COMPUTER SCIENCE

UNIT 1 & UNIT 3 (PAST PAPER)

DR. SAW MYAT SANDAR
PEARSON EDEXCEL INTERNATIONAL GCSES (O LEVEL) |

Sample Question

Wasim has designed an algorithm to convert any measurement made in centimetres into inches and vice versa.

```
1 SEND 'Which units are you entering/ enter "I" for inches
  and "C" for centimetres.' TO DISPLAY

2 RECEIVE choice FROM (STRING) KEYBOARD

3 SEND 'Please enter the measurement as a whole number' TO
  DISPLAY

4 RECEIVE valueToConvert (INTEGER) FROM KEYBOARD

5 IF choice = 'I' THEN

6       SET conversion TO valueToConvert * 0.39

7 ELSE

8       SET conversion TO valueToConvert * 2.54

9 END IF

10 SEND conversion TO DISPLAY
```

(a) The variable 'valueToConvert' has been typecast as a STRING. Identify two other variables and state the data type of each.

1 ....

Data type..

2 ....

Data type.. **(4)**

(b) When Wasim asks his friends to test the program it will carry out a calculation even if an entry other than 'I' or 'C' is made. Edit the algorithm by adding lines before line 2 to authenticate the user entry for the units they wish to convert.

....

....

....

....

....

....

....

....

....

....

....

.... **(6)**

7 (a) 1) choice – string

2) conversion – real

(b) This is one possible solution.

```
SET acceptable TO False
WHILE acceptable = False DO
    SEND 'Which units are you entering/ enter "I"
for inches and "C" for centimetres.' TO DISPLAY
    RECEIVE choice FROM (STRING) KEYBOARD
    IF choice = 'True' OR choice = 'C' THEN
        SET acceptable TO true
    ELSE
        SEND 'Sorry that is not recognised.' TO DISPLAY
    END IF
END WHILE
```

```
SET acceptable TO False

WHILE acceptable = False DO

        SEND 'Which units are you entering/ enter "I" for inches and "C"
        for centimetres.' TO DISPLAY

        RECEIVE choice FROM (STRING) KEYBOARD

        IF choice= "I" OR choice= "C" THEN

            SET acceptable TO True

        ELSE

            SEND "Sorry" TO DISPLAY

        END IF

END WHILE
```

2017

7 Algorithms can be designed using pseudocode or flowcharts. Then, they need to be translated into code that a computing device can execute.
Figure 2 shows the pseudocode for an algorithm.

```
 1  # This is the pseudocode for an algorithm
 2  SET inNum TO 0
 3  SET result TO 1
 4  SET i TO 0
 5
 6  SEND "Enter a number: " TO DISPLAY
 7  RECEIVE inNum FROM (INTEGER) KEYBOARD
 8
 9  IF (inNum < 0) THEN
10      SEND "Invalid input" TO DISPLAY
11  ELSE
12      IF (inNum = 0) THEN
13          SEND "Answer is 1" TO DISPLAY
14      ELSE
15          FOR i FROM 1 TO inNum DO
16              SET result TO result * i
17          END FOR
18          SEND "The answer is " & result TO DISPLAY
19      ENDIF
20  ENDIF
```

Figure 2

(a) Use the information in Figure 2 to answer these questions.
(i) Complete the table to show the output for the given input.
(3)

Input	Output message
0	
-12	
5	

Input	Output message	
0	Answer is 1	(1)
-12	Invalid input	(1)
5	The answer is 120	(1)

(ii) State the purpose of this algorithm.
(1)

multiplies every number between 1 and inNum.
calculates factorial of inNum.

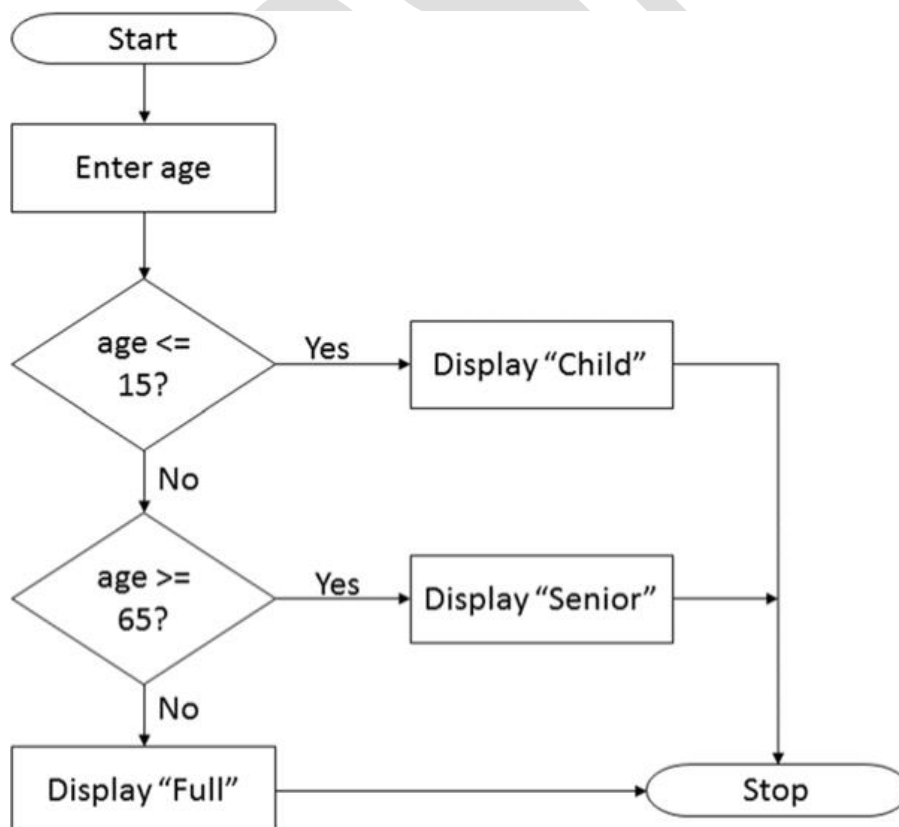(b) A bus company sets fares for different groups of passengers.
The fares are:
• a child fare for passengers 15 years old and younger
• a senior fare for passengers 65 years old and older
• a full fare for all other passengers.
Construct a flowchart of an algorithm that will determine the fare for one
passenger when an age is input.
No validation of input is required.
(5)

```
        Start
          |
          v
    +-----------+
    | Enter age |
    +-----------+
          |
          v
       /age <=\    Yes
      < 15?    >------>  Display "Child" ------+
       \      /                                |
          | No                                 |
          v                                    |
       /age >=\    Yes                         |
      < 65?    >------>  Display "Senior" ----->|
       \      /                                |
          | No                                 |
          v                                    |
    +-------------+                            v
    |Display "Full"|------------------------>  Stop
    +-------------+
```

⊐ᒋⵥᒐⵥⴺ<ᒐ>ⵥᒐⵔᴑⴺᒋᴦ

(c) Users are forced to change their passwords every 28 days. This requires an algorithm that reports the days in any given month.

The algorithm must report the number of days in a month based on a number entered (e.g. 1 = January, 2 = February etc.).

This pseudocode algorithm does not produce accurate results. These are the test results.

Input	Expected behaviour	Actual behaviour
2	The month is February and it has 28 days.	The month is March and it has 31 days.
13	The month number 13 is not valid.	Potential runtime error: index out of range.
-4	The month number -4 is not valid.	Potential runtime error: index out of range.

Figure 3 shows the errors are on lines 12, 13, and 14.

```
1
2  SET monthNames TO ["January", "February", "March", "April",
3                     "May", "June", "July", "August", "September",
4                     "October", "November", "December"]
5
6  SET monthDays TO [31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31]
7
8  SEND "Enter the month number.  0 to exit." TO DISPLAY
9  RECEIVE number FROM (INTEGER) KEYBOARD
10
11 WHILE NOT (number = 0) DO
12     IF (number > 1) OR (number < 12) THEN
13         SET month TO monthNames[number]
14         SET days TO monthDays[number]
15
16         SEND "The month is " & month & " and it has " & days & "days."
17     ELSE
18         SEND "The month number: " & number & " is invalid."
19     ENDIF
20
21     SEND "Enter the month number.  0 to exit." TO DISPLAY
22     RECEIVE number FROM (INTEGER) KEYBOARD
23
24 END WHILE
```

Figure 3 Write the corrected replacement codes for lines 12, 13, and 14.   (4)

Line 12 ....

Line 13 ....

Line 14 ....

