Cambridge Assessment International Education

Cambridge International AS & A Level

CANDIDATE NAME: Muhammad Azfar Sikander

CENTRE NUMBER | | | | | |

CANDIDATE NUMBER | | | | |

CS BY AQIB KHAN

4026563148

COMPUTER SCIENCE

9618/22

Paper 2 Fundamental Problem-solving and Programming Skills

October/November 2023

2 hours

You must answer on the question paper.

You will need: Insert (enclosed)

INSTRUCTIONS

- Answer **all** questions.
- Use a black or dark blue pen.
- Write your name, centre number and candidate number in the boxes at the top of the page.
- Write your answer to each question in the space provided.
- Do **not** use an erasable pen or correction fluid.
- Do **not** write on any bar codes.
- You may use an HB pencil for any diagrams, graphs or rough working.
- Calculators must **not** be used in this paper.

INFORMATION

- The total mark for this paper is 75.
- The number of marks for each question or part question is shown in brackets [ ].
- No marks will be awarded for using brand names of software packages or hardware.
- The insert contains all the resources referred to in the questions.

THE AQIB KHAN
AN INSTITUTE OF O/A LEVEL

This document has **20** pages. Any blank pages are indicated.

DC (LK/SG) 315868/1
© UCLES 2023

TAK (The AQIB KHAN)- An institute of O/A level

Aki.Khan CS By AQIB Khan CS By AQIB KHAN TheAkikhan.com +92 3145030717

Refer to the **insert** for the list of pseudocode functions and operators.

1 A shop sells car accessories. A customer order is created if an item cannot be supplied from current stock. A program is being developed to create and manage the customer orders.

(a) The following identifier table shows some of the data that will be stored for each order.

Complete the identifier table by adding meaningful variable names and appropriate data types.

Example value	Explanation	Variable name	Data type
`"Mr Khan"`	The name of the customer	CustomerName	STRING
`3`	The number of items in the order	OrderQuantity	INTEGER
`TRUE`	To indicate whether this is a new customer	NewCustomer	BOOLEAN
`15.75`	The deposit paid by the customer	DepositPaid	REAL

[4]

(b) Other variables in the program have example values as shown:

Variable	Example value
`Total`	`124.00`
`DepRate`	`2.00`
`Description`	`"AB12345:Cleaning Brush (small)"`

Complete the table by evaluating each expression using the example values.

Expression	Evaluates to
`(Total * DepRate) + 1.5`	249.50
`RIGHT(Description, 7)`	"(small)"
`(LENGTH(Description) - 8) > 16`	TRUE
`NUM_TO_STR(INT(DepRate * 10)) & '%'`	"20%"

[4]

CS BY AQIB KHAN

THE AQIB KHAN AN INSTITUTE OF O/A LEVEL

(c) The data that needs to be stored for each customer order in part **(a)** is not all of the same type.

Describe an effective way of storing this data for many customer orders while the program is running.

First, define a record type for data of one customer, using the record, define a 1D array with each element of data type record. This way, many different data types will be stored in a single element.

[3]

