QUESTION 4.

6 A string-handling function has been developed. The pseudocode for this function

For the built-in functions list, refer to the **Appendix** on page 18.

```
FUNCTION SSM(String1, String2 : STRING) RETURNS INTEGER
    DECLARE n, f, x, y : INTEGER

    n ← 0
    f ← 0

    REPEAT
        n ← n + 1
        x ← n
        y ← 1
        WHILE MID(String1, x, 1) = MID(String2, y, 1)

            IF y = LENGTH(String2)
                THEN
                    f ← n
                ELSE
                    x ← x + 1
                    y ← y + 1
            ENDIF

        ENDWHILE

    UNTIL (n = LENGTH(String1)) OR (f <> 0)

    RETURN f

ENDFUNCTION
```

(a) Complete the trace table below by performing a dry run of the function when it is called as follows:

```
SSM("RETRACE", "RAC")
```

n	f	x	y	MID(String1, x, 1)	MID(String2, y, 1)
0	0				

[6]

(b) (i) Describe the purpose of function `SSM`.

.....

.....

.....

.....[2]

(ii) One of the possible return values from function `SSM` has a special meaning.

State the value and its meaning.

Value .....

Meaning .....
[2]

(iii) There is a problem with the logic of the pseudocode. This could generate a run-time error.

Describe the problem.

.....

.....

.....

.....[2]

Appendix

Built-in functions

In each function below, if the function call is not properly formed, the function returns an error.

`MID(ThisString : STRING, x : INTEGER, y : INTEGER) RETURNS STRING`

returns the string of length `y` starting at position `x` from `ThisString`

Example: **`MID("ABCDEFGH", 2, 3)`** will return string **`"BCD"`**

`LEFT(ThisString : STRING, x : INTEGER) RETURNS STRING`

returns the leftmost `x` characters from `ThisString`

Example: **`LEFT("ABCDEFGH", 3)`** will return string **`"ABC"`**

`RIGHT(ThisString: STRING, x : INTEGER) RETURNS STRING`

returns the rightmost `x` characters from `ThisString`

Example: **`RIGHT("ABCDEFGH", 3)`** will return string **`"FGH"`**

`ASC(ThisChar : CHAR) RETURNS INTEGER`

returns the ASCII value of character `ThisChar`

Example: **`ASC('W')`** will return **`87`**

`LENGTH(ThisString : STRING) RETURNS INTEGER`

returns the integer value representing the length of string `ThisString`

Example: **`LENGTH("Happy Days")`** will return **`10`**

String operator

`&` operator

concatenates (joins) two strings

Example: **`"Summer" & " " & "Pudding"`** produces **`"Summer Pudding"`**

BLANK PAGE

QUESTION 5.

6 A string-handling function has been developed. The pseudocode for this function

For the built-in functions list, refer to the **Appendix** on page 18.

```
FUNCTION SSM(String1, String2 : STRING) RETURNS INTEGER
    DECLARE n, f, x, y : INTEGER

    n ← 0
    f ← 0

    REPEAT
        n ← n + 1
        x ← n
        y ← 1
        WHILE MID(String1, x, 1) = MID(String2, y, 1)

            IF y = LENGTH(String2)
                THEN
                    f ← n
                ELSE
                    x ← x + 1
                    y ← y + 1
            ENDIF

        ENDWHILE

    UNTIL (n = LENGTH(String1)) OR (f <> 0)

    RETURN f

ENDFUNCTION
```

(a) Complete the trace table below by performing a dry run of the function when it is called as follows:

```
SSM("RETRACE", "RAC")
```

n	f	x	y	MID(String1, x, 1)	MID(String2, y, 1)
0	0				

[6]

(b) (i) Describe the purpose of function `SSM`.

.....

.....

.....

.....[2]

(ii) One of the possible return values from function `SSM` has a special meaning.

State the value and its meaning.

Value .....

Meaning .....
[2]

(iii) There is a problem with the logic of the pseudocode. This could generate a run-time error.

Describe the problem.

.....

.....

.....

.....[2]

Appendix

Built-in functions

In each function below, if the function call is not properly formed, the function returns an error.