(number >= 1)   (1)

Boolean operator AND   (1)

(number <= 12)   (1)

Both instances of [number -1]   (1)

```
12         IF (number >= 1) AND (number <= 12) THEN
13             SET month TO monthNames[number - 1]
14             SET days TO monthDays[number - 1]
15
```

2018

5 Ships carry cargo around the world in containers.
(a) Containers come in two sizes.
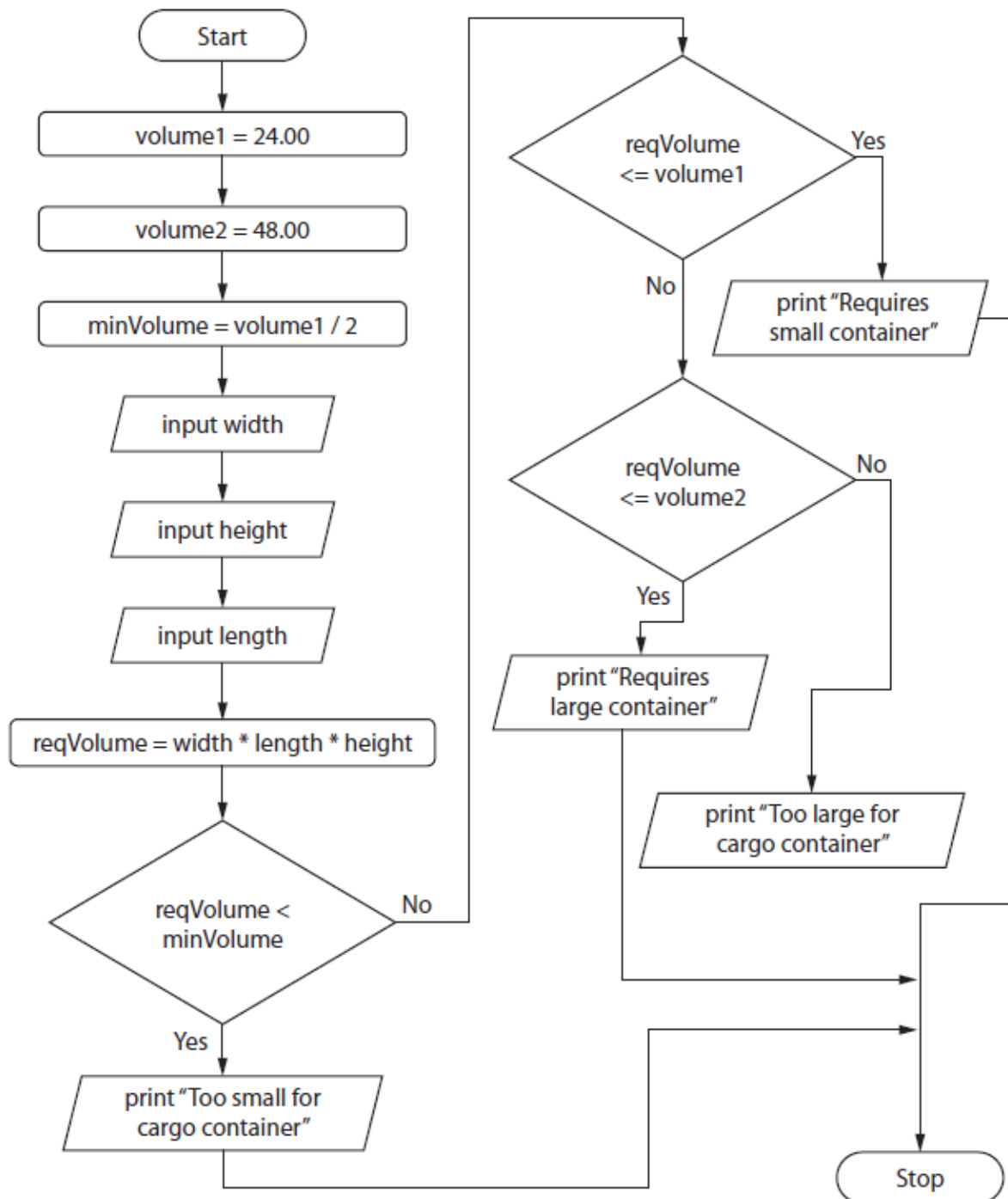**Figure 2** shows an algorithm written using flowchart symbols.

Figure 2

(i) Complete the table to show the output for each cargo item.

(3)

Cargo item			Output
width	length	height	
4	4	2	
2	2	2	
3	8	5	

(ii) State the purpose of the algorithm in **Figure 2**.

(1)

....

Width	Length	Height	Output
4	4	2	Requires large container (1)
2	2	2	Too small for cargo container (1)
3	8	5	Too large for cargo container (1)

(b) Each ship is registered to a state.
Figure 3 shows a partially completed algorithm written in pseudocode.
The completed algorithm must:
• print each state to the display on a new line
• count the number of states
• create a message as a single string (e.g. there are *number* states)
• print the message to the display.
Complete the algorithm in the space provided in **Figure 3**.

(4)

```
SET numStates TO 0

SET states TO ["France", "Singapore", "Malta", "Panama", "Greece", "Italy"]




FOR EACH state FROM states DO










END FOREACH
```

Figure 3

```
SET numStates TO 0
SET states TO ["France", "Singapore",
               "Malta", "Panama",
               "Greece", "Italy"]
FOR EACH state FROM states DO
    SEND state TO DISPLAY
    SET numStates TO numStates + 1
END FOREACH
SEND "There are " & numStates & "states" TO DISPLAY
```

SEND states [numStates] TO DISPLAY (1)

or

SEND state TO DISPLAY (1)

SET numStates TO numStates + 1 (1)

or

LENGTH(states) (1)

Concatenation of message and variable uses "
and &

& numStates & TO DISPLAY (1)

(c) Cargo ships have maximum weight loads.
Figure 4 shows an algorithm written using pseudocode.
The algorithm should identify the size of cargo ship required for any load.
There is an error on line 9.

```
2    SET loadWeight TO [20000, 28000, 40000, 50000]
3    SET index TO 0
4    SET found TO FALSE
5
6    SEND "Enter cargo weight" TO DISPLAY
7    RECEIVE target FROM (INTEGER) KEYBOARD
8
9    WHILE (NOT found) DO
10        IF (loadWeight [index] >= target) THEN
11            SEND loadWeight [index] TO DISPLAY
12            SET found TO TRUE
13        ELSE
14            SET index TO  index + 1
15        END IF
16    END WHILE
17
18    IF (NOT found) THEN
19        SEND "No ship available" TO DISPLAY
20    END IF
```

Figure 4

(i) Trace tables are used to identify errors in algorithms.
Complete the trace table for an input of 50500 to show what happens due to
the error on line 9 in the pseudocode in **Figure 4**.
You may not need to fill in all the rows in the table.

(2)

target	found	index	loadWeight[index]
50500	FALSE	0	

(ii) Construct a single line of pseudocode to correct line 9.
(2)

....

target	found	index	loadWeight[index]
50500	FALSE	0	20000
		1	28000
		2	40000
		3	50000
		4	

WHILE ((NOT found) AND (index < LENGTH
(loadWeight))) DO

2020

6 A software engineer is working with some algorithms.

(a) An algorithm needs to be developed.

The algorithm must:

• check for a change of hour every minute

• use the library subprogram getHour() to get the hour part of the current time in the 24-hour clock (0 to 23)

• output "Good morning" when the hour is between 3 and 12, inclusive

• output "Good afternoon" when the hour is between 13 and 19, inclusive

• output "Good night" at all other times.

Complete the flowchart to represent this algorithm, in the space provided on the next page.

The call to the library subprogram getHour() has already been included.