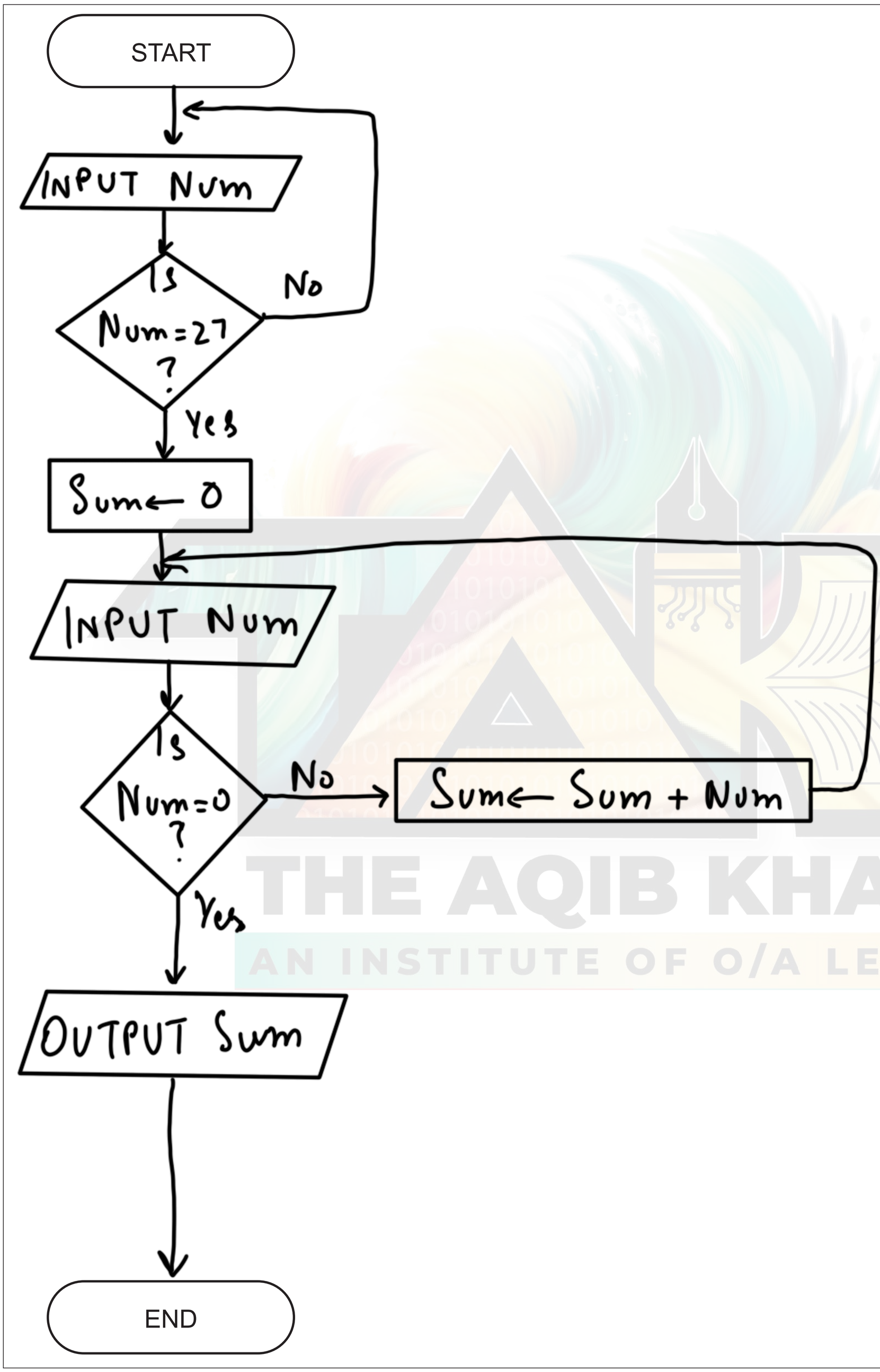
CS BY AQIB KHAN

THE AQIB KHAN
AN INSTITUTE OF O/A LEVEL

2 An algorithm will:

1. input a sequence of integer values, one at a time
2. ignore all values until the value 27 is input, then sum the remaining values in the sequence
3. stop summing values when the value 0 is input and then output the sum of the values.

(a) Draw a program flowchart to represent the algorithm.



[5]

© UCLES 2023

THE AQIB KHAN
AN INSTITUTE OF O/A LEVEL
CS BY AQIB KHAN
TAK (The AQIB KHAN) An institute of O/A level
Aki.Khan CS By AQIB Khan CS By AQIB KHAN TheAkikhan.com +92 3145030717

(b) The solution to the algorithm includes iteration.

Give the name of a suitable loop structure that could be used.

Justify your answer.

Name Conditional loop

Justification Loop keeps iterating until a condition is either true or false.