`MID(ThisString : STRING, x : INTEGER, y : INTEGER) RETURNS STRING`

returns the string of length `y` starting at position `x` from `ThisString`

Example: **`MID("ABCDEFGH", 2, 3)`** will return string **`"BCD"`**

`LEFT(ThisString : STRING, x : INTEGER) RETURNS STRING`

returns the leftmost `x` characters from `ThisString`

Example: **`LEFT("ABCDEFGH", 3)`** will return string **`"ABC"`**

`RIGHT(ThisString: STRING, x : INTEGER) RETURNS STRING`

returns the rightmost `x` characters from `ThisString`

Example: **`RIGHT("ABCDEFGH", 3)`** will return string **`"FGH"`**

`ASC(ThisChar : CHAR) RETURNS INTEGER`

returns the ASCII value of character `ThisChar`

Example: **`ASC('W')`** will return **`87`**

`LENGTH(ThisString : STRING) RETURNS INTEGER`

returns the integer value representing the length of string `ThisString`

Example: **`LENGTH("Happy Days")`** will return **`10`**

String operator

`&` operator

concatenates (joins) two strings

Example: **`"Summer" & " " & "Pudding"`** produces **`"Summer Pudding"`**

BLANK PAGE

QUESTION 6.

4 A company employs Ahmed as a programmer.

(a) At College, before joining the company, Ahmed used two items of software for pro

- a text editor
- a compiler

Describe how he could have developed programs using these software tools.

Include in the description the terms 'object code' and 'source code'.

.....

.....

.....

.....

.....

.....

.....[3]

(b) Ahmed now uses an Integrated Development Environment (IDE) for programming.

(i) State **one** feature an IDE provides to help with the identification of syntax errors.

.....

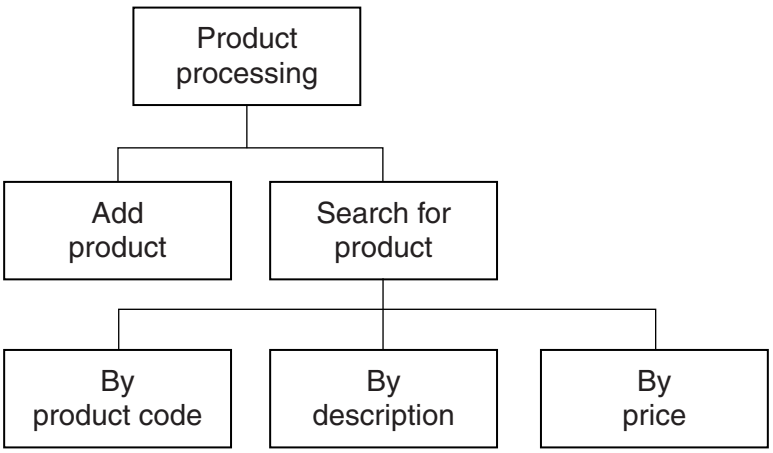
.....[1]

(ii) State **one** feature an IDE provides to carry out white box testing.

.....

.....[1]

(c) The company maintains a file of product data. Ahmed is to write a program to add a new product and search for a product based on the structure diagram shown:



The program records the following data for each product:

- product code
- product description
- product retail price

The text file `PRODUCTS` stores each data item on a separate line, as shown below:

File **PRODUCTS**

0198
Plums(10kg)
11.50
0202
Onions(20kg)
10.00
~
0376
Mango chutney(1kg)
02.99
~
0014
Mango(10kg)
12.75

The program uses the variables shown in the identifier table.

Identifier	Data type	Description
PRODUCTS	TEXT FILE	Storing the code, description and retail price for all current products
PCode	ARRAY[1:1000] OF STRING	Array storing the product codes
PDescription	ARRAY[1:1000] OF STRING	Array storing the product descriptions
PRetailPrice	ARRAY[1:1000] OF REAL	Array storing the product retail prices
i	INTEGER	Array index used by all three arrays

(i) The first operation of the program is to read all the product data held i
and write them into the three 1D arrays.

Complete the pseudocode below.

```
OPEN .....

i ← 1

WHILE .....

  READFILE ("PRODUCTS", .....)

  READFILE ("PRODUCTS", .....)

  READFILE ("PRODUCTS", .....)

  .....

  .....

ENDWHILE

CLOSE "PRODUCTS"

OUTPUT "Product file contents written to arrays"
```

[5]

When Ahmed designed the `PRODUCTS` file, he considered the alternative file structure shown opposite.

It stores one product per line in the text file.

File **PRODUCTS**

0198 Plums(10kg)	11.50
0202 Onions(20kg)	10.00
~	
0376 Mango chutney(1kg)	02.99
~	
0014 Mango(10kg)	12.75

(ii) State **one** benefit and **one** drawback of this file design.

Benefit .....

.....

Drawback .....