Use this subprogram symbol to show waiting for 1 minute.

```
| wait     |
| 1 minute |
```
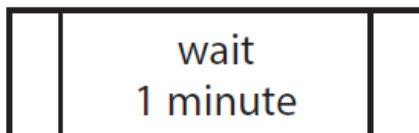
(6)

Draw your flowchart here.

```
( START )
    |
    v
| getHour() |
```
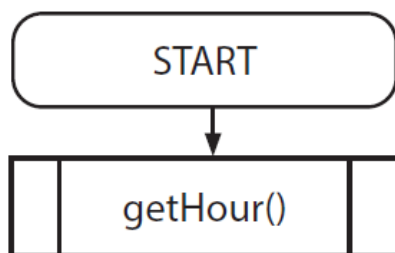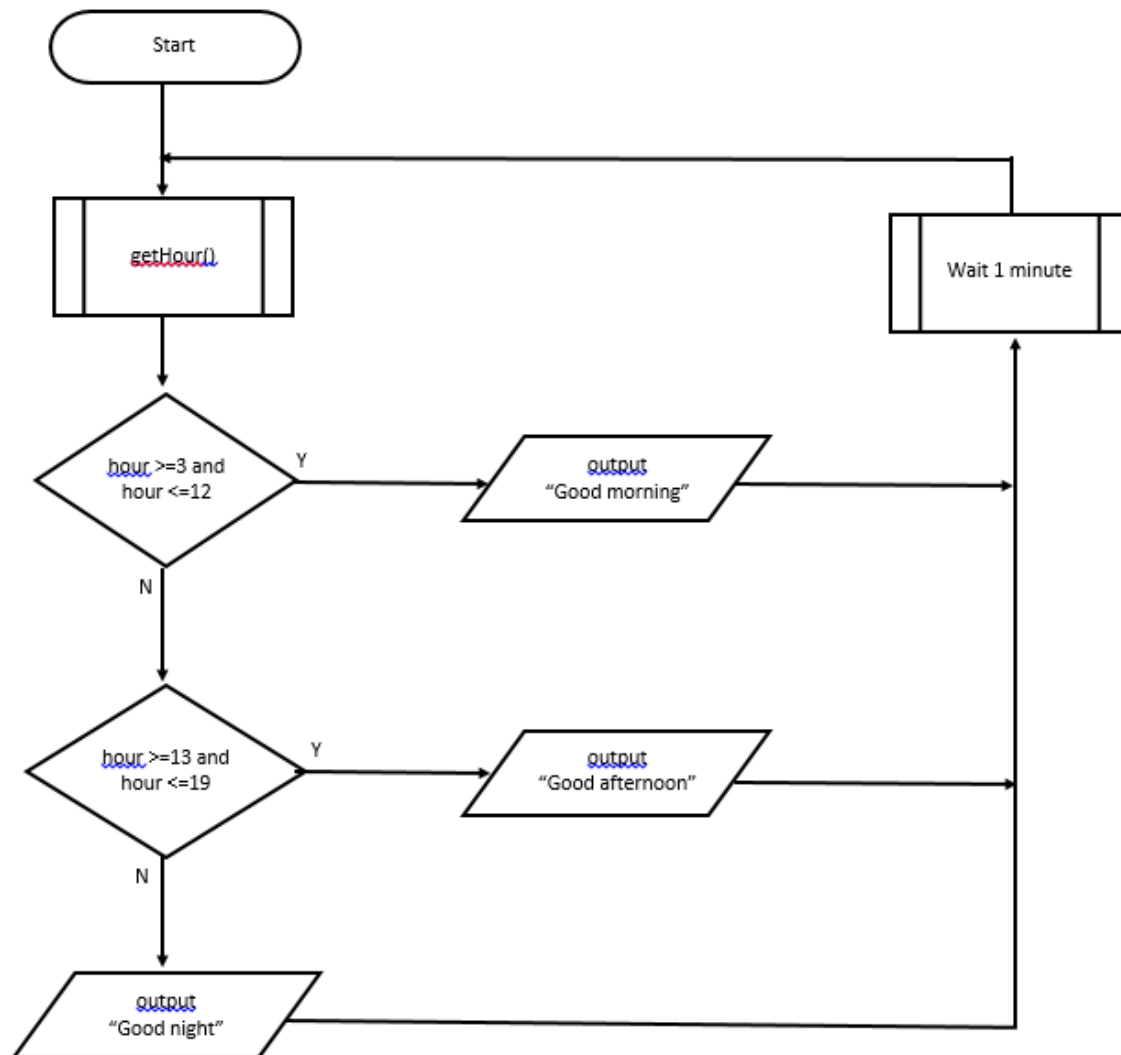
(b) Another algorithm determines when to change the flavoured syrups for a drinks dispensing machine.
Figure 4 shows this algorithm written in pseudocode.

```
 1  SET flavours TO ["anise", "mango", "cola", "apple", "papaya", "strawberry",
    "lychee", "banana"]
 2
 3  SET volume TO [0.7, 0.2, 0.6, 0.1, 0.05, 0.8, 0.4, 0.6]
 4
 5  SET i TO 0
 6
 7  WHILE i < 8 DO
 8
 9      IF (volume[i] < 0.1) THEN
10          SEND (flavours[i] & " needs changing") TO DISPLAY
11
12      ELSE
13          IF (volume[i] >= 0.3) AND (volume[i] <= 0.5) THEN
14              SEND (flavours[i] & " needs ordering")  TO DISPLAY
15          END IF
16      END IF
17
18      SET i TO i + 1
19
20  END WHILE
```

(i) Give the output produced by the algorithm.
(2)
....

The algorithm works with a fixed number of flavours.
(ii) Give the number of the line in the algorithm that would need to be amended
to allow for any number of flavours.
(1)
....

(iii) State how the pseudocode needs to be changed to make this amendment.
(1)
....

papaya needs changing (1)
lychee needs ordering (1)

7

WHILE i < LENGTH(flavours) DO (1)

2021

5 Isaac is a program developer.
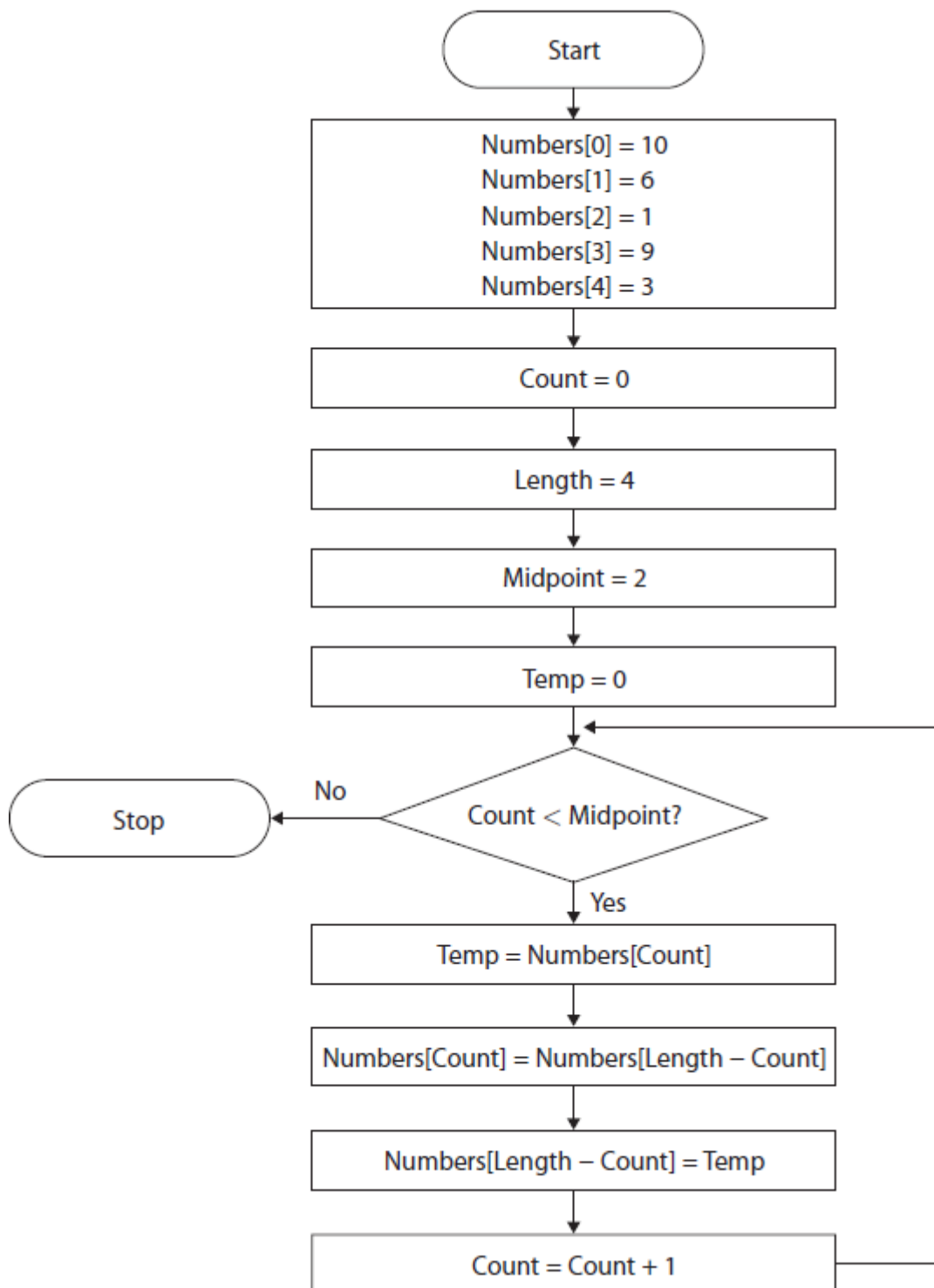
(a) **Figure 3** shows an algorithm Isaac has written.