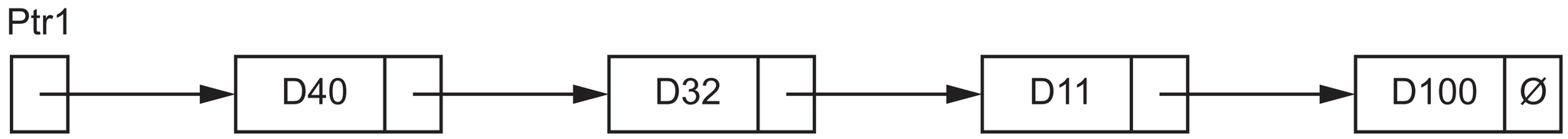
[2]

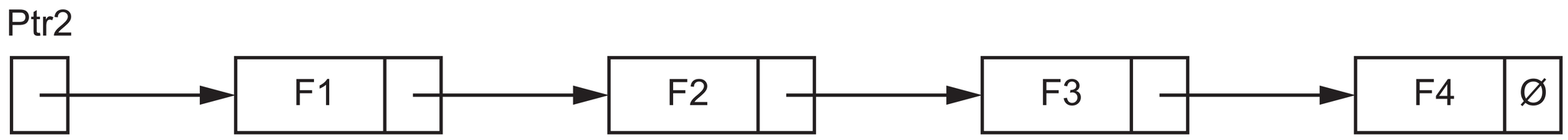
CS BY AQIB KHAN

THE AQIB KHAN
AN INSTITUTE OF O/A LEVEL

3 The diagram represents a linked list Abstract Data Type (ADT).

- Ptr1 is the start pointer. Ptr2 is the free list pointer.
- Labels D40, D32, D11 and D100 represent the data items of nodes in the list.
- Labels F1, F2, F3 and F4 represent the data items of nodes in the free list.
- The symbol Ø represents a null pointer.





(a) The linked list is implemented using two variables and two 1D arrays as shown.

The pointer variables and the elements of the Pointer array store the indices (index numbers) of elements in the Data array.

Complete the diagram to show how the linked list as shown above may be represented using the variables and arrays.

Variable	Value
Start_Pointer	4
Free_List_Pointer	5

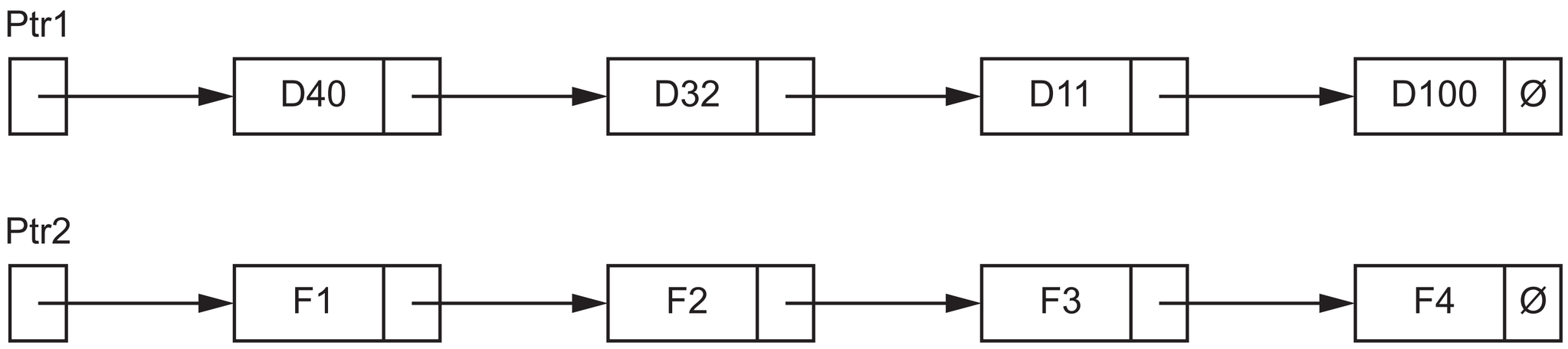
Index	Data array	Pointer array
1	D32	2
2	D11	3
3	D100	0
4	D40	1
5	F1	6
6	F2	7
7	F3	8
8	F4	0

[5]

CS BY AQIB KHAN

THE AQIB KHAN
AN INSTITUTE OF O/A LEVEL

(b) The original linked list is to be modified. A new node D6 is inserted between nodes D32 and D11.



The algorithm required is expressed in four steps as shown.