.....[2]

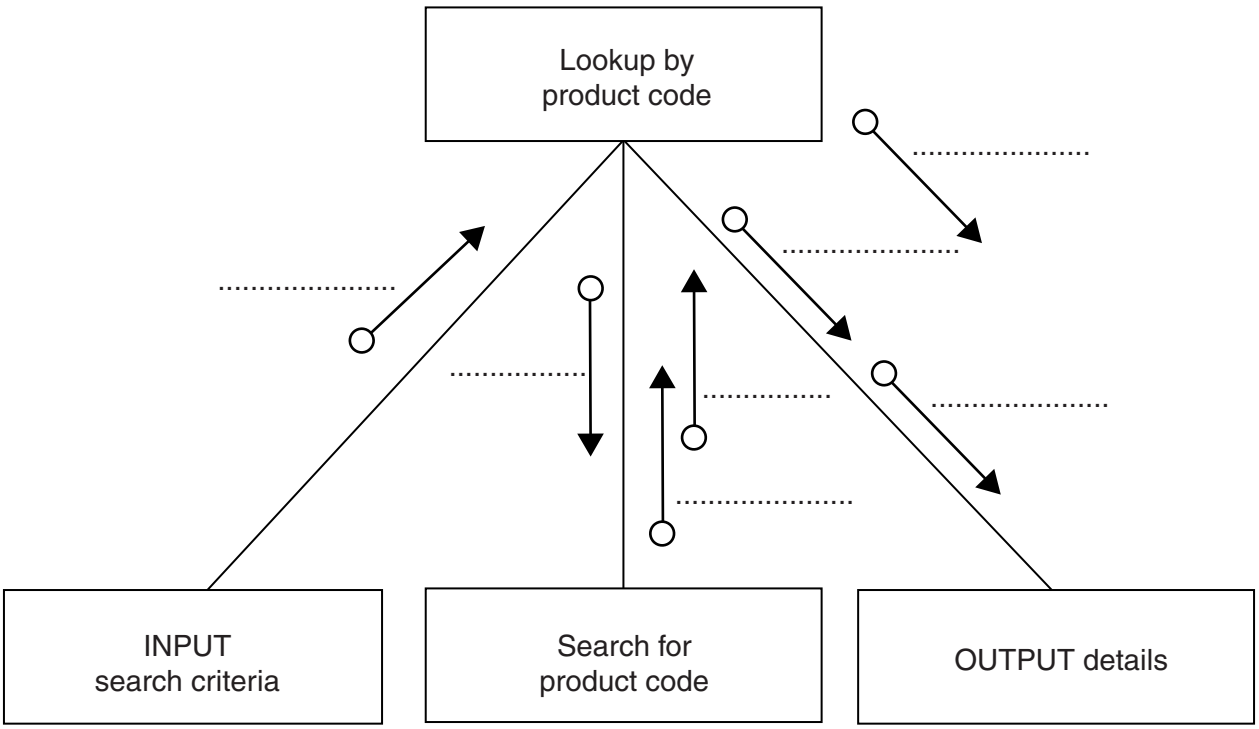
(d) To code the 'Search by product code' procedure, Ahmed draws a structure c… different stages.

The procedure uses the variables shown in the identifier table.

Identifier	Data type	Description
`SearchCode`	`STRING`	Product code input by the user
`ThisIndex`	`INTEGER`	Array index position for the corresponding product
`ThisDescription`	`STRING`	Product description found
`ThisRetailPrice`	`REAL`	Product retail price found

You can assume that before the procedure is run, all the product data is read from file `PRODUCTS` and then stored in three 1D arrays as described in **part (c)(i)**.

Label the structure chart to show the input(s) and output(s).



[4]

(e) A first attempt was made at writing the 'Search for product code' module.
Ahmed designs this as a function `ProductCodeSearch`.

The function returns an integer value as follows:

- if the product code is found, it returns the index position of the 1D array `PCode` be searched
- if the product code is not found, the function returns -1

Write **program code** for function `ProductCodeSearch`.

Visual Basic and Pascal: You should include the declaration statements for variables.
Python: You should show a comment statement for each variable used with its data type.

Programming language .....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....[6]

QUESTION 7.

5 A team keeps a record of the scores made by each of their eight players in a num

The data in the two tables below shows:

- the scores of the eight players after twenty games
- the eight player names.

