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.

```
# This is the pseudocode for an algorithm
SET inNum TO 0
SET result TO 1
SET i TO 0

SEND "Enter a number: " TO DISPLAY
RECEIVE inNum FROM (INTEGER) KEYBOARD

IF (inNum < 0) THEN
   SEND "Invalid input" TO DISPLAY
ELSE
    IF (inNum = 0) THEN
        SEND "Answer is 1" TO DISPLAY
    ELSE
        FOR i FROM 1 TO inNum DO
            SET result TO result * i
        END FOR
        SEND "The answer is " & result TO DISPLAY
    ENDIF
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)

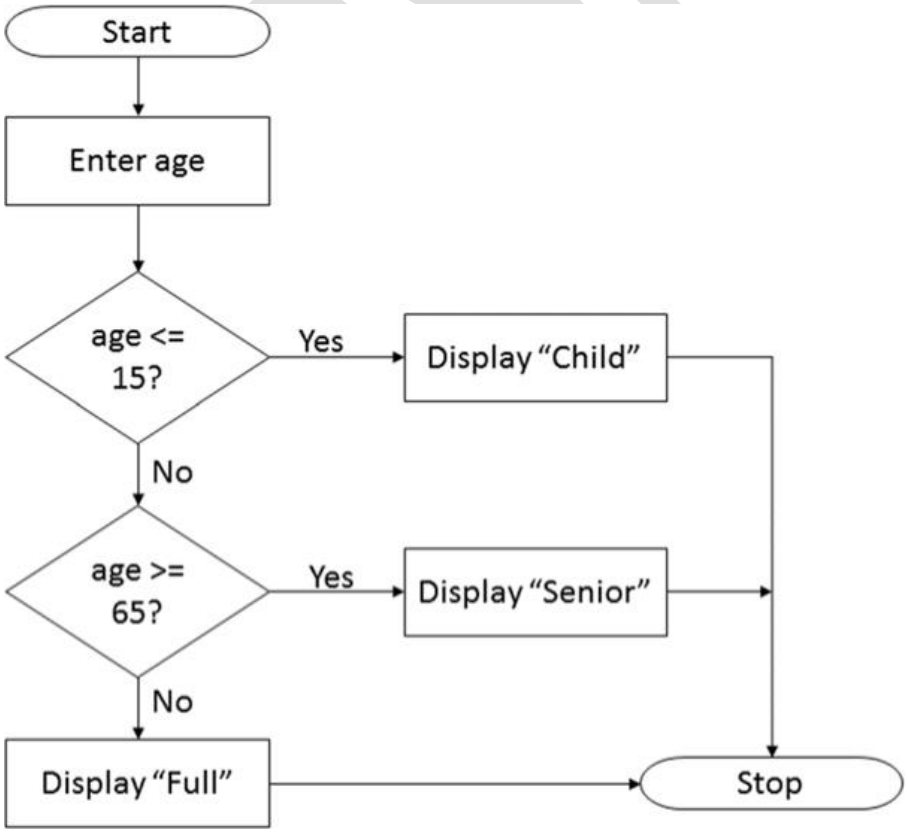
Input	Output message
0	
-12	
5	

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)



(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.