Complete the steps.

1. Assign the data item ..... D6 ..... to ..... F1 ..... .
2. Set the ..... pointer ..... of this node to point to ..... D11 ..... .
3. Set Ptr2 to point to ..... F2 ..... .
4. Set pointer of ..... D32 ..... to point to ..... D6 ..... .

[4]

CS BY AQIB KHAN
THE AQIB KHAN
AN INSTITUTE OF O/A LEVEL

4 A procedure `Count()` will:

1. input a value (all values will be positive integers)
2. count the number of odd values and count the number of even values
3. repeat from step 1 until the value input is 99
4. output the two count values, with a suitable message.

The value 99 must not be counted.

(a) Write pseudocode for the procedure `Count()`.

```
PROCEDURE Count()
    DECLARE Even,Odd ,Num : INTEGER

    Odd←0, Even←0
    INPUT Num
    WHILE Num <> 99 DO
      IF Num >0 THEN
          IF MOD(Num,2)=0 THEN
              Even← Even+1
          ELSE
              Odd← Odd +1
          ENDIF
      ENDIF

      INPUT Num
    ENDWHILE

    OUTPUT "Number of even: ", Even
    OUTPUT " Number of odd: ", Odd
ENDPROCEDURE
```

[6]

CS BY AQIB KHAN
THE AQIB KHAN
AN INSTITUTE OF O/A LEVEL

(b) The procedure `Count()` is to be tested.

Typical test data would consist of odd and even values, for example:

23, 5, 64, 100, 2002, 1, 8, 900, 99

The purpose of this test would be to test a typical mix of even and odd values and check the totals.

Give **three** test data sequences that may be used to test **different** aspects of the procedure.

Do **not** include invalid data.

Sequence 1:

Test data 40, 60, 70, 99

Purpose of test. To test only even

Sequence 2:

Test data 1, 7, 97, 99

Purpose of test. To test only odd

Sequence 3:

Test data 99

Purpose of test. To test counting in program.

[3]

CS BY AQIB KHAN
THE AQIB KHAN
AN INSTITUTE OF O/A LEVEL

5 A global 1D array of integers contains four elements, which are assigned values as shown:

```
Mix[1] ← 1
Mix[2] ← 3
Mix[3] ← 4
Mix[4] ← 2
```

A procedure `Process()` manipulates the values in the array.

The procedure is written in pseudocode:

```
PROCEDURE Process(Start : INTEGER)
   DECLARE Value, Index, Count : INTEGER

   Index ← Start
   Count ← 0

   REPEAT
      Value ← Mix[Index]
      Mix[Index] ← Mix[Index] - 1
      Index ← Value
      Count ← Count + 1
   UNTIL Count = 5

   Mix[4] ← Count * Index

ENDPROCEDURE
```

Complete the trace table on the opposite page by dry running the procedure when it is called as follows:

```
CALL Process(3)
```

CS BY AQIB KHAN
THE AQIB KHAN
AN INSTITUTE OF O/A LEVEL

Index	Value	Count	Mix[1]	Mix[2]	Mix[3]	Mix[4]
3	4	0	1	3	4	2
4		1			3	
2	2	2				1
3	3	3		2		
3	3	4			2	
2	2	5			1	10

[6]

CS BY AQIB KHAN

THE AQIB KHAN
AN INSTITUTE OF O/A LEVEL

6 **(a)** A procedure `CreateFiles()` will take two parameters:

- a string representing a file name
- an integer representing the number of files to be created.

The procedure will create the number of text files specified.

Each file is given a different name. Each file name is formed by concatenating the file name with a suffix based on the file number. The suffix is always three characters.

For example, the call `CreateFiles("TestData", 3)` would result in the creation of the three files, TestData.001, TestData.002 and TestData.003.

Each file will contain a single line. For example, file TestData.002 would contain the string:

```
This is File TestData.002
```

Write pseudocode for `CreateFiles()`.