Figure 3

(i) Complete the trace table. You may not need to use all of the rows.

(5)

2021

Count	Length	Midpoint	Temp	Numbers array				
				[0]	[1]	[2]	[3]	[4]
0	4	2	0	10	6	1	9	3

Count	Length	Midpoint	Temp	Numbers array				
				0	1	2	3	4
0	4	2	0	10	6	1	9	3
1			10	3				10
2			6		9		6	

(ii) Give the purpose of the algorithm.
(1)

To reverse the contents of the array

....

(iii) Explain why the variable Temp is needed.
(2)

You need to swap the contents of array values (1) and without Temp one of the values would be lost (2)

....

(b) **Figure 4** shows an algorithm Isaac has written using pseudocode.
The algorithm should display the average of the numbers that have been input.

```
1      SET total TO 0
2      SET number TO 0
3      SET count TO 0
4      WHILE number <> −1 DO
5         SEND 'Input a number or −1 to end the program' TO DISPLAY
6         RECEIVE number FROM (INTEGER) KEYBOARD
7         SET total TO total + number
8         SET count TO count + 1
9      END WHILE
10     SET average TO total / count
11     SEND 'The average is ' & average TO DISPLAY
```

Figure 4

Isaac uses the input 2, 3, 5, 2, −1 to test the algorithm. He discovers an error.

Expected result	Actual result
The average is 2.75	The average is 2.2

(i) Explain why the **Actual** result is not the same as the **Expected** result.
(2)

(ii) Give the number of the line that contains the error.
(1)

(iii) Amend a single line of pseudocode to correct the error.
(1)

| 5(b)(i)	Award **two** marks for a linked explanation such as: • Isaac has included the -1 as a number in the addition (1) but the number has not been added to the count (1) • Isaac has misunderstood the WHILE loop (1) as it should not execute after the -1 has been input (1) • Isaac is expecting the wrong result (1) it should be 3 (1) • The count is 1 too many as the -1 is counted as a number (1) and the total is incorrect as 1 is subtracted from the total (1)
5(b)(ii)	Line 3 (1) Line 10 (1)
5(b)(iii)	SET count TO -1 (1) SET average TO (total + 1) / (count − 1)

2022

4 Reba likes writing programs.

(a) She is writing a guessing game.

She needs a flowchart to show the logic of the game.

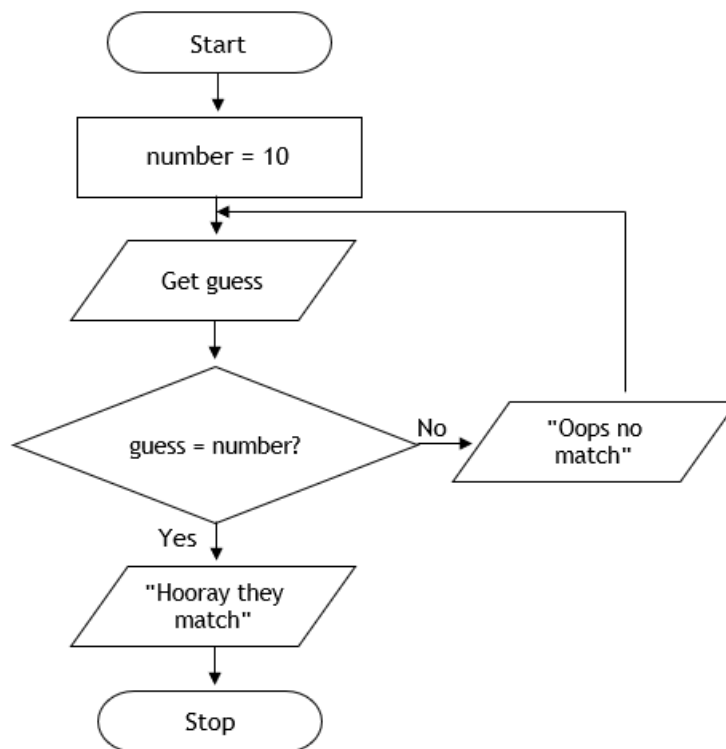(i) These are the components needed to draw the flowchart.
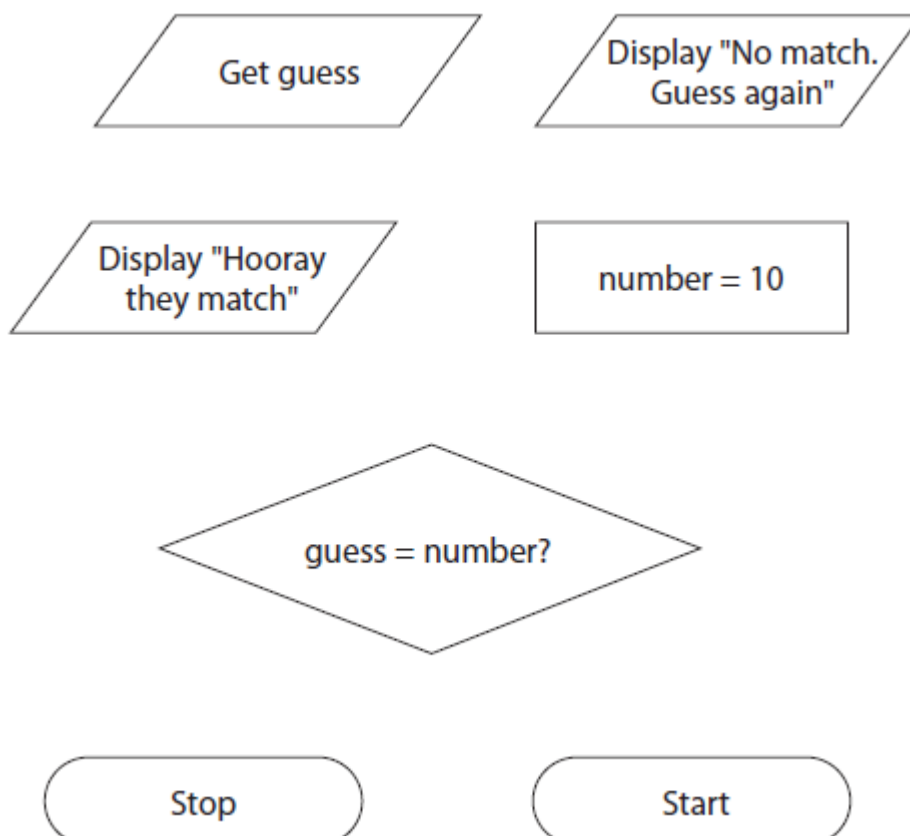
Draw the flowchart for the algorithm in the box on the next page.
Use each component once.
Do not add any additional components.
Use as many arrows and yes/no labels as you need.
(5)

(c) **Figure 2** shows the pseudocode for an early version of an algorithm that Reba
has written for another game.
The algorithm:
• asks the user to input a colour or input −1 to end the game
• awards 1 point for red
• awards 8 points for orange
• generates the score for the game
• displays the results of the game.

```
 1 SET Colour TO ""
 2 SET Score TO 0
 3 SET RedPoints TO 0
 4 SET OrangePoints TO 0
 5 SET NumOranges TO 0
 6
 7 WHILE Colour <> "-1" DO
 8     RECEIVE Colour FROM (STRING) KEYBOARD
 9     IF Colour = "red" THEN
10         SET RedPoints TO RedPoints + 1
11     ELSE
12         IF Colour = "orange" THEN
13             SET OrangePoints TO OrangePoints + 8
14             SET NumOranges TO NumOranges + 1
15         END IF
16     END IF
17 END WHILE
18
19 SET Score TO RedPoints + OrangePoints
20
21 SEND ("Score: "& Score) TO DISPLAY
22 SEND ("Number of reds: "& RedPoints) TO DISPLAY
23 SEND ("Number of oranges: "& OrangePoints) TO DISPLAY
```

Figure 2

Reba inputs: red, orange, red, red, orange, −1
The outputs are not as she expects.
(i) Complete the trace table to show the outputs.
(4)

Colour	Score	RedPoints	OrangePoints	NumOranges	Outputs
	0	0	0	0	
red					
orange					
red					
red					
orange					
−1					

(ii) Give the line number of the pseudocode that contains the error.
(1)

23

(iii) Write a replacement line of pseudocode to correct the error.
(1)

SEND ("Number of oranges: "& NumOranges) TO DISPLAY

Colour	Score	Red Points	Orange Points	Num Oranges	Outputs
	0	0	0	0	
red		1			
orange			8	1	
red		2			
red		3			
orange			16	2	
-1					
	19				
					Score: 19
					Number of reds: 3
					Number of oranges: 16

2023

6 Programmers share algorithms with different people and write algorithms for different reasons.
(a) A programmer is showing a new algorithm to a group of non-technical managers.
State an appropriate method for writing the algorithm.
Justify your answer.
(2)
Method
....
Justification
....