	1	2	3		8
1	12	17	67		31
2	35	82	44		29
3	61	39	80		17
4	81	103	21		11
5	56	0	98		4
…					
19	45	6	81		77
20	12	11	3		6

1	Vorma
2	Ravi
3	Chada
4	Nigam
5	Bahri
6	Smith
7	Goyal
8	Lata

The team wants a computer program to input and record the player data.

(a) A programmer designs the following pseudocode for the input of a player's score from one game.

```
01  INPUT GameNumber
02  INPUT PlayerNumber
03  INPUT PlayerGameScore
04  PlayerScore[GameNumber, PlayerNumber] ← PlayerGameScore
```

Describe the data structure the programmer has used for the storage of all player scores.

..... [2]

(b) The player names are permanently stored in a text file `NAMES.TXT`, with on line. The player names will be read by the program and stored in a 1D array.

The design given in **part (a)** will be expanded so that the user is prompted for th name instead of the player number. Step `02` now becomes:

```
02.1    Read the player names from file NAMES.TXT into the array PlayerName
02.2    INPUT ThisPlayerName
02.3    Search the PlayerName array for ThisPlayerName to find the PlayerNumber
```

(i) State the computing term for the expansion of one or more steps in the original design.

..... [1]

(ii) Write the **program code** for step `02.1`

Visual Basic and Pascal: You should include the declaration statements for variables.
Python: You should show a comment statement for each variable used with its data type.

Programming language .....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

..... [4]

(iii) Program code is to be designed and written for step `02.3`

The program will use these identifiers:

Identifier	Data type	Description
`PlayerName`	`ARRAY[1 : 8] OF STRING`	Stores the player names (read from the file)
`ThisPlayerName`	`STRING`	Input by the user (step `02.2`)
`Found`	`BOOLEAN`	Flags when `ThisPlayerName` is found when searching the `PlayerName` array
`i`	`INTEGER`	Array index

Write **program code** to carry out the linear search for step `02.3`

There is no requirement to declare or comment on variables used.

Programming language .....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

..... [4]

(c) The team wants the program to produce a report, with the following specifica

The program outputs the total number of player scores that are:

- 50 and over but less than 100
- 100 or higher.

You can assume that before the section runs, the program has assigned all eight player scores to the `PlayerScore` data structure.

A first attempt at the pseudocode is shown below:

```
01 Total50 ← 0
02 Total100 ← 0
03 FOR PlayerIndex  ← 1 TO 8
04   FOR GameIndex ← 1 TO 20
05     IF PlayerScore[GameIndex, PlayerIndex] > 100
06       THEN
07         Total100 ← Total100 + 1
08       ELSE
09         IF PlayerScore[GameIndex, PlayerIndex] > 50
10           THEN
11             Total50 ← Total50 + GameIndex
12         ENDIF
13     ENDIF
14   ENDFOR
15 ENDFOR
16 OUTPUT Total50
17 OUTPUT Total100
```

(i) Describe the control structure used in lines `03` and `04` and lines `14` and `15`.

.....

.....

..... [2]

(ii) Consider the following two statements.

Write either TRUE **or** FALSE next to each statement.

Statement	TRUE or FALSE
The pseudocode considers all the scores for a player, before progressing to the next player.	
The pseudocode considers all scores in a game, before progressing to the next game.	

[1]

(iii) The programmer has made logic errors in the design.

State a line number at which an error occurs.

Explain the error or write the corrected pseudocode statement.

Line number .....

Explanation .....

..... [1]