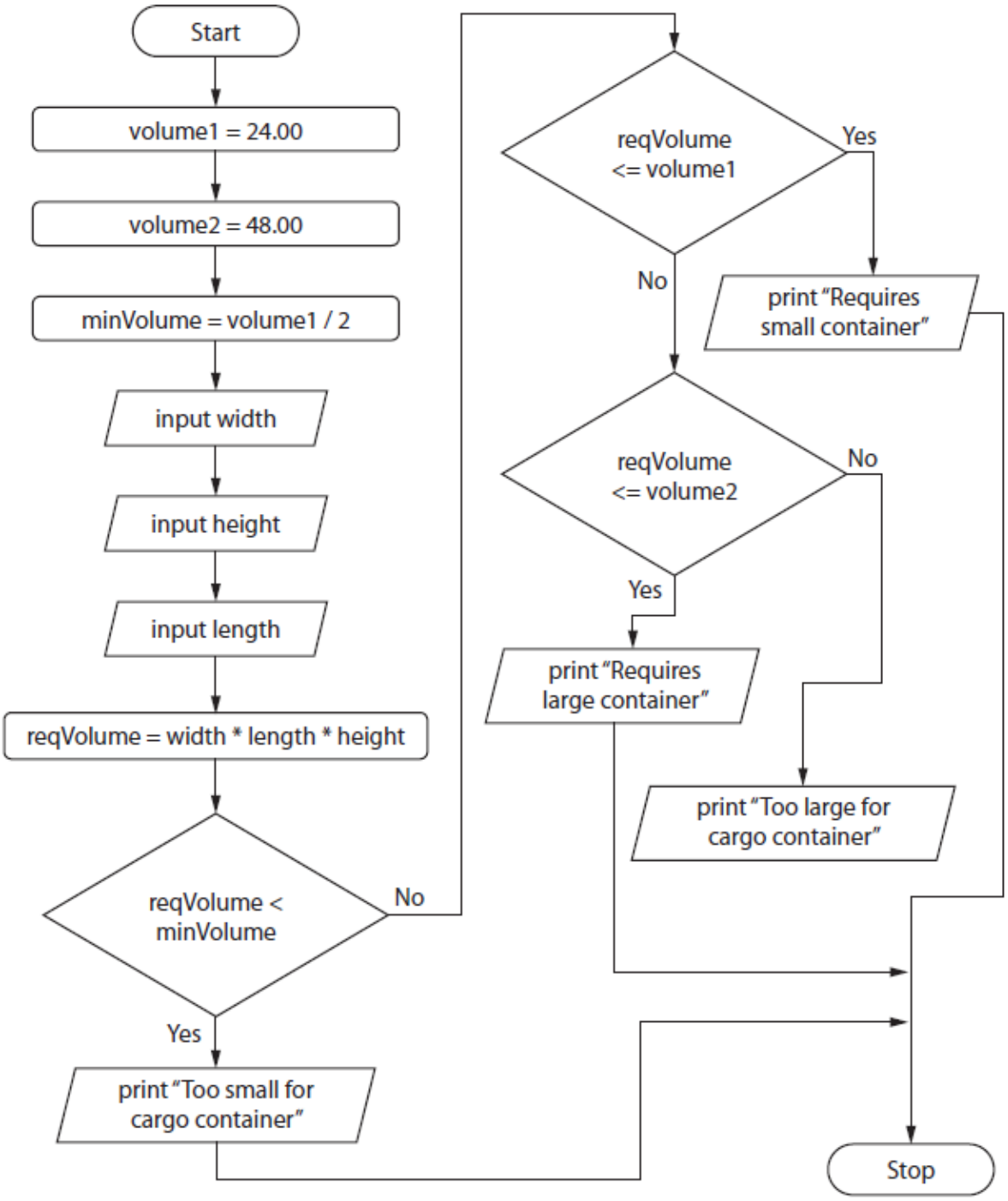


Figure 2

(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)

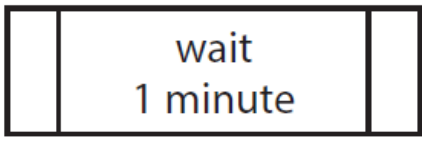
.....

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

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

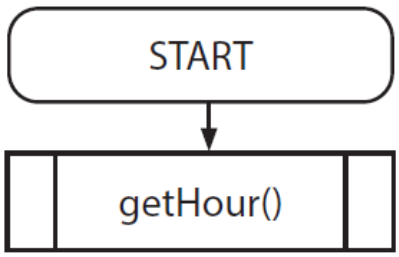
2020

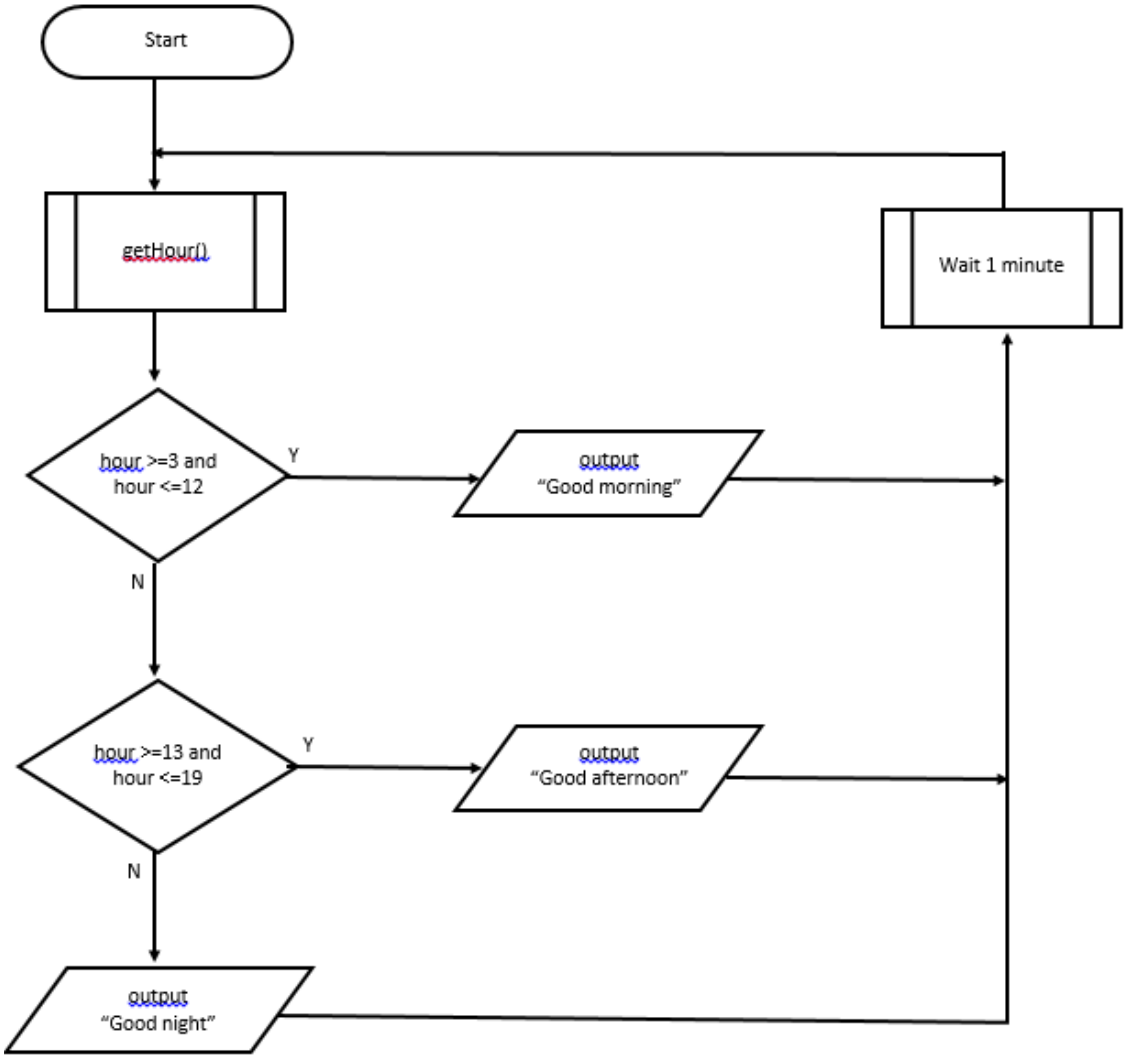
6 A software engineer is working with some algorithms.
(a) An algorithm needs to be developed.
The algorithm must:
• check for a change of hour every minute
• use the library subprogram getHour() to get the hour part of the current time in the 24-hour clock (0 to 23)
• output “Good morning” when the 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.



(6)

Draw your flowchart here.





(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.