Method
- Flowchart (1)

Justification

- It is a visual representation / does not use many words / does not rely on use of English language (1)

- It does not rely on understanding specific syntax (1)

- Overview without unnecessary detail (1)

(b) **Figure 1** shows an algorithm that displays a string based on the number input by the user.

```
1  SEND ("Enter a number: ") TO DISPLAY
2  RECEIVE inNum FROM (INTEGER) KEYBOARD
3  IF ((inNum = 1) OR (inNum = 2)) THEN
4      IF (inNum = 1) THEN
5          SEND ("First") TO DISPLAY
6      ELSE
7          IF (inNum = 2) THEN
8              SEND ("Second") TO DISPLAY
9          END IF
10     END IF
11 ELSE
12     SEND ("Invalid input") TO DISPLAY
13 END IF
```

Figure 1

Give **one** reason why the selection statement on line 7 is not required.
(1)

(c) **Figure 2** shows an algorithm that manipulates arrays.

The algorithm works with any number of scores.

```
1 SET oldScores TO [10, 20, 30, 40, 50]
2 SET newScores TO [0, 0, 0, 0, 0]
3 SET newIndex TO 0
4
5 FOR oldIndex FROM (LENGTH (oldScores) - 1) TO 0 STEP -1 DO
6 SET newScores[newIndex] TO oldScores[oldIndex]
7 SET newIndex TO newIndex + 1
8 END FOR
```

Figure 2

(i) Describe what happens to the variable **oldIndex** when line 5 is executed.

(2)

(ii) State the purpose of the algorithm in **Figure 2**.

(1)

Question Number	Answer	Additional Guidance	Mark
6(b)	Award **one** mark for any of the following: • The test on line 3 has already limited the values to 1 and 2 (1) • The only value left for inNum to be (on line 7) is 2 (1)		1

Question Number	Answer	Additional Guidance	Mark
6(c)(i)	Award up to **two** marks for a linked description, such as: • The oldIndex value will go from highest to lowest (1) the step value is negative/decremented by 1 (each iteration) (1)	Allow 4 to 0 instead of highest and lowest	2

Question Number	Answer	Additional Guidance	Mark
6(c)(ii)	Award **one** mark for any of the following: • Reverse the (array of) scores (1) • Copy the oldScores into newScores, in reverse order (1)		1

2024 May Jun

6   Pseudocode and flowcharts are used to create algorithms.

(a) The number of minutes a train is early or late are recorded each day for a week.

- A value of 0 is recorded if the train is on time.

- A negative value is recorded if the train is early.

- A positive value is recorded if the train is late.

The algorithm in **Figure 2** has been created to calculate and output the total number of trains that were early, on time or late.

01	
02	`SET arrival TO [-2, 1, 5, 0, -3, 4, 1]`
03	
04	`SET early TO 0`
05	`SET late TO 0`
06	`SET index TO 0`
07	
08	`WHILE NOT (index > LENGTH(arrival)) DO`
09	
10	`    IF arrival[index] >= 0 THEN`
11	`        SET late TO arrival[index]`
12	`    ELSE`
13	`        IF arrival[index] < 0 THEN`
14	`            SET early TO early + 1`
15	`        END IF`
16	`    END IF`
17	
18	`    SET index TO index + 1`
19	
20	`END WHILE`
21	
22	`SEND "Trains early:   " & early TO DISPLAY`
23	`SEND "Trains on time: " & (late + early) TO DISPLAY`
24	`SEND "Trains late:   " & late TO DISPLAY`
25	

Figure 2

Identify the type of iteration construct used in **Figure 2**.

(1)

⊠ **A** Post-conditioned loop

⊠ **B** Count controlled loop

⊠ **C** Array loop

⊠ **D** Pre-conditioned loop

(b) There are **four** errors in the pseudocode in **Figure 2**.

Write the correct code for lines 08, 10, 11 and 23

(i) Line 08

(1)

....

....

(ii) Line 10

(1)

....

....

(iii) Line 11

(1)

....

....

(iv) Line 23

(1)

....

....

(c) Give **one** reason for the variable `index` being incremented in line 18.

(1)

....

....

Question Number	Answer	Additional Guidance
6(a)	**The only correct answer is D** **A** is not correct because it is not a REPEAT UNTIL / DO WHILE loop **B** is not correct because it is not a FOR loop **C** is not correct because there is no such loop	

6(b)(i)	Award **one** mark for any of the following:
	
	• `WHILE NOT (index >= LENGTH(arrival))`
	• `WHILE NOT (index > LENGTH(arrival)-1)`
	• `WHILE index < LENGTH(arrival)`
	• `WHILE index <= LENGTH(arrival)-1`
		Allow hard coded values for the number of items in the list.
		For example,
		`index >= 7`
		`index > 6`

6(b)(ii)	Award **one** mark for any of the following: • `IF arrival[index] > 0` • `IF arrival[index] >= 1`	
6(b)(iii)	Award **one** mark for: • `late` is incremented (1) `SET late TO late + 1`	

Question Number	Answer	Additional Guidance

6(b)(iv)	Award **one** mark for:
	
	• On time is calculated as the remainder when early and late trains are subtracted from the total number of trains (1)
	`LENGTH(arrival) - (late + early)`
		Allow number of trains to be hard coded as 7 in place of
		`LENGTH(arrival)`
		
		Allow equivalent expressions:
		`LENGTH(arrival) - late - early`

6(c)	Award **one** mark for any of the following: • Otherwise there would be an infinite loop (1) • It is used to index all elements in the (`arrival`) array (1) • To access the next value in the (`arrival`) array on the next iteration (1) • So that the condition used in the while loop evaluates to True when all the elements in arrival have been processed (1)	

(d) In the game FizzBuzz a user enters a number.

These conditions apply:

·   if the number is divisible by 3 the program displays Fizz

·   if the number is divisible by 5 the program displays Buzz

·   if the number is divisible by both 3 and 5 the program displays FizzBuzz

·   if the number is not divisible by either 3 or 5, the program displays the number entered.

Hint: If one number is divisible by another number there is no remainder.

