AS · Cambridge (CIE) · Computer Science

42 mins 14 questions

Exam Questions

Constructs

Selection / Iteration

1 The final CASE condition (> 200) in the pseudocode example could be replaced with a keyword.

Give the keyword.

Answer

Mark Scheme and Guidance

OTHERWISE

(1 mark)

2 (a) One type of run-time error can cause a program to stop responding ('freezing').

Identify a particular type of programming construct that can generate this type of error and explain why it occurs.

Construct .....

Explanation.....

Answer

Mark Scheme and Guidance

MP1 Construct: A (pre/post) conditional loop

MP2 Explanation: The terminating condition is never satisfied

(2 marks)

(b) The function Process() contains a selection construct using a CASE structure.

Write pseudocode using a single selection construct with the same functionality **without** using a CASE structure.

Answer

Mark Scheme and Guidance

Example solution:

```
IF Index Mod 2 = 0 THEN
ReturnValue ← TO_UPPER(RIGHT(Label, Count))
ELSE
ReturnValue ← "****"
ENDIF
```

Mark as follows:
MP1 IF...THEN...ELSE...ENDIF
MP2 Both correct assignments **and** the correct test/logic

(2 marks)

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

The following table contains pseudocode examples.

Each example may contain statements that relate to one or more of the following:

- selection
- iteration (repetition)
- subroutine (procedure or function).

Complete the table by placing **one or more** ticks ('✓') in each row.

Pseudocode example	Selection	Iteration	Subroutine

FOR Index ← 1 TO 3
IF Safe[Index] = TRUE THEN
Flap[Index] ← 0
ENDIF
NEXT Index			

CASE OF Compound(3)			
REPEAT UNTIL AllDone() = TRUE			
WHILE Result[3] <> FALSE			

Answer

Pseudocode example	Selection	Iteration	Subroutine

FOR Index ← 1 TO 3
IF Safe[Index] = TRUE THEN
Flap[Index] ← 0
ENDIF
NEXT Index
	✓	✓	

CASE OF Compound(3)	✓		✓
REPEAT UNTIL AllDone() = TRUE		✓	
WHILE Result[3] <> FALSE		✓	

(4 marks)

4 An algorithm will:

1. prompt and input a sequence of 100 integer values, one at a time
2. sum the positive integers
3. output the result of the sum.

The algorithm requires the use of basic constructs. One of these is sequence.

Identify **one other** basic construct required by the algorithm **and** describe how it is used.

Construct.....

Use.....

Answer

Mark Scheme and Guidance

MP1 Construct: Iteration / Repetition
MP2 Use: To loop through all 100 inputs // To loop 100 times

ALTERNATIVE:

MP1 Construct: Selection
MP2 Use: To test whether the value input is positive

(2 marks)

5 A global 1D array of strings contains three elements which are assigned values as shown:

```
Data[1] ← "aaaaaa"
Data[2] ← "bbbbbb"
Data ← [3] "cccccc"
```

Procedure Process() manipulates the values in the array.

The procedure is written in pseudocode as follows:

```
PROCEDURE Process(Format : STRING)
DECLARE Count, Index, L : INTEGER
DECLARE Result : STRING
DECLARE C : CHAR

Result ← "****"

FOR ← Count 1 TO LENGTH(Format) STEP 2
C ← MID(Format, Count, 1)
L ← STR_TO_NUM(MID(Format, Count + 1, 1))

Index (Count + 1) DIV 2

CASE OF C
'X' : ← Result TO_UPPER(Data[Index])
'Y' : ← Result TO_LOWER(Data[Index])
'Z' : ← Result "**" & Data[Index]
ENDCASE

Data[Index] LEFT(Result, L)
NEXT Count

ENDPROCEDURE
```

The procedure is to be modified. If variable C is assigned a value other than 'X', 'Y' or 'Z', then procedure Error() is called and passed the value of variable C as a parameter.

This modification can be implemented by adding a **single line** of pseudocode.