Assume both parameters are valid and that the integer value is between 1 and 999, inclusive.

```
PROCEDURE CreateFiles(File:STRING,Amount:INTEGER)
    DECLARE Suffix,Line,Name:STRING
    DECLARE i:INTEGER

    FOR i←1 TO Amount
        Suffix←NUM_TO_STR(i)
        CASE OF LENGTH(Suffix)
            =1:Suffix←".00"&Suffix
            =2:Suffix←".0"&Suffix
        ENDCASE
        Name←File&Suffix
        Line←"This is File "&File&Suffix
        OPENFILE Name FOR WRITE
        WRITEFILE Name,Line
        CLOSEFILE Name
    NEXT i
ENDPROCEDURE
```

[6]

CS BY AQIB KHAN

THE AQIB KHAN
AN INSTITUTE OF O/A LEVEL

(b) A module `CheckFiles()` will count the number of files produced by `CreateFiles()` in part **(a)**.

`CheckFiles()` will take a string representing a file name and return the number of files found.

(i) Identify the type of module that should be used for `CheckFiles()`.

Function ..... [1]

(ii) Write the module header for `CheckFiles()`.

FUNCTION CheckFiles(File : STRING)
RETURNS INTEGER ..... [1]

(iii) State the file mode that should be used in `CheckFiles()`.

Read ..... [1]