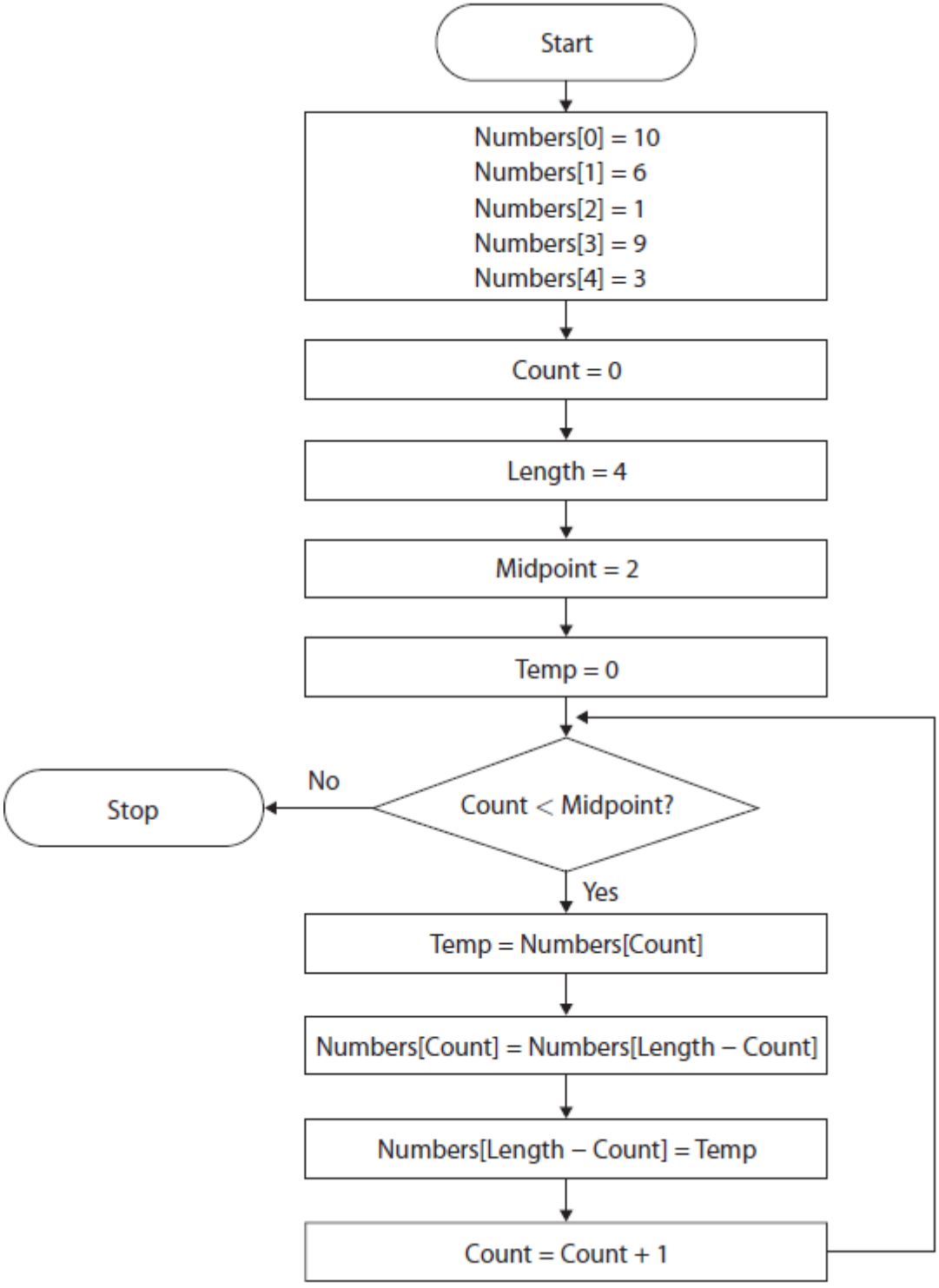


Figure 3

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

(5)

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

				Numbers array				
Count	Length	Midpoint	Temp	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 (1)

.....

(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)

2021 ON

Algorithms can be used to perform calculations and to process data.
(a) State what is meant by the term **algorithm**.
(1)

- a step-by-step description of a process that completes a task (1)
- a set of instructions that describes how to get something done (1)

(b) **Figure 3** shows an algorithm.