(i) Write the single line of pseudocode.

[1]

(ii) State where this new line should be placed.

[1]

Answer

Mark Scheme and Guidance

(i)

OTHERWISE : CALL Error(C)

(ii)

After the 'Z' clause in the CASE construct // before the ENDCASE

(2 marks)

6 A program is being designed in pseudocode.

The program contains a global 1D array Data of type string containing 200 elements.

The first element has the index value 1.

A procedure Process() is written to initialise the values in the array:

```
PROCEDURE Process(Label : STRING)
DECLARE Index : INTEGER
Index ← 0
INPUT Data[Index]
WHILE Index < 200
Index ← Index + 1
CASE OF (Index MOD 2)
0 : Data[Index] ← TO_UPPER(Label)
1 : Data[Index] ← TO_LOWER(Label)
OTHERWISE : OUTPUT "Alarm 1201"
ENDCASE
NEXT Index
OUTPUT "Completed " & Index & " times"
ENDPROCEDURE
```

The procedure contains a statement that is not needed.

Identify the pseudocode statement **and** explain why it is not needed.

Statement .....

Explanation .....

Answer

Mark Scheme and Guidance

One mark per point:

Statement:

- The OTHERWISE statement

Explanation:

- The result of MOD 2 can only be 0 or 1

(2 marks)

7 A program uses a global 1D array of type string and a text file.

An algorithm that forms part of the program is expressed as follows:

- copy the first line from the file into the first element of the array
- copy the second line from the file into the second element of the array
- continue until all lines in the file have been copied into the array.

Sequence is one programming construct. Identify **one other** programming construct that will be required when the algorithm from part **(a)** is converted into pseudocode **and** explain its use.

Construct.....

Use.....

Answer

Mark Scheme and Guidance

One mark per point:

Construct: a conditional loop

Use: To keep repeating until the **end of the file is reached**

ALTERNATIVE:

Construct: a selection statement

Use: To test / check the value returned by the EOF() function

(2 marks)

8 A record structure is declared to hold data relating to components being produced in a factory:

```
TYPE Component
DECLARE Item_ID : STRING
DECLARE Reject : BOOLEAN
DECLARE Weight : REAL
ENDTYPE
```

The factory normally produces a batch (or set) of 1000 components at a time. A global array is declared to store 1000 records for a batch:

```
DECLARE Batch : ARRAY [1:1000] OF Component
```

Two global variables contain the minimum and maximum acceptable weight for each component. The values represent an inclusive range and are declared as:

```
DECLARE Min, Max : REAL
```

A program uses a variable ThisIndex as the array index to access a record.

Write a pseudocode clause to check whether or **not** the weight of an individual component is within the acceptable range.

Answer

Mark Scheme and Guidance

Example solution using AND

```
IF Batch[ThisIndex].Weight >= Min AND
Batch[ThisIndex].Weight <= Max THEN
```

Alternative solution using OR

```
IF Batch[ThisIndex].Weight < Min OR
Batch[ThisIndex].Weight > Max THEN
```

Mark as follows:

1. Reference to Batch[ThisIndex].Weight
2. A valid check for one boundary
3. A valid check for other boundary with correct logic operator

(3 marks)

9 A program is being designed in pseudocode.

The program contains the following declaration:

```
DECLARE Data : ARRAY[1:1000] OF STRING
```

A procedure ArrayInitialise() is written to initialise the values in the array:

```
PROCEDURE ArrayInitialise(Label : STRING)
DECLARE Index : INTEGER
Index ← 1
WHILE Index <= 1000
CASE OF (Index MOD 2)
0 : Data[Index] ← FormatA(Label)
Index ← Index + 1
1 : Data[Index] ← FormatB(Label)
Index ← Index + 1
ENDCASE
ENDWHILE
ENDPROCEDURE
```

Functions FormatA() and FormatB() apply fixed format case changes to the parameter string.

The design of the procedure does **not** use the most appropriate loop construct.