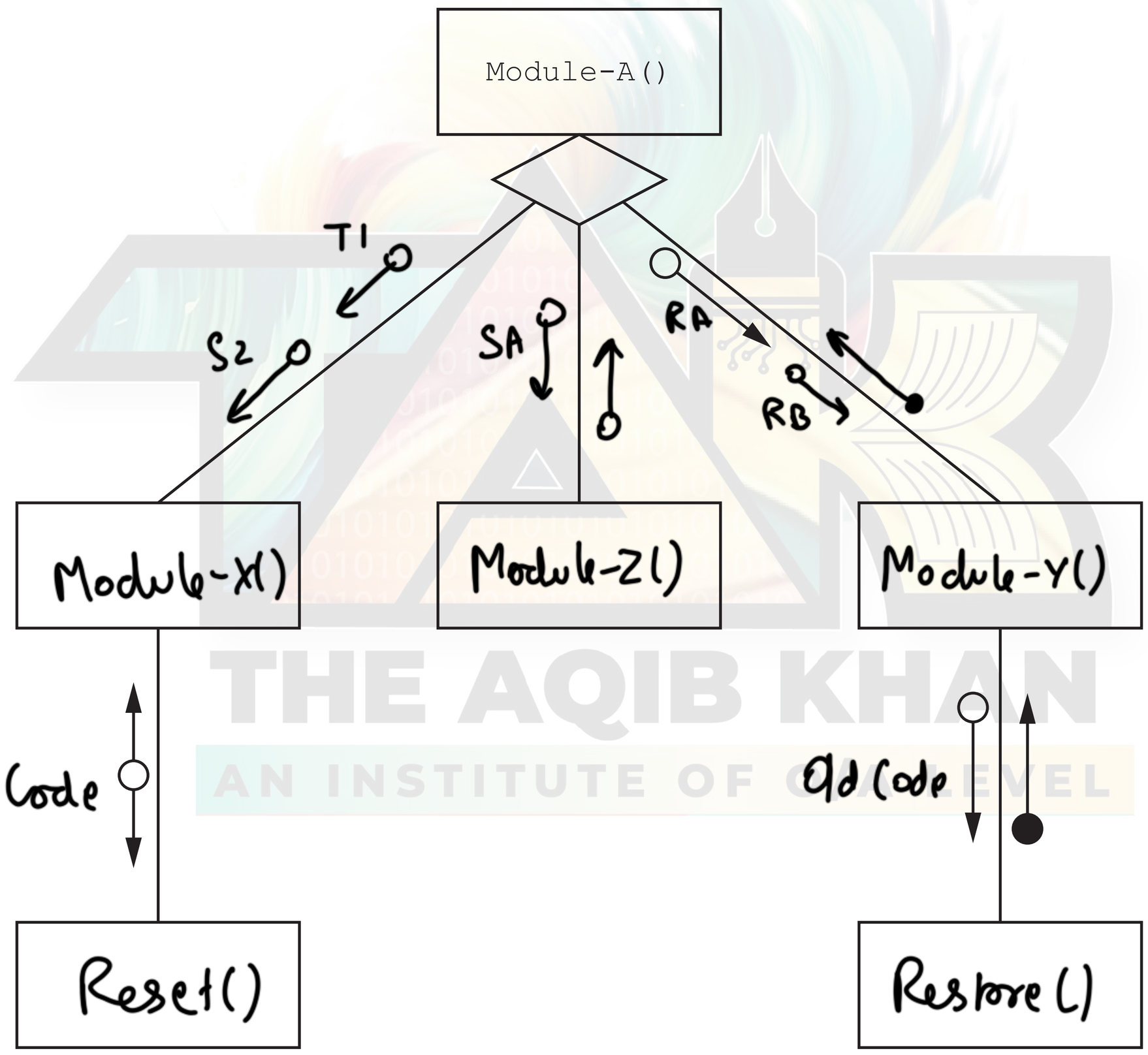
CS BY AQIB KHAN
THE AQIB KHAN
AN INSTITUTE OF O/A LEVEL

7 A program contains six modules:

Pseudocode module header
`PROCEDURE Module-A()`
`PROCEDURE Module-X(T1 : INTEGER, S2 : REAL)`
`PROCEDURE Reset(BYREF Code : INTEGER)`
`FUNCTION Restore(OldCode : INTEGER) RETURNS BOOLEAN`
`FUNCTION Module-Y(RA : INTEGER, RB : BOOLEAN) RETURNS BOOLEAN`
`FUNCTION Module-Z(SA : INTEGER) RETURNS INTEGER`

`Module-X()` calls `Reset()`
`Module-Y()` calls `Restore()`

(a) Complete the structure chart for these modules.



[4]

(b) Explain the meaning of the diamond symbol as used in the diagram in part **(a)**.

Selection, Module-A() will call only one of the 3 modules.

[2]

© UCLES 2023 9618/22/O/N/23
TAK (The AQIB KHAN) An institute of O/A level
Aki.Khan CS By AQIB Khan CS By AQIB KHAN TheAkikhan.com +92 3145030717

BLANK PAGE

CS BY AQIB KHAN

THE AQIB KHAN
AN INSTITUTE OF O/A LEVEL

8 A class of students are developing a program to send data between computers. Many computers are connected together to form a wired network. Serial ports are used to connect one computer to another.

Each computer:

- is assigned a unique three-digit ID
- has three ports, each identified by an integer value
- is connected to between one and three other computers.

Data is sent as individual message strings.
Each string contains the destination ID (the ID of the computer that is to receive the message) followed by the data:

```
<DestinationID><Data>
```

Messages may pass through several computers on the way to their destination.
When a message arrives at a computer, that is **not** the destination, the program needs to forward it on to another computer using one of its serial ports.

The port to use is obtained from information that is stored in an array `RouteTable`.

`RouteTable` is a global 2D array of integers. It is declared in pseudocode as follows:

```
DECLARE RouteTable : ARRAY[1:6,1:3] OF INTEGER
```

The values in the first two columns of `RouteTable` define a range of ID values.
Column 3 gives the corresponding port number to use when forwarding the message to a computer with an ID within this range.

For example, the contents of `RouteTable` could be:

	Column 1	Column 2	Column 3
Row 1	100	199	1
Row 2	200	259	2
Row 3	-1	<undefined>	<undefined>
Row 4	260	399	2
Row 5	400	599	3
Row 6	600	999	1

In this example, a message that arrives with a `DestinationID` of "283" will be forwarded using port 2.

Row 3 in the example shows an unused row. These may occur anywhere. Unused rows have the column 1 element set to −1. The value of unused elements in the other two columns is undefined.

THE AQIB KHAN
AN INSTITUTE OF O/A LEVEL
CS BY AQIB KHAN

The programmer has defined the first program module as follows:

Module	Description
`GetPort()`	• takes a `DestinationID` as a parameter of type string • searches for the range corresponding to the `DestinationID` in the array • returns the port number, or returns −1 if no corresponding range is found

(a) Write pseudocode for module `GetPort()`.