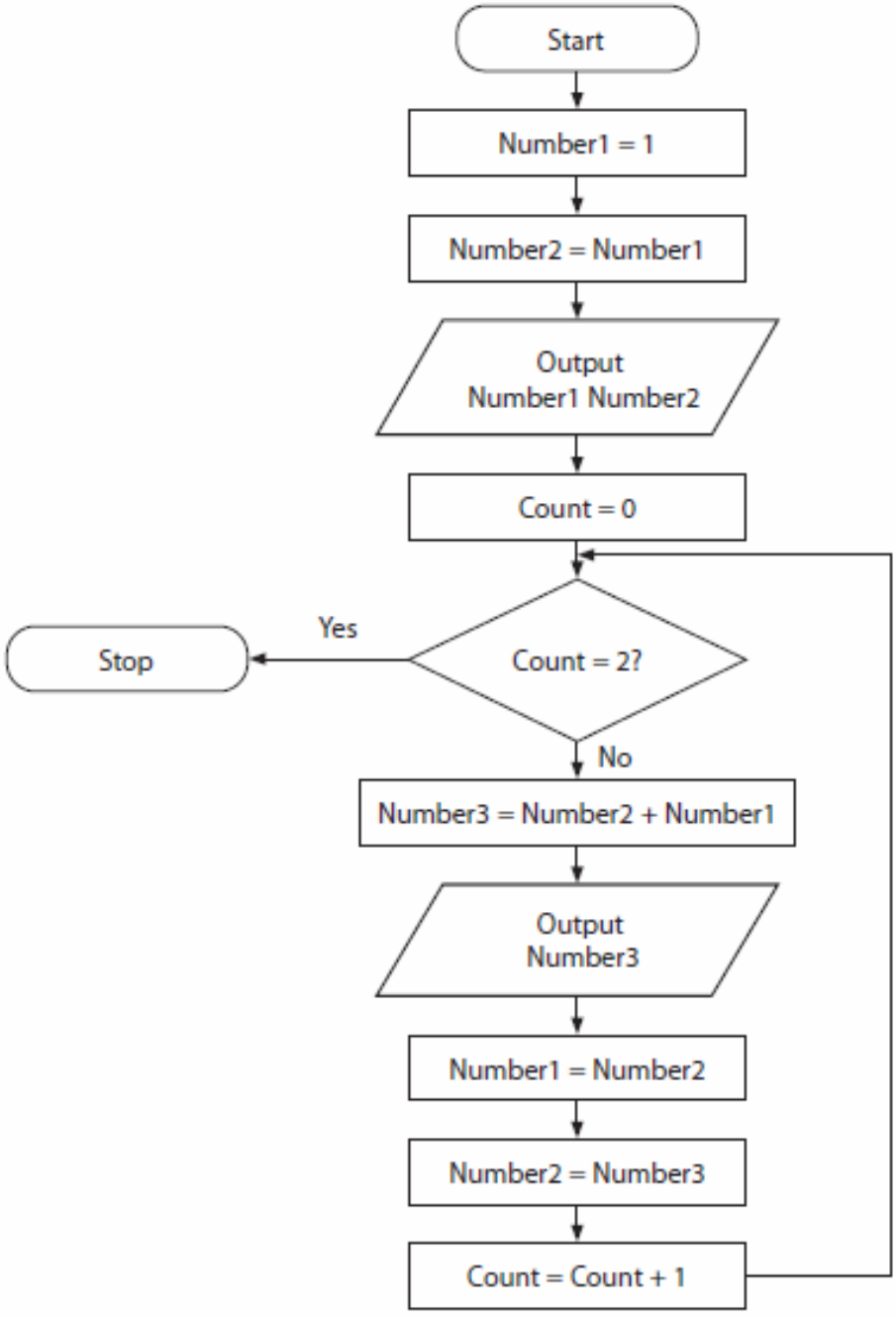


Figure 3

(i) Complete this trace table for the algorithm.

You may not need to use all the rows.

(5)

Number1	Number2	Number3	Count	Output	Count = 2?
1	1	–	0	1 1	False

Number 1	Number 2	Number 3	Count	Output	Count = 2?
1	1	-	0	1 1	False
1	2	2	1	2	False
2	3	3	2	3	True

(ii) The benefits of using a trace table include that they allow variable states, outputs and decisions to be recorded.

Give **one other** benefit of using a trace table to test an algorithm.