Suggest a **more appropriate** construct that could be used **and** explain your choice.

Construct .....

Explanation .....

Answer

Mark Scheme and Guidance

One mark per point

- Count-controlled loop
- the number of iterations is known

(2 marks)

10 (a) Refer to the insert for the list of pseudocode functions and operators.

The following pseudocode represents part of the algorithm for a program:

```
CASE OF ThisValue
< 30 : Level ← "Low" // less than 30
Check ← 1
< 20 : Level ← "Very Low" // less than 20
Check ← ThisValue / 2
30 TO 40 : Level ← "Medium" // between 30 and 40
Check ← ThisValue / 3
Data[ThisValue] ← Data[ThisValue] + 1
> 40 : Level ← "High"
ENDCASE
```

Complete the table by writing the answer for each row:

	Answer
The value assigned to Level when ThisValue is 40	
The value assigned to Check when ThisValue is 36	
The value assigned to Level when ThisValue is 18	
The number of elements in array Data that may be incremented	

Answer

Mark Scheme and Guidance

One mark per row:

	Answer
The value assigned to Level when ThisValue is 40	**"Medium"**
The value assigned to Check when ThisValue is 36	**12**
The value assigned to Level when ThisValue is 18	**"Low"**
The number of elements in array Data that may be incremented	**11**

(4 marks)

(b) The pseudocode contains four assignments to variable Level. One of these assignments will never be performed.

Identify this assignment **and** explain why this is the case.

Answer

Mark Scheme and Guidance

One mark for identifying assignment:

MP1 Level ← "Very Low" // the level is assigned value “very low”

Explanation points:

MP2 because CASE clauses are checked in sequence // because of the order of the clauses

MP3 a value < 30 satisfies the first clause // Clause '< 20' will never be tested

(3 marks)

(c) The following line is added immediately before the ENDCASE statement:

OTHERWISE : Level ← "Undefined"

State why this assignment is never performed.

Answer

Mark Scheme and Guidance

MP1 all of the possible values are addressed via all / four / three / the other clauses // there are no other possible values to map to OTHERWISE

(1 mark)

11 Sequence is one programming construct.

Identify **two other** programming constructs that will be required when the algorithm is converted into pseudocode.

Construct 1 .....

Construct 2 .....

Answer

Mark Scheme and Guidance

MP1 An iterative construct // a (count-controlled) loop **MP2** A selection construct // an IF statement

(2 marks)

12 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.

The solution to the algorithm includes iteration.

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

Justify your answer.

Name .....

Justification .....

Answer

Mark Scheme and Guidance

One mark per point:

