Section A

A farmer records the milk production of a herd of cows. Every cow has a unique 3-digit identity code. Each cow can be milked twice a day, seven days a week. The volume of milk from each cow is recorded in litres correct to one decimal place (yield) every time the cow is milked. The size of the herd is fixed. At the end of the week the total and the average yield for each cow for that week is calculated.
The farmer identifies the cow that has produced the most milk that week. The farmer also identifies any cows that have produced less than 12 litres of milk on four or more days that week.
A program is required to record the yield for each cow every time it is milked. calculate the total weekly volume of milk for the herd and the average yield per cow in a week. The program must also identify the cow with