These are the components needed for the flowchart.
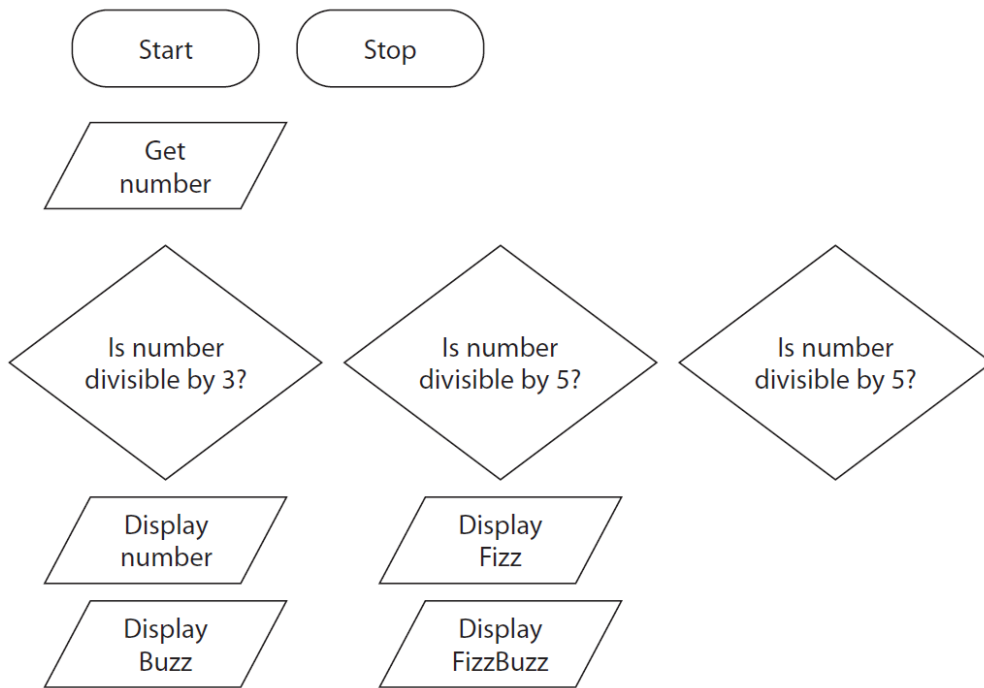
Draw the flowchart for the algorithm used in the game in the box on the next page.
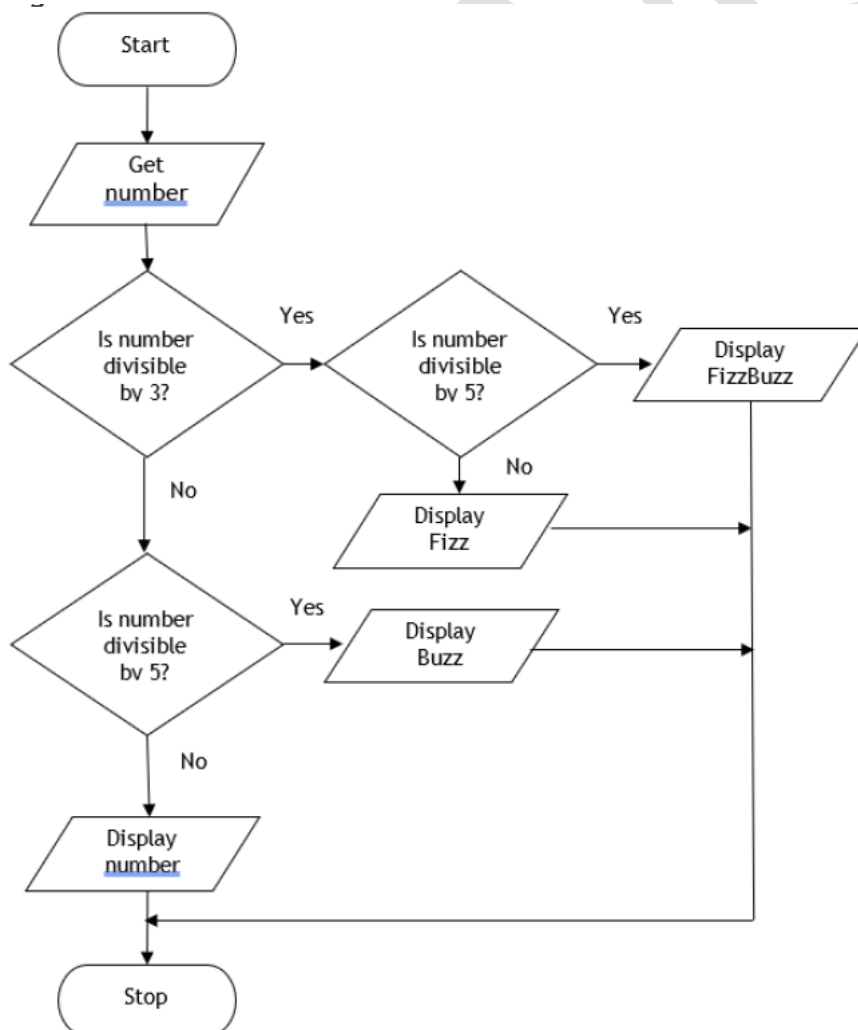
2019 May Jun
3 (e)

(e) Information sent across networks is represented in bit patterns.

   (i) The bit pattern 1101 0001 uses sign and magnitude representation.

     Convert this bit pattern to a denary number.

(2)

3(e)(i)	-81
	One mark for negative sign One mark for 81.

   (ii) Convert the denary number 75 to 8-bit binary.

(2)

3(e)(ii)	One mark for each nibble in correct order
	0100 1011

(iii) The addition of these two 8-bit binary patterns generates an error condition.

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

Explain this error condition.

(2)

Any **two** from:

1. This is an overflow (1) error
2. It is caused by adding two 1s in the most significant bit (1)
3. The resulting number is too large to fit in the number of bits available / requires 9 bits / the number is larger than 2^8 / there

4   One function of an operating system is file management.

   (a) Both kibibyte and kilobyte can be used as measures of file size.

   Compare kibibyte and kilobyte.

(2)

Kilobyte is equivalent to 1000 bytes, whereas a kibibyte is equivalent to 1024 bytes (1)

Kilobyte is equivalent to 10^3 bytes, whereas a kibibyte is equivalent to 2^{10} bytes (1)

Kilobyte is a base 10 measurement, whereas a kibibyte is a base 2 measurement (1)

   (b) An image is 2322 pixels high and 4128 pixels wide.

   The image is stored with a 16-bit colour depth.

   The metadata for the image is 975 bytes.

   Construct an expression to show how the file size, in megabytes, is calculated.

   You do **not** need to do the calculation.

(4)

- Calculates correct bits/bytes(1)
- Adds metadata (1)
- Calculates correct megabytes (1)
- Completely correct response (1)

Examples:

$$\frac{\frac{2322 \times 4128 \times 16}{8} + 975}{1000 \times 1000}$$

$$\frac{(2322 \times 4128 \times 2) + 975}{1000 \times 1000}$$

$$\frac{((2322 \times 4128 \times 16 \text{ bits per colour}) \div 8 \text{ bits per byte}) + 975 \text{ bytes}}{1000 \text{ bytes per KB} \times 1000 \text{ kilobytes per MB}}$$

$$\frac{((2322 \times 4128 \times 2 \text{ bytes per colour}) + 975 \text{ bytes}}{1000 \text{ bytes per KB} \times 1000 \text{ kilobytes per MB}}$$

(d) A text file is stored on a hard disc.

The file holds information from one side of a sheet of paper.

The sheet of paper is represented as a grid, 80 columns wide and 66 rows long.

Each cell in the grid contains a single 2-byte Unicode character.

The file also contains 40 characters of metadata.

The hard disc allocates space in blocks of 1024 bytes.

Construct an expression to show the number of blocks required to store the file.

You do **not** need to do the calculation.

(4)

Award 4 marks for 11
Award 3 marks for a value between 10 and 11

One mark for each:

- 80 x 66 x 2 or 80 x 66 x 16 ÷ 8   (1)
- + 40 or + 80   (1)
- ÷ 1024   (1)
- Ceiling or RoundUp   (1)

2020 May Jun

2 Computers use binary digits to represent data.

(a) Sign and magnitude and two's complement are two ways of representing signed integer numbers.

(i) Convert the denary number -94 to a binary pattern using sign and magnitude representation.

(2)

1101 1110

(ii) 1000 1001 is a signed integer that uses two's complement representation.

Convert this signed integer to denary.

(2)

- (1)
119 (1)

(b) Denary numbers use base 10

Identify the base for hexadecimal numbers.

(1)

   ☒ **A** 32

   ☒ **B** 16

   ☒ **C** 8

   ☒ **D** 2

B 16

(c) Computers encode characters using ASCII and Unicode.

(i) State the number of characters that can be represented using standard ASCII.

(1)

128

(ii) The ASCII system uses 7 bits to represent a character. The ASCII code for the character 'A' using denary is 65; other alphabetical characters follow on from this in sequence.

Identify the ASCII code for 'H'.

(1)

☒ **A** 1000111

☒ **B** 1001010

☒ **C** 1001000

☒ **D** 1000101

(iii) Explain why Unicode was developed.

(2)

C

Standard ASCII only provides 128 different patterns (1) can't represent all major languages/symbols/characters (1)

Unicode uses a minimum of 16 bits (1) so can represent at least 2^{16} characters (1)

2021 May Jun

1   Binary digits (bits) are grouped together to represent different types of data.

   (a) Hexadecimal notation is sometimes used to represent patterns of binary digits.

      Identify **one** reason why programmers use hexadecimal notation.

      (1)

         ☒   **A**   Easier for a computer to understand.

         ☒   **B**   Easier for humans to read.

         ☒   **C**   Takes up less computer memory.

         ☒   **D**   Quicker than binary to execute.

B

   (b) Convert the bit pattern 0101 1010 to hexadecimal.

