7 Algorithm design and problem-solving – Error Checking
QUESTIONS

2 This section of program code asks for 50 numbers to be entered. The total and average of the numbers are calculated.

```
1   Total = 0
2   Counter = 50
3   PRINT 'When prompted, enter 50 numbers, one at a time'
4   REPEAT
5     PRINT 'Enter a number'
6     INPUT Number
7     Total + Number = Total
8     Number = Number + 1
9   UNTIL Counter = 50
10  Average = Number * Counter
11  PRINT 'The average of the numbers you entered is ', Average
```

There are **four** errors in this code.

State the line number for each error and write the correct code for that line.

Error 1 Line number ..... Counter = 0 Count <-- 0

Correct code .....
7

Error 2 Line number .....

Correct code ..... Counter <-- Counter +1

Error 3 Line number 8

Correct code ..... Total <-- Total + Number

Error 4 Line number 10

Correct code ..... Average <-- Total / Counter Total/50

[4]

7 Algorithm design and problem-solving – Error Checking
QUESTIONS

2 Read this section of program code that should input 50 numbers and then output the average of the positive numbers only.

```
1   Total = 0
2   PosCount = 0
3   FOR Counter = 1 TO 50
4     INPUT Num
5     IF Num < 0 THEN Total = Total + Num
6     IF Num > 0 THEN Counter = Counter + 1
7     Average = Total/PosCount
8   NEXT Counter
9   PRINT Num
```

There are **four** errors in this code.

Locate these errors and suggest code corrections to remove each error.

1 .....

.....

2 .....

.....

3 .....

.....

4 .....

.....[4]

7 Algorithm design and problem-solving – Error Checking
QUESTIONS

2 Read this section of program code that:

- inputs 10 numbers
- checks whether each number is within a specified range
- totals the numbers within the range and outside the range

```
1   InRange = 0
2   OutRange = 1000
3   FOR Count = 1 TO 10
4     INPUT Num
5     IF Num > 10 AND Num < 20 THEN InRange = InRange + 1
6       ELSE OutRange = OutRange - 1
7   Count = Count + 1
8   NEXT X
9   PRINT InRange, OutRange
```

(a) There are four errors in this code.

Locate these errors and suggest a correction to remove each error.

Error 1 .....

Correction .....

.....

Error 2 .....

Correction .....

.....

Error 3 .....

Correction .....

.....

Error 4 .....

Correction .....

.....[4]

(b) Decide, with reasons, whether the numbers 10 and 20 are within or outside the range.

Number	Within range (✓)	Outside range (✓)	Reason
10			....
20			....

[4]

7 Algorithm design and problem-solving – Error Checking
QUESTIONS

4 An algorithm has been written in pseudocode to allow the names of 50 cities and their countries to be entered and stored in a two-dimensional (2D) array. The contents of the array are then output.

```
01 DECLARE City ARRAY[1:50, 1:2] OF BOOLEAN
02 DECLARE Count : INTEGER
03 DECLARE Out : INTEGER
04 Count ← 1
05 IF
06     OUTPUT "Enter the name of the city"
07     INPUT City[Count, 2]
08     OUTPUT "Enter the name of the country"
09     INPUT City[Count, 2]
10     Count ← Count + 1
11 UNTIL Count = 50
12 FOR Out ← 1 TO 1
13     OUTPUT "The city ", City[Out, 1], " is in ", City[Out, 2]
14 NEXT Out
```

(a) Identify the **four** errors in the pseudocode and suggest corrections.

Error 1 .....

Correction .....

.....

Error 2 .....

Correction .....

.....

Error 3 .....

Correction .....

.....

Error 4 .....

Correction .....

.....

[4]

7 Algorithm design and problem-solving – Error Checking
QUESTIONS

5 An algorithm has been written in pseudocode to check if a new password is in a list of previously used passwords `OldList[]`
If the password is **not** found, the new password will be stored at the end of the list to replace `"XXXX"` already stored there.

```
01 OUTPUT "Enter your new password "
02 INPUT NewPassword
03 Posn ← 1
04 Found ← FALSE
05 REPEAT
06     IF Password = OldList[Posn]
07       THEN
08         Found ← TRUE
09       ELSE Posn ← Posn + 1
10     ENDIF
11 UNTIL Found AND OldList[Posn] = "XXXX"
12 IF Found
13   THEN
14     OUTPUT "Password has been used before"
15   ELSE
16     INPUT "New password accepted"
17     OldList[Posn] ← NewPassword
18 ENDIF
```

(a) Identify the **three** errors in the pseudocode and suggest corrections.

Error 1 .....

Correction .....

.....

Error 2 .....

Correction .....

.....

Error 3 .....

Correction .....

.....