(1)

- helps visualise how the algorithm works (1)
- helps detect (logic) errors (1)

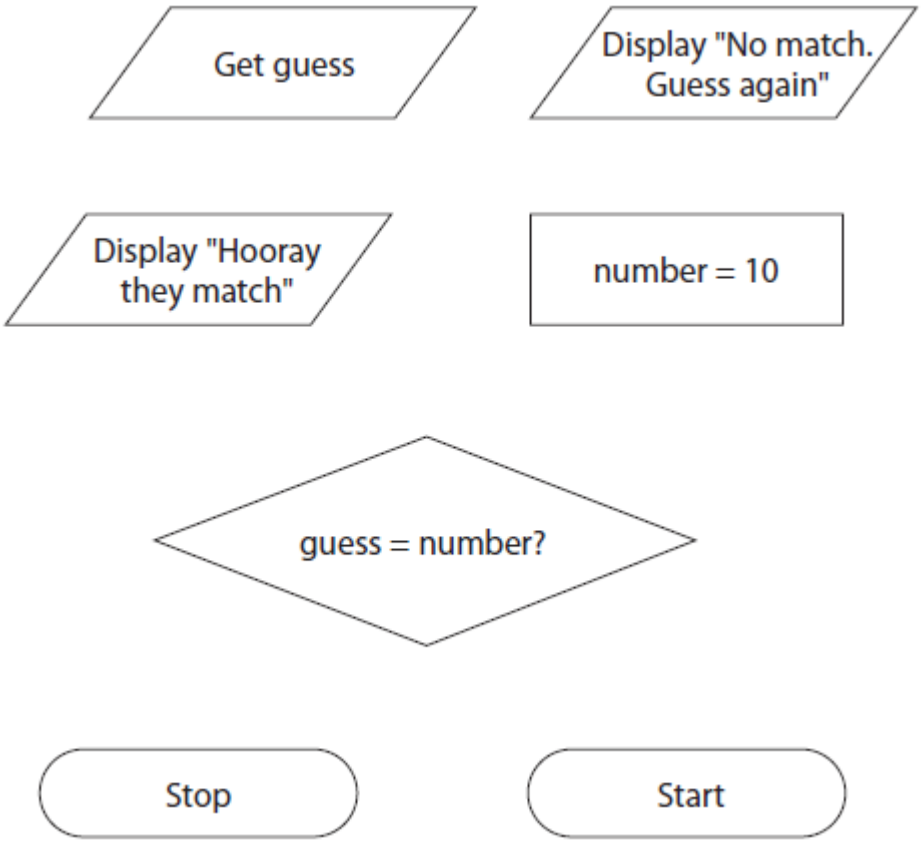
(c) An algorithm is needed to count and display the number of vowels in a word.
The vowels are a, e, i, o, and u.
The completed algorithm must:
• count the number of vowels
• create a message as a single string (e.g. there are *number* vowels)
• print the message to the display.
Here is a partially completed algorithm written in pseudocode.
Complete the algorithm in the spaces provided.
(5)

```
SET word TO 'elephant'

FOR EACH letter FROM word DO

END FOREACH
```

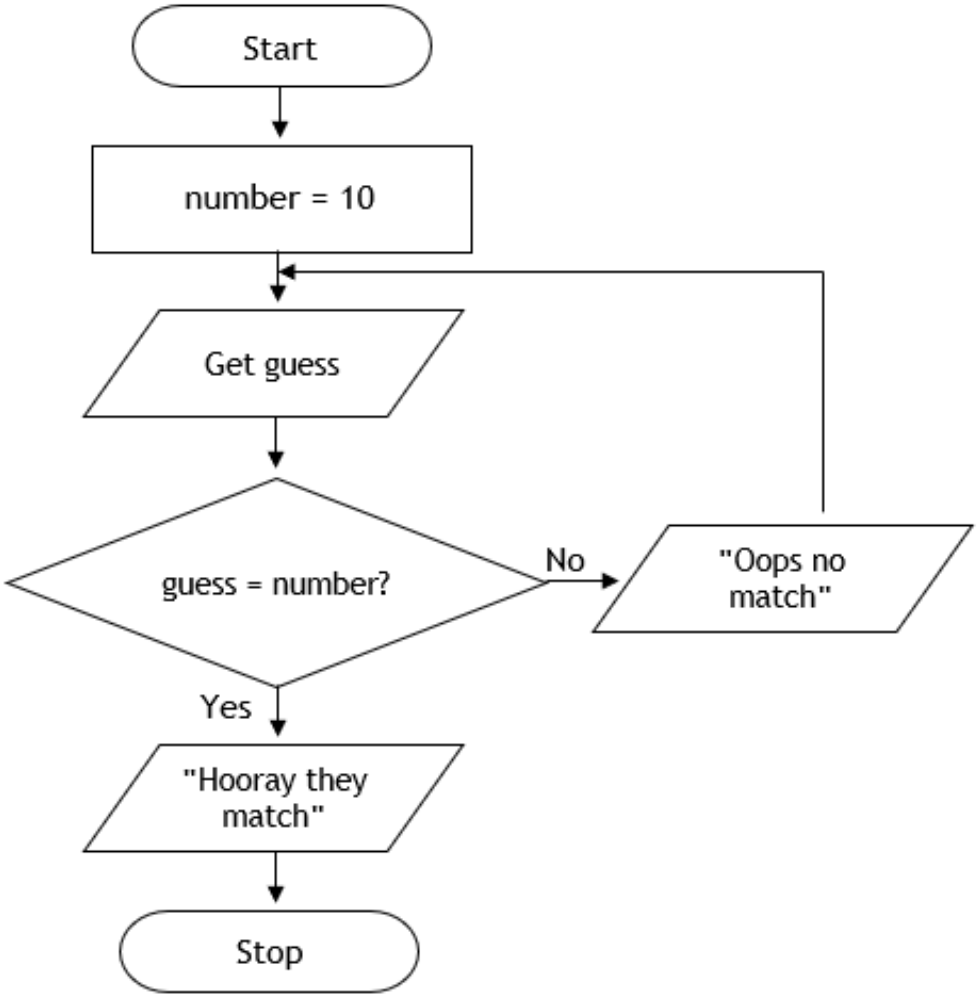
```
SET word TO "elephant"
SET count TO 0

FOR EACH letter FROM word DO
    IF letter = 'a' OR letter = 'e' OR letter = 'i' OR  letter = 'o' OR letter = 'u'  THEN

        SET count TO count + 1
   END IF

END FOREACH

SEND 'The number of vowels is ' & count TO DISPLAY
```

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)


Start
number = 10
Get guess
guess = number?