      (2)

5A

   (c) Identify the number of binary patterns that can be represented by 8 bits.

      (1)

         ☒   **A**   128

         ☒   **B**   256

         ☒   **C**   512

         ☒   **D**   1024

B

   (d) Complete the table by adding these two 8-bit binary integers.

      (2)

0	0	1	1	0	0	1	0
0	0	1	1	0	0	1	1

0110 0101

   (e) Give the result of performing a logical left shift of 1 place on the binary integer 0100.

      (1)

01000 (OR) 1000

   (f) Give the number of bits per character used by standard ASCII.

      (1)

7

⊐⌐∨⌐∨⊐<⌐>∨⌐⊡⊐⌐⌐

(g) **Figure 1** shows a black and white bitmap image.

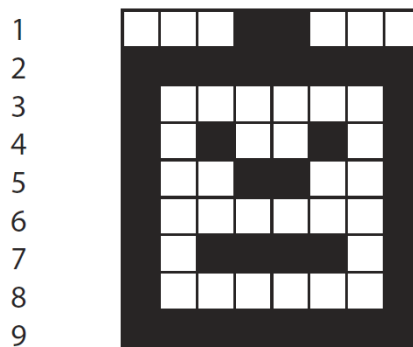The pixels in row 5 are represented by the binary pattern 1001 1001

Figure 1

   (i) State what is meant by the term **pixel**.

(1)

A picture element

  (ii) Construct an expression to show the number of pixels in the image.

(1)

8 x 9 (OR) 9 x 8 (OR) 72

  (iii) The image is changed so that any pixel can be one of 16 different colours.

State the minimum number of bits that would be needed to represent **one** pixel.

(1)

4

  (ii) Identify which **one** of these file formats uses lossy compression.

(1)

    ☒ **A** BMP

    ☒ **B** DOC

    ☒ **C** JPEG

    ☒ **D** PNG

C

(iii) Here is a string of data.

CCCWWWCCWWWWWWCCC

Give the result of compressing the string using a run-length encoding algorithm.

(1)

3c3w2c6w3c (1)

c3w3c2w6c3 (1)

(g) A video file is to be transmitted over the internet.

- The network transmission speed is 54 Mbps.
- The file size is 6 gigabytes (GB).

Construct an expression to show how the transmission time, in seconds, is calculated.

You do **not** have to do the calculation.

(4)

$$\frac{6 \times 1000 \times 1000 \times 1000 \times 8}{54 \times 1000 \times 1000}$$

2021 Nov

1 Computers use binary to represent and store data.

    (a) Binary and hexadecimal notation can be used to represent numbers.

        (i) Convert the denary number 77 to 8-bit binary.

        (2)

0100 1101

      (ii) Convert the denary number –126 to 8-bit binary using two's complement.

        (2)

1000 0010

    (iii) Convert the binary pattern 11000110 to hexadecimal.

        (2)

C (1)
6 (1)

    (b) Images are stored as binary data.

        (i) A bitmap image is 400 pixels wide by 200 pixels high.

           It has a colour depth of 12 bits.

           Construct an expression to show how the image size, in bytes, is calculated.

           Do **not** calculate the answer.

        (2)

400 pixels x 200 pixels x 12 bits / 8 bits per byte
Allow expression in word form

$$\frac{400 \times 200 \times 12}{8}$$

400 x 200 **OR** 80000(1)

12/8 **OR** 1.5 (1)

⊐┌∨⌐∨⊐< ⌐>∨ ⌐□⊐⌐⌐

(ii) **Figure 1** is an image of an icon, set out in an 8×8 pixel grid.
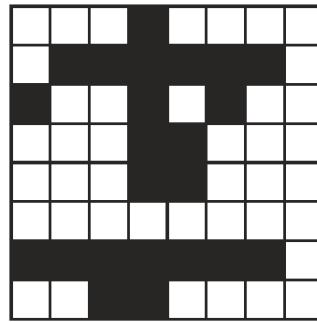
Figure 1

A run-length encoding compression algorithm uses:

* W for white pixels

* B for black pixels.

Give the result of compressing the first 4 lines of the icon using the algorithm.

(3)

Spaces are to help legibility.
Award **three** marks for:

Line 1   3W 1B 4W
Line 2   1W 6B 1W
Line 3   1B 2W 1B 1W 1B 2W
Line 4   3W 2B 3W

Award **two** marks for any three lines correctly encoded

Award **one** mark for any two lines correctly encoded

2022 May Jun

2 Computers use binary to represent and store data.

    (a) The denary number 78 is the ASCII code for the character **N**.

        (i) Convert the denary number 78 to 8-bit binary.

(2)

0100 1110

        (ii) Identify the number of characters that can be represented using standard ASCII.

(1)

           ☒ **A** 64

           ☒ **B** 128

           ☒ **C** 256

           ☒ **D** 512

B

        (iii) Explain **one** reason for using Unicode rather than ASCII to encode languages other than English.

(2)

Unicode can represent all/more characters/any language (1) whereas ASCII can only represent English/Latin/128 characters/doesn't have enough characters (1)

    (b) Convert the denary number –43 to 8-bit binary using sign and magnitude representation.

(2)

1010 1011

    (c) Complete the table by adding these two 8-bit binary integers.

(2)

0	0	1	1	0	1	0	0
0	0	0	1	0	1	1	0

0100 1010

(i) An image has five colours.

Complete the table by adding a unique binary pattern for each colour.

Each pattern must use the **same minimum colour depth**.

(2)

Colour	Binary pattern
Green	
Black	
White	
Red	
Blue	

Award up to **two** marks for:

- 3 bits used for all patterns (1)/
- No pattern repeated (1)

Example:

Colour	Binary pattern
Green	000
Black	001
White	010
Red	011
Blue	100

(ii) Another image is 3579 pixels high and 6128 pixels wide.

The image is stored with a 32-bit colour depth.

The metadata for the image is 732 bytes.

Construct an expression to show how the file size, in **megabytes**, is calculated.

You do **not** need to do the calculation.

(4)

Award **one** mark for each of:

- 3579 x 6128 x 32 (1)
- ÷ 8 (1)
- + 732 (1)
- ÷ (1000 x 1000) (1)

Examples

$$\frac{\dfrac{3579 \times 6128 \times 32}{8} + 732}{1000 \times 1000}$$

2023 May Jun

2 Computers manipulate binary patterns. People interpret those patterns.

(a) Complete the table by adding the hexadecimal notation for each of the denary values.

(2)

Denary	Hexadecimal
8	
12	

Denary	Hexadecimal
8	8 (1)
12	C (1)

(b) Identify the expression to give the number of unique binary patterns that can be stored in six bits.

(1)

☒ **A** 6^2

☒ **B** 6×2

☒ **C** $6^2 - 1$

☒ **D** 2^6

D

(c) Binary patterns are manipulated by shifts.

(i) Give the result of applying a logical shift right by two to the binary pattern 0101 1100

(1)

0001 0111

(ii) Give the result of applying an arithmetic shift right by three to the binary pattern 1100 0101

(1)

1111 1000

(d) Binary patterns can be interpreted as signed or unsigned integers.

    (i) Convert the denary unsigned integer 60 to 8-bit binary.

(2)

0011 1100

    (ii) Here is a binary bit pattern for a signed integer in sign and magnitude format.

1001 0110

    Convert the binary bit pattern to denary.

    Be sure to include a sign symbol in your answer.

(2)

-22

   (iii) Negating a signed integer means changing its sign without changing its value.

    The negation of +16 is −16. The negation of −24 is +24.

    Here is the binary bit pattern for a signed integer in two's complement format.

1110 0101

    Convert the binary pattern to its negation in two's complement.

(2)

0001 1011

(e) Construct an expression to convert 13 kilobytes to kibibytes.

You **do not** need to do the calculation.

(2)

Award **one** mark for each of:

1. 13 x 1000 // 13000 in the numerator (1)
2. 1024 in the denominator (1)

Examples:

$$\frac{13 \times 1000}{1024}$$

$$\frac{13000}{1024}$$

$$\frac{13 \times 500}{512}$$

$$\frac{13 \times 250}{256}$$

$$\frac{13 \times 125}{128}$$

$$\frac{13 \times 10^3}{2^{10}}$$

(d) Here is an image of secondary storage.

Two files (W and Z) are stored on it.

Each file is made up of several blocks (e.g. Z1, Z2, Z3).

	W4	Z1				W2		W3	Z3		W1		Z2

Complete the image to show the state after running a defragmentation utility.