Assume `DestinationID` contains a valid three-digit string.

```
FUNCTION GetPort(ID: STRING) RETURNS INTEGER
    DECLARE i

    FOR i ← 1 TO 6
        IF RouteTable[i,1] >= 0 THEN
            IF STR_TO_NUM(ID) > RouteTable[i,1] AND
            STR_TO_NUM(ID) < RouteTable[i,2] THEN
                RETURN RouteTable[i,3]
            ENDIF
        ENDIF
    NEXT i

    RETURN -1
END FUNCTION
```

[7]

CS BY AQIB KHAN
THE AQIB KHAN
AN INSTITUTE OF O/A LEVEL

(b) Copies of the same program will run on each computer. The program contains a global variable `MyID` of type string, which contains the unique ID of the computer in which the program is running.

When messages are received, they are placed on one of two stacks. Stack 1 is used for messages that have reached their destination and stack 2 is used for messages that will be forwarded on to another computer.

Additional modules are defined:

Module	Description

`StackMsg()`
(already written)
	• takes two parameters:
	  ○ a string representing a message
	  ○ an integer representing the stack number
	• adds the message to the required stack
	• returns `TRUE` if the message is added to the required stack, otherwise returns `FALSE`

`ProcessMsg()`	• takes a message as a parameter of type string • ignores any message with a zero-length data field • extract the `DestinationID` from the message • checks whether the `DestinationID` is this computer or whether the message is to be forwarded • uses `StackMsg()` to add the message to the appropriate stack • outputs an error if the message could not be added to the stack

CS BY AQIB KHAN
THE AQIB KHAN
AN INSTITUTE OF O/A LEVEL

Write pseudocode for module `ProcessMsg()`.

Module `StackMsg()` must be used.

```
PROCEDURE ProcessMsg(Msg : STRING)
    DECLARE ID : STRING
    DECLARE Processed : BOOLEAN
    IF LENGTH(Msg) >= 4 THEN
        ID ← LEFT(Msg, 3)
        IF ID = MyID THEN
            Processed ← StackMsg(Msg, 1)
        ELSE
            Processed ← StackMsg(Msg, 2)
        ENDIF

        IF NOT Processed THEN
            OUTPUT "Error, Message not added to
                    Stack"
        ENDIF

ENDPROCEDURE
```

[7]

CS BY AQIB KHAN

THE AQIB KHAN
AN INSTITUTE OF O/A LEVEL

(c) The program contains a module `GetFile()` which receives text files sent from another computer.

Lines from the file are sent one at a time. Each message contains one line and `ProcessMsg()` from part **(b)** adds each message as it is received onto stack 1.

Module `GetFile()` removes messages from stack 1 and writes the data to a text file.

There is a problem. Under certain circumstances, the received file does not appear as expected.

Assume that while a file is being received `ProcessMsg()` receives only messages containing lines from the file.

(i) Describe the circumstances and explain the problem.

Circumstances More than one line is transmitted.

Explanation Stack operates using FILO, lines would appear in reverse order.

[3]

(ii) Suggest a more appropriate Abstract Data Type that could be used to store the messages that would not have the same problem.

Queue [1]

THE AQIB KHAN
AN INSTITUTE OF O/A LEVEL
CS BY AQIB KHAN

Permission to reproduce items where third-party owned material protected by copyright is included has been sought and cleared where possible. Every reasonable effort has been made by the publisher (UCLES) to trace copyright holders, but if any items requiring clearance have unwittingly been included, the publisher will be pleased to make amends at the earliest possible opportunity.

To avoid the issue of disclosure of answer-related information to candidates, all copyright acknowledgements are reproduced online in the Cambridge Assessment International Education Copyright Acknowledgements Booklet. This is produced for each series of examinations and is freely available to download at www.cambridgeinternational.org after the live examination series.

Cambridge Assessment International Education is part of Cambridge Assessment. Cambridge Assessment is the brand name of the University of Cambridge Local Examinations Syndicate (UCLES), which is a department of the University of Cambridge.

TAK (The AQIB KHAN) An institute of O/A level
Aki.Khan CS By AQIB Khan CS By AQIB KHAN TheAkikhan.com +92 3145030717