[3]

(b) Complete this flowchart for the corrected algorithm:

[6]

7 Algorithm design and problem-solving – Error Checking
QUESTIONS

5 An algorithm has been written in pseudocode to allow some numbers to be input. All the positive numbers that are input are totalled and this total is output at the end.
An input of `0` stops the algorithm.

```
01 Exit ← 1
02 WHILE Exit <> 0 DO
03     INPUT Number
04     IF Number < 0
05       THEN
06         Total ← Total + Number
07       ELSE
08         IF Number = 0
09           THEN
10             Exit ← 1
11         ENDIF
12     ENDIF
13 ENDIF
14 OUTPUT "The total value of your numbers is ", Number
```

(a) Identify the **four** errors in the pseudocode and suggest a correction for each error.

Error 1 .....

Correction .....

.....

Error 2 .....

Correction .....

.....

Error 3 .....

Correction .....

.....

Error 4 .....

Correction .....

.....

[4]

(b) Describe how you could change the corrected algorithm to record and output how many positive numbers have been included in the final total.

You do **not** need to rewrite the algorithm.

.....

.....

.....

.....

.....

.....

.....

..... [4]

7 Algorithm design and problem-solving – Error Checking
QUESTIONS

6 An algorithm has been written in pseudocode to allow 100 positive numbers to be input. The total and the average of the numbers are output.

```
01 Counter ← 100
02 Total ← 0
03 WHILE Counter > 100 DO
04    INPUT Number
05    IF Number > 0
06       THEN
07         Total ← Total + Counter
08         Counter ← Counter + 1
09    ENDCASE
10 ENDWHILE
11 OUTPUT "The total value of your numbers is ", Total
12 OUTPUT "The average value of your numbers is ", Total / 100
```

(a) Identify the **four** errors in the pseudocode and suggest corrections.

Error 1 .....

Correction .....

.....

Error 2 .....

Correction .....

.....

Error 3 .....

Correction .....

.....

Error 4 .....

Correction .....

.....

[4]

(b) Describe the changes you should make to the corrected algorithm so that a count-controlled loop is used to allow 100 positive numbers to be input.

You do **not** need to rewrite the algorithm.

.....

.....

.....

.....

.....

.....

.....

.....

.....

..... [5]

7 Algorithm design and problem-solving – Error Checking
QUESTIONS

7 An algorithm has been written in pseudocode to calculate a check digit for a four-digit number. The algorithm then outputs the five-digit number including the check digit.
The algorithm stops when –1 is input as the fourth digit.

```
01 Flag ← FALSE
02 REPEAT
03     Total ← 0
04     FOR Counter ← 1 TO 4
05       OUTPUT "Enter a digit ", Counter
06       INPUT Number[Counter]
07       Total ← Total + Number * Counter
08       IF Number[Counter] = 0
09         THEN
10           Flag ← TRUE
11       ENDIF
12     NEXT Counter
13     IF NOT Flag
14       THEN
15         Number[5] ← MOD(Total, 10)
16         FOR Counter ← 0 TO 5
17             OUTPUT Number[Counter]
18         NEXT
19     ENDIF
20 UNTIL Flag
```

(a) Give the line number(s) for the statements showing:

Totalling .....

Count-controlled loop .....

Post-condition loop .....

[3]

(b) Identify the **three** errors in the pseudocode and suggest a correction for each error.

Error 1 .....

Correction .....

.....

Error 2 .....

Correction .....

.....

Error 3 .....

Correction .....

.....

[3]

7 Algorithm design and problem-solving – Error Checking
QUESTIONS

6 The energy efficiency of an electrical appliance is the percentage of useful energy out compared with the total energy in.

An algorithm has been written in pseudocode to calculate the energy efficiency of an appliance. Values for total energy in and useful energy out are input. The efficiency is calculated and output as a percentage.
The entry of the number –1 for either value stops the algorithm.

```
01 REPEAT
02     OUTPUT "Enter total energy in "
03     INPUT TotalEnergyIn
04     OUTPUT "Enter useful energy out "
05     OUTPUT UsefulEnergyOut
06     IF TotalEnergyIn <> -1 AND UsefulEnergy <> -1
07       THEN
08         Efficiency ← (UsefulEnergyOut / TotalEnergyIn) * 100
09         OUTPUT "Efficiency is ", Efficiency, "%"
10     ENDIF
11 UNTIL TotalEnergyIn <> -1 OR UsefulEnergyOut <> -1
```

(a) Identify the **three** errors in the pseudocode and suggest corrections.

Error 1 .....

Correction .....

.....

Error 2 .....

Correction .....

.....

Error 3 .....

Correction .....

.....

[3]

(b) Write pseudocode to check for an efficiency of 92% or over for this appliance and to output "A-rated" if the efficiency is 92% or over.

.....

.....

.....

.....

..... [2]