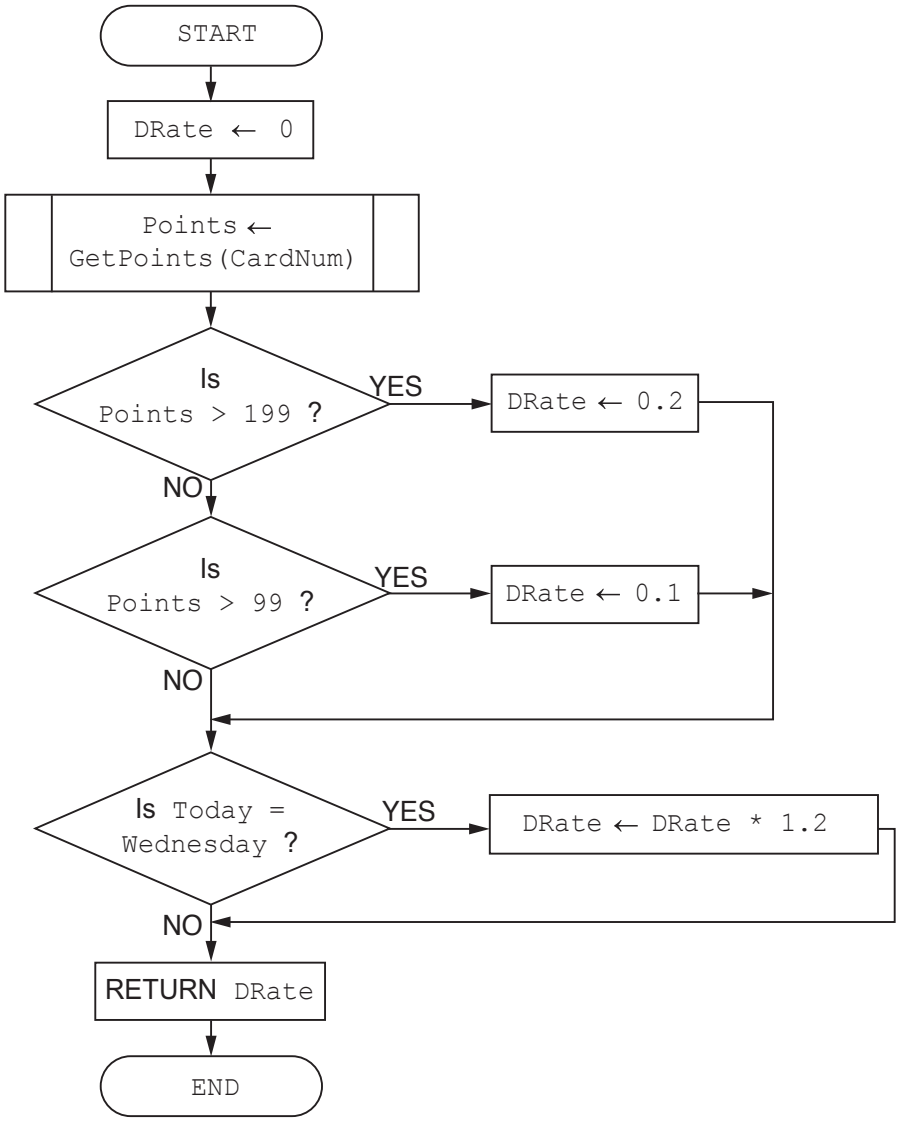
QUESTION 8.

2 Shop customers have a discount card with a unique card number. Customers co
they have bought items. The more points they have, the bigger the discount. If th
Wednesday, their discount is increased by 20%.

The function `GetDiscountRate()` takes a card number as a parameter and returns the dis
rate for a customer based on the number of points they have collected. A flowchart for the funct
is shown.

The function uses the following variables and functions.

Identifier	Data type	Description
`DRate`	`REAL`	The discount rate
`CardNum`	`STRING`	The unique customer card number
`Points`	`INTEGER`	The number of points collected
`GetPoints()`	`FUNCTION`	Takes the card number as a parameter and returns the number of points already collected
`Today()`	`FUNCTION`	Returns the day number: 1 for Monday, 2 for Tuesday etc.



(a) Write **pseudocode** to implement the `GetDiscountRate()` function.

Your solution should follow the flowchart for the function as closely as possibl
declarations should be included.

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....[8]

(b) A programmer writes the function `GetDiscountRate()` in a high-level lang

(i) A run-time error could occur when the function is used.

Name **and** describe **one** other type of error that the function could contain.

Name .....

Description .....

.....

.....

[2]

(ii) Function `GetPoints()` has not been written yet.

Name **and** describe a strategy that can be used to test `GetDiscountRate()` before the `GetPoints()` function has been written.

Name .....

Description .....

.....

.....

[2]

(c) There are different ways to minimise the risk of errors when writing programs, such as the use of constants and library routines.

(i) Identify **two** values that could be replaced by constants in the function `GetDiscountRate()`.

.....

.....[1]

(ii) Write **pseudocode** to declare **one** of the constants you have given in **part (c)(i)**.

.....[2]

(iii) Explain how the use of constants helps to minimise programming errors.

.....

.....

.....

.....[2]

(iv) Give a reason why the use of library routines helps to minimise the ris
writing a program.

.....

.....

(v) Constants and library routines help to minimise the risk of errors.

Name another way that you can minimise the risk of errors when writing a program. Explain how this helps.

Name .....

Explanation .....

.....

.....

[2]

QUESTION 9.