(2)

W1	W2	W3	W4	Z1	Z2	Z3									

								Z1	Z2	Z3	W1	W2	W3	W4

Z1	Z2	Z3	W1	W2	W3	W4								

								W1	W2	W3	W4	Z1	Z2	Z3

(e) Here is part of a file that contains electric meter readings.

04631   04984   05103   05163   05271   05383   05487   05722

Explain the effect on the file of applying a run-length encoding algorithm to this data file.

(2)

The file size will not be decreased/will increase (1) because there are very few/no repeating patterns / because a run length of one would be added to each digit (1)

2020 May Jun

5 Programmers work as part of a team to develop software.

(a) They use truth tables to determine if an expression evaluates to true or false.

Complete this truth table for the Boolean expression

P AND (Q OR R)

You may not need to use all rows.

(3)

P	Q	R	Q OR R	P AND (Q OR R)

P	Q	R	Q OR R	P AND (Q OR R)
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	1	0
1	0	0	0	0
1	0	1	1	1
1	1	0	1	1
1	1	1	1	1

(b) The team is working on software for a ride-on lawn mower.

The lawn mower must not start unless it is safe for the driver.

- The brake (B) must be engaged.

- The driver's seat sensor (S) must be engaged.

- The blade safety lock (L) must be removed.

Construct a Boolean expression, using AND, OR, and NOT with the letters B, S and L to show the safe starting condition.

(2)

B AND S AND NOT L
Award **one** mark for each of:
NOT L or B AND S (1)
Fully correct (1)

2024 May Jun

3   Computers use binary to represent and store data.

(a) Two images are stored.

- Image 1 has a colour depth of 8 bits.

- Image 2 has a colour depth of 16 bits.

(i) State the number of colours that can be represented in Image 1.

(1)

....

(ii) Image 2 is 1920 pixels wide by 1080 pixels high.

Construct an expression to show the file size of the image in mebibytes.

You do **not** need to do the calculation.

(3)

(iii) One pixel in Image 1 is represented by the unsigned binary integer

1100 1010

Identify the denary equivalent.

(1)

- ☒ **A** 54

- ☒ **B** 191

- ☒ **C** 202

- ☒ **D** −74

3(a)(i)	Award **one** mark for any of the following: • 256 (1) • 2^8 (1) • 2x2x2x2x2x2x2x2 (1)	
3(a)(ii)	Award **one** mark for each of the following up to a maximum of **three** marks: • 1920 x 1080 in the numerator (1) • x 16/8 (16 in the numerator and 8 in the denominator) or (2 in the numerator) (1) • / (1024 x 1024) in the denominator (1) 1920 pixels x 1080 pixels x 8 bits / 8 bits per byte / 1024 x 1024 bytes per MiB $$\frac{1920 \times 1080 \times \frac{16}{8}}{1024 \times 1024}$$	Max 2 marks if not a fully correct expression. Allow equivalence e.g. $1024 = 2^{10}$ $1024^2 = 2^{20}$ 2^{-20} is equivalent to $1/2^{20}$
3(a)(iii)	**The only correct answer is C** *A is not correct because the MSB is 1 so the denary value must be more than 128* *B is not correct because the LSB is 0 so the denary value must be even* *D is not correct because the denary number is signed*	

(b) A register holds the value 1100 0001 as an 8-bit binary value in two's complement.

   (i) Convert the two's complement binary integer 1100 0001 to denary.

                                               (2)

   (ii) Give the result of applying an arithmetic shift left by one to the binary pattern 1100 0001

                                               (1)

   (iii) The value 1111 1111 is added to the value 1100 0001 stored in the register.

      An error occurs.

      Explain this error.

                                               (2)

3(b)(i)	Award **one** mark for each of: • Negative sign (1) • 63 (1) Example: - 63
3(b)(ii)	Award **one** mark for: 1000 0010 (1)
3(b)(iii)	Award up to **two** marks for a linked explanation such as: • An overflow error has occurred (1) because the result requires more bits to store than are available (1)

(c) The ASCII code for the character **1** is 011 0001

   (i) Give the ASCII code for the character **5**

(1)

....

   (ii) Give **two** benefits of using ASCII to encode characters.

(2)

1 ....

....

2 ....

....

3(c)(i)	Award **one** mark for any of the following: • 011 0101 (1) • 53 (1) • Hex value of 35 (1) Do not accept 11 0101 (six bits, ASCII is at least seven)

3(c)(ii)	Award **one** mark for any of the following up to a maximum of **two** marks: • It is a standard/universal encoding method (1) • It can represent all the characters in the English alphabet (1) • It only uses 7/8 bits **per character** (1) • It can represent 128/256 characters (in standard/extended format) (1) • It takes up less storage space than Unicode (1) • It can be used with bit masking (1) • It is compatible with UTF-8 (1)

(d) Tarik is writing an instruction manual for the computer game.

He decides to use software to compress the instruction manual before emailing it to a proofreader for checking.

State and justify the most appropriate compression method.

(3)

Method

....

Justification

....

....

| 5(d) | Award **one** mark for:
• Lossless (1)

Award **one** mark for any of the following up to a maximum of **two** marks:
• No data will be lost / the process is reversable (1)
• The manual will still be readable (1)
• Removal of data would change the meaning of text (1)
• Proofreader will be able to edit the text (1) | Allow converse if candidate gives reasons why lossy would not be used for justification

Allow RLE as a valid lossless compression method. |

2021 Nov

(c) Akiko is also working on a burglar alarm system.

The alarm must only be triggered if:

- the alarm (A) is set

- it is dark outside (O)

- a window (W) or a door (D) has been opened.

Construct a Boolean expression, using AND, OR and NOT with the letters A, O, W, and D to show the conditions that will trigger the alarm.

(4)

A AND O AND (W OR D)

Award **one** mark each to a maximum of **four** marks for:

- A AND O (1)
- W OR D (1)
- Brackets around W OR D (1)
- AND between A AND O (W OR D) (1)

2023 May Jun

(d) Programs use logic statements to control physical hardware.

(i) A window shuts when the temperature is too cool or it is a rainy night.

The values are defined as:

- A shows it is night time
- B shows it is too cool
- C shows it is raining.

Complete the truth table to show the results of each operation.

Two rows have been done for you.

(3)

A	B	C	A AND C	(A AND C) OR B
0	0	0	0	0
0	0	1		
0	1	0		
0	1	1		
1	0	0		
1	0	1		
1	1	0		
1	1	1	1	1

2023 May Jun

A	B	C	A AND C	(A AND C) OR B
0	0	0	0	0
0	0	1	0	0
0	1	0	0	1
0	1	1	0	1
1	0	0	0	0
1	0	1	1	1
1	1	0	0	1
1	1	1	1	1

(ii) A warehouse has an automated alarm system.

When the alarm system is activated it will sound if:

- a movement sensor (M) is activated

- a pressure pad (P) is activated

- a key code (C) to deactivate the alarm system has not been entered.

Construct a logic statement, using AND, OR and NOT with the letters M, P and C, to show the conditions that will sound the alarm.

(3)

Award **one** mark for each of:

- M OR P (1)
- NOT C (1)
- <expr1> AND <expr2> (1)

Example statements:

(M OR P) AND (NOT C)
(M OR P) AND NOT C
(M AND NOT C) OR (P AND NOT C)
M AND NOT C OR P AND NOT C

For information only:

M	P	C	M OR P	NOT C	(M OR P) AND (NOT C)
0	0	0	0	1	0
0	0	1	0	0	0
0	1	0	1	1	1
0	1	1	1	0	0
1	0	0	1	1	1
1	0	1	1	0	0
1	1	0	1	1	1
1	1	1	1	0	0

2024 May Jun

(e) Members of the team developing the game are either coders or designers.

Coders (C) write the application code.

Designers (D) design the graphics for the game and write instruction manuals.

Team members have one role only.

Construct a logic statement, using AND, OR and NOT with the letters C and D, to show the conditions for being a member of the team.

(3)

| 5(e) | Award **one** mark for each of:

Method 1
• (C AND NOT D) (1)
• (NOT C AND D) (1)
• <expr1> OR <expr2> (1)
Method 2
• (C OR D) (1)
• NOT (C AND D) (1)
• <expr1> AND <expr2> (1)

Example statements:
    (C AND NOT D) OR (NOT C AND D)
    (C OR D) AND NOT (C AND D) | Order of precedence is NOT, AND, OR

Note other equivalent solutions

MAX 2 if all terms correct apart from missing brackets that are required.

Do not credit circuit diagrams. |