’;

I. Upper / lower case.

1

(b) **2 marks for AO2 (apply)**

S	T	A	R
1010	1011	00	01

1010 1011 00 01;

2 marks for **all four** codes correct ;
1 mark for **any two** codes correct ;

A. Clearly written codes that are correct as shown above, even if they are not written in the table.

2

[3]

58. (a) **Mark is for AO2 (apply)**

78;

1

(b) **All marks AO2 (apply)**

4; (This must be the left hand digit to gain the mark)
E; (This must be the right hand digit to gain the mark)

Maximum 1 mark: if final answer not correct.

2

[3]

59. **All marks AO1 (understanding)**

(The answer is incorrect because) the number will (still) be represented using binary in a computer's memory;
so it will take up the same amount of memory space;

[2]

60. **All marks AO1 (understanding)**

(Shifting the bit pattern) three places; to the left;

Mark as follows:
1 mark: for correct direction of shift
1 mark: for correct number of times to shift

[2]