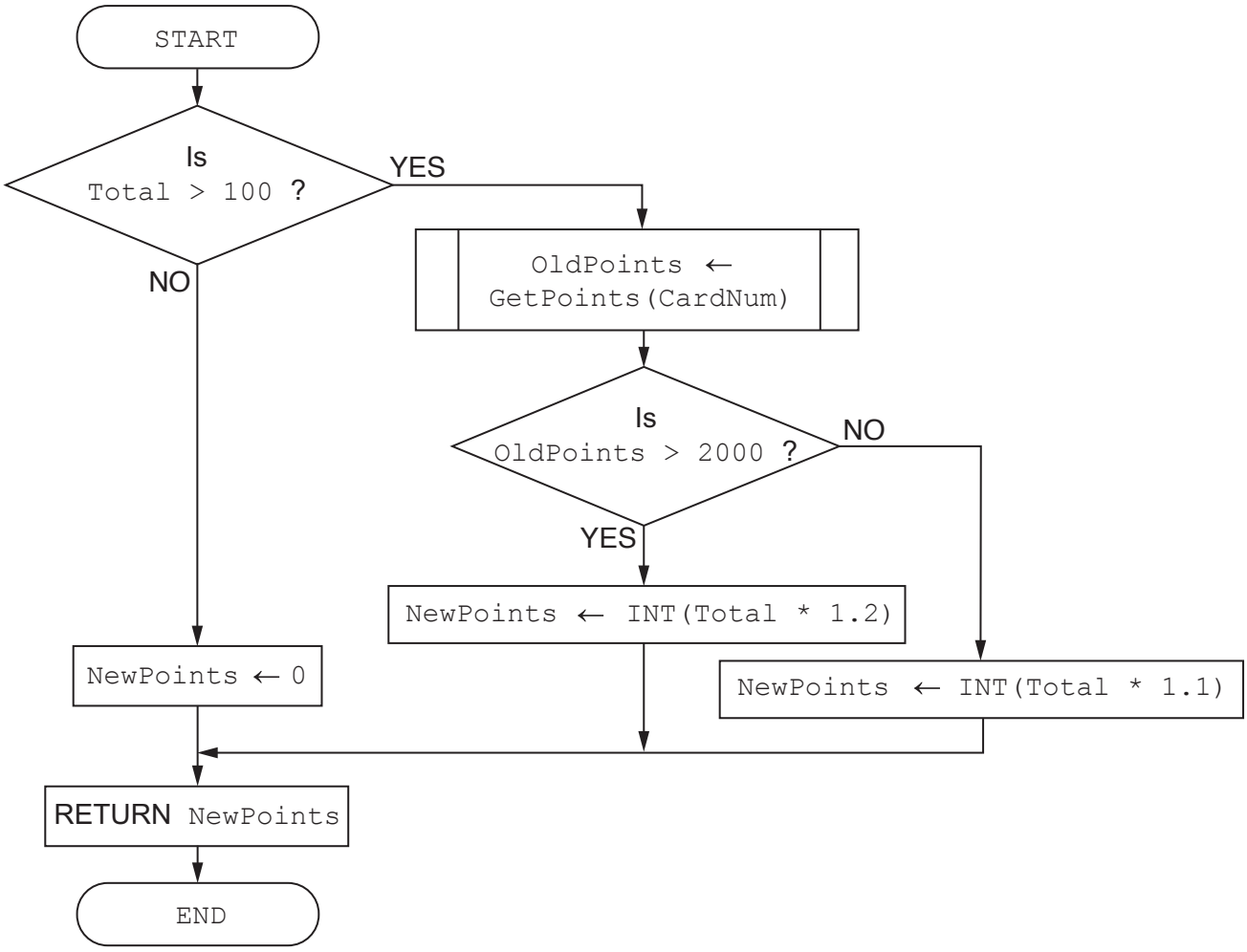
2 Shop customers have a discount card with a unique card number. Customers co
time they buy items. The number of points they collect depends on:

- the total amount they spend
- the number of points already collected.

The function `CalcPoints()` takes the card number and the total amount spent as parameters. I
returns the number of new points collected. A flowchart for the function is shown.

The function uses the following variables and functions.

Identifier	Data type	Description
`CardNum`	`STRING`	A numeric string representing the unique card number
`OldPoints`	`INTEGER`	The number of points already collected
`NewPoints`	`INTEGER`	The number of new points collected
`Total`	`REAL`	The amount spent
`GetPoints()`	`FUNCTION`	Takes the card number as a parameter and returns the number of points already collected
`INT()`	`FUNCTION`	Refer to the **Appendix** on page 16



(a) (i) Write **pseudocode** for the `CalcPoints()` function.