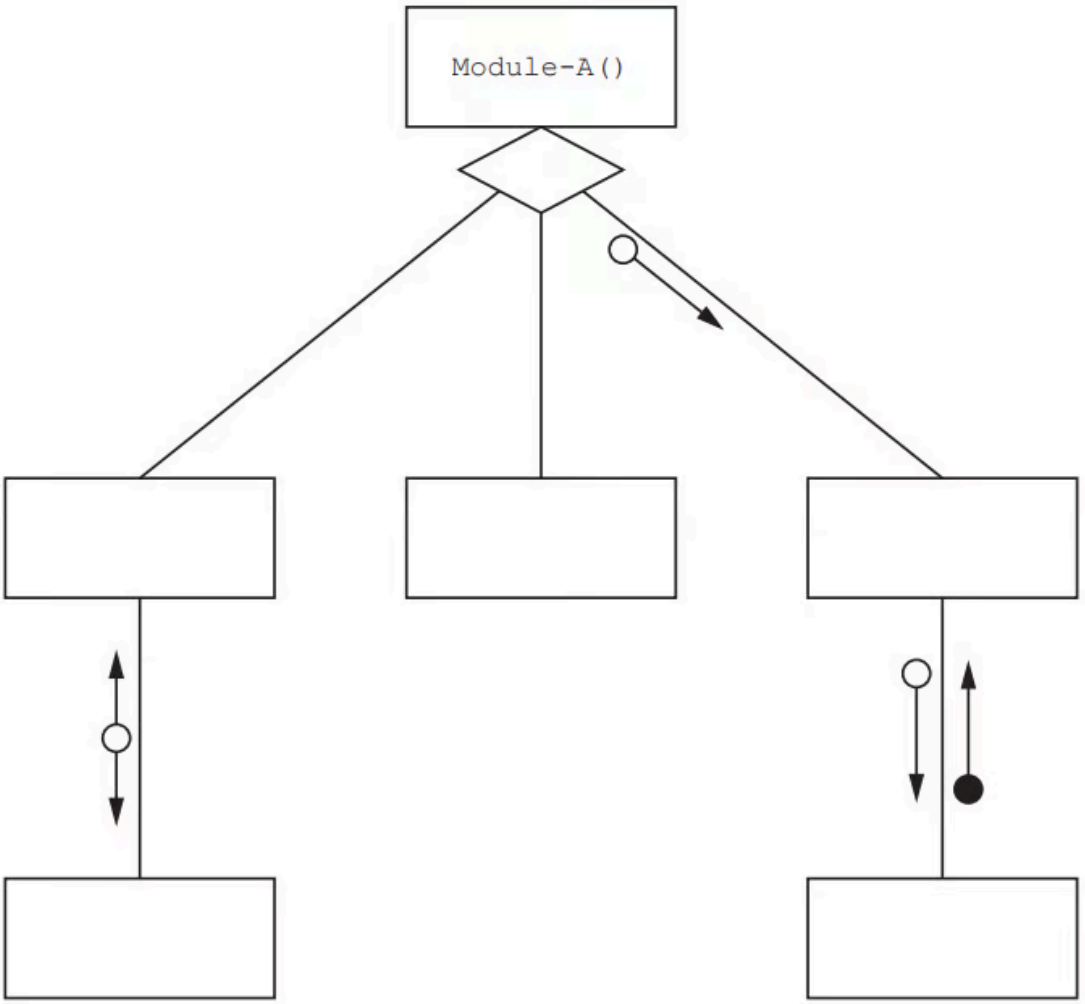
1. Name: (pre / post) conditional loop
2. Justification: the number of iterations is not known // loop ends following a specific input (in the loop)

(2 marks)

13 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()



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

Answer

Mark Scheme and Guidance

Means that Module-A calls either one of Module-X, Module-Y or ModuleZ (which one is called is decided at runtime).

One mark for reference to selection
One mark for naming all four modules correctly

(2 marks)

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

```
Mix[1] ← 4
Mix[2] ← 2
Mix[3] ← 3
Mix[4] ← 5
```

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

The procedure is written in pseudocode as follows:

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

Index ← Start
Total ← 0

WHILE Total < 20
Value ← Mix[Index]
Total ← Total + Value

IF Index < 4 THEN
Mix[Index] ← Mix[Index] + Mix[Index+1]
ELSE
Mix[Index] ← Mix[Index] + Mix[1]
ENDIF
Index ← (Value MOD 4) + 1

ENDWHILE

Mix[1] ← Total * Index

ENDPROCEDURE
```

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

```
CALL Process(2)
```

Index	Value	Total	Mix[1]	Mix[2]	Mix[3]	Mix[4]

Answer

Mark Scheme and Guidance

Index	Value	Total	Mix[1]	Mix[2]	Mix[3]	Mix[4]
2		0	4	2	3	5
	2	2				
				5		
3						
	3	5				
					8	
4						
	5	10				
						9
2						
	5	15				
				13		
2						
	13	28				
				21		
2						
			56			

MP1 Row 1 (initialisation)

Each iteration (1 – 5):
MP2 1 – Total 2
MP3 2 – Total 5
MP4 3 – Total 10
MP5 4 – Total 15
MP6 5 – Total 28 **and** final Mix[1] = 56

(6 marks)