No
"Oops no match"
Yes
"Hooray they match"
Stop

(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					

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

(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

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

5 Hardware devices execute programs to carry out a variety of tasks.

(a) A program controls a bee character in an animation.

The bee can turn to face North, East, South or West.

The bee can move any number of steps in the direction it is facing.

Complete the table to show one input and one output required to move the bee.

(2)

Requirement	Example
Input	
Process	Calculate the path the bee will move along to its new position
Output	

Req.	Example
Input	1. Enter the direction the bee will face (1) 2. Enter the number of steps that the bee will move (1)
Process	Calculate the path the bee will move along to its new position
Output	3. The bee appears in its new position (1) 4. The bee flies to its new position (1) 5. The bee faces its new direction (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`	

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)	

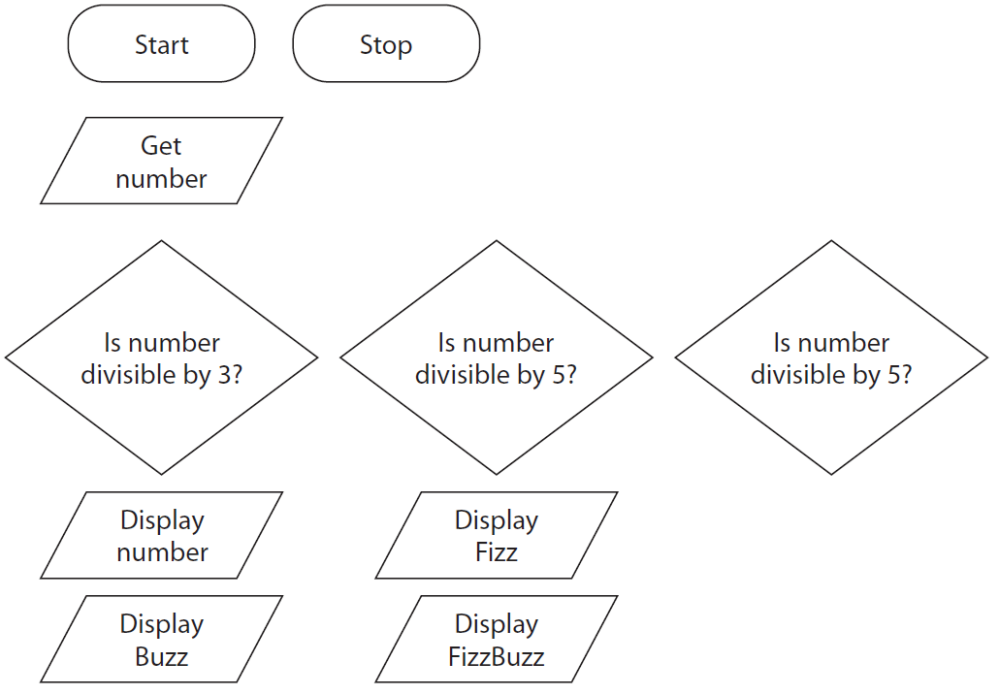
(d) In the game FizzBuzz a user enters a number.

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.