Your solution should follow the flowchart as closely as possible.

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....[7]

(ii) The value of the total amount spent is calculated by an Electronic Point
system. This system does not have the prices of all items. For these item
amount has to be entered manually.

A function, `GetTotal()`, prompts the user to input this value.

If the user enters a valid value greater than 0 and less than 10 000, the function returns the value. The function prompts the user to re-enter the value each time the user enters an invalid value.

Write **pseudocode** to complete the `GetTotal()` function.

```
FUNCTION GetTotal() RETURNS REAL
```

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

```
ENDFUNCTION
```

[5]

(b) The function `CalcPoints()` is written in a high-level language. It has bee
does not contain any syntax or logic errors.

(i) Name **and** describe **one** other type of error that the high-level language co
contain.

Name .....

Description .....

.....

.....

[2]

(ii) The function `CalcPoints()` is tested using white-box testing.

State **two** different values of `Total` that could be used to test different paths through the algorithm. Justify your choices.

Value .....

Justification .....

.....

.....

Value .....

Justification .....

.....

.....

[4]

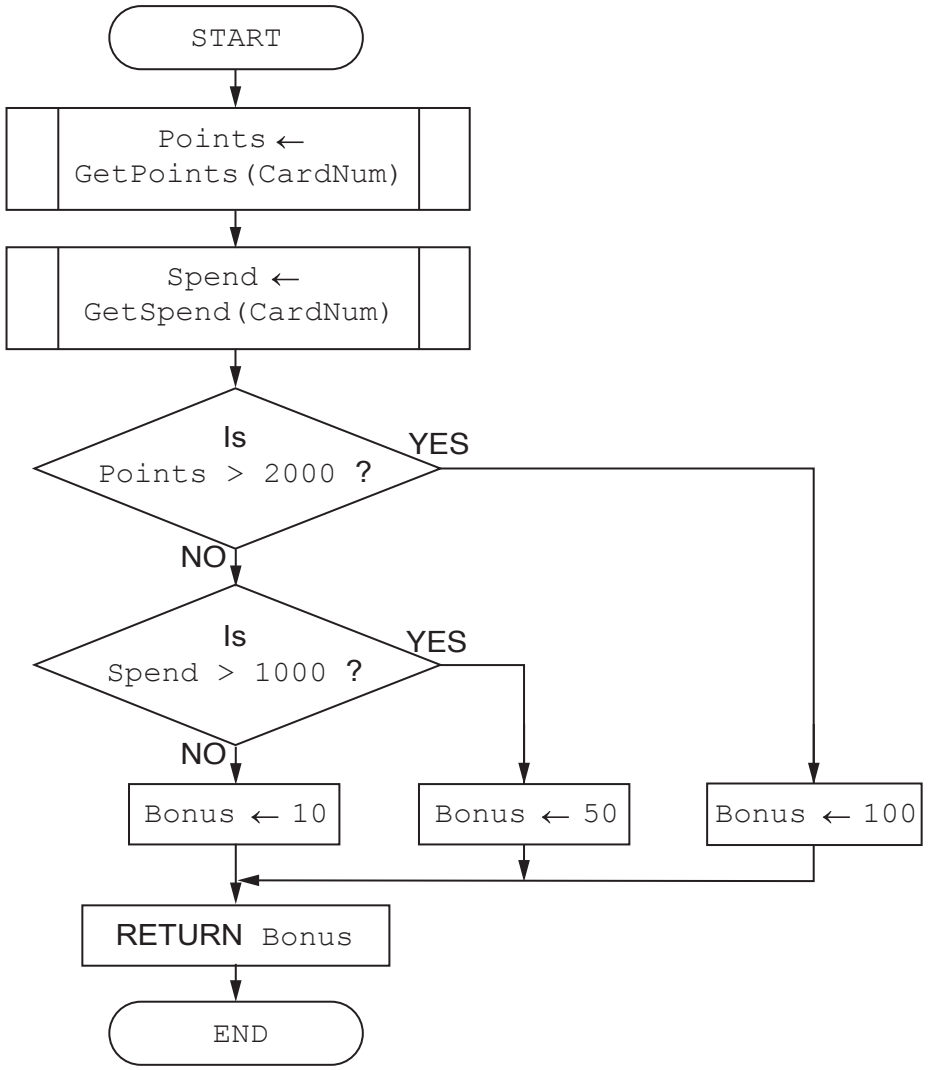
QUESTION 10.

2 Shop customers have a discount card with a unique card number. Customers co
they buy items. At the end of each year, customers are given bonus (extra) points
total amount they have spent during the year, and the number of points they have on t

The function `CalcBonus()` takes the card number as a parameter. It returns the bonus poi
the customer. A flowchart for the function is shown.

The function uses the following variables and functions.

Identifier	Data type	Description
`CardNum`	`STRING`	A numeric string representing the unique card number
`Points`	`INTEGER`	The number of points collected
`Spend`	`REAL`	The total amount that customer has spent during the year
`Bonus`	`INTEGER`	The number of bonus points
`GetPoints()`	`FUNCTION`	Takes the card number as a parameter and returns the number of points already collected
`GetSpend()`	`FUNCTION`	Takes the card number as a parameter and returns the total amount that customer has spent during the year



(a) (i) Write **pseudocode** for the `CalcBonus()` function.

Your solution should follow the flowchart for the function as closely as possib

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....[5]

(ii) The function `GetCardNumber()` prompts the user to input a card n
number input is valid.

A valid card number has 16 characters. Each character is a numeric characte
`'9'`).

Write **pseudocode** to complete the `GetCardNumber()` function.

You should refer to the function `IS_NUM()` in the **Appendix** on page 16.

```
FUNCTION GetCardNumber() RETURNS STRING
```

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

```
ENDFUNCTION
```

[6]

(b) The function `CalcBonus()` is written in a high-level language.

(i) The function is tested using black-box testing and does not contain any syntax errors.

Name **and** describe **one** other type of error that black-box testing could find.

Name .....

Description .....

.....

.....

[2]

(ii) The function `CalcBonus()` is tested using white-box testing.

State **two** different pairs of values for `Spend` and `Points` that can be used to test different paths through the function. Justify your choices.

`Spend` ..... `Points` .....

Justification .....

.....

.....

`Spend` ..... `Points` .....

Justification .....

.....

.....

[4]

(c) Name **two** types of program maintenance **and** state the reason why each is needed.

Name .....

Reason .....

.....

.....

Name .....

Reason .....

.....

.....

[4]

QUESTION 11.

4 A program is being written to control the operation of a portable music player. The program controls the output volume.

The player has two buttons, one to increase the volume and one to decrease it. When a button is pressed, a procedure `Button()` is called with a parameter value representing the button as follows:

Button	Parameter value
Volume increase	10
Volume decrease	20

For example, pressing the volume increase button three times followed by pressing the volume decrease button once would result in the calls:

```
CALL Button(10) // VolLevel increased by 1
CALL Button(10) // VolLevel increased by 1
CALL Button(10) // VolLevel increased by 1
CALL Button(20) // VolLevel decreased by 1
```

The program makes use of two global variables of type `INTEGER` as follows:

Variable	Description
`VolLevel`	The current volume setting. This must be in the range 0 to 49.
`MaxVol`	A value that can be set to limit the maximum value of `VolLevel`, in order to protect the user's hearing. A value in the range 1 to 49 indicates the volume limit. A value of zero indicates that no volume limit has been set.

The procedure `Button()` will modify the value of `VolLevel` depending on which button has been pressed and whether a maximum value has been set.

(a) Write **pseudocode** for the procedure `Button()`. Declare any additional var

The value of `MaxVol` should **not** be changed within the procedure.

Parameter validation is **not** necessary.

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

..... [6]

(b) The procedure `Button()` is to be tested using black-box testing.

Fill in the gaps below to define **three** tests that could be carried out.

TEST 1 – `VolLevel` is changed

Parameter value: 10

`MaxVol`: .....

`VolLevel` value before call to `Button()`: 48

`VolLevel` expected value after call to `Button()`: .....

TEST 2 – `VolLevel` is **not** changed

Parameter value: 10

`MaxVol`: 34

`VolLevel` value before call to `Button()`: .....

`VolLevel` expected value after call to `Button()`: .....

TEST 3 – `VolLevel` is **not** changed

Parameter value: .....

`MaxVol`: 40

`VolLevel` value before call to `Button()`: 0

`VolLevel` expected value after call to `Button()`: .....

[6]

(c) The testing stage is part of the program development cycle.

(i) The program for the music player has been completed. The program does not have any syntax errors, but testing could reveal further errors.

Identify **and** describe **one different** type of error that testing could reveal.

Type .....

Description .....

.....

.....

[2]

(ii) Stub testing is a technique often used in the development of modular programs.

Describe the technique.

.....

.....

.....

.....

.....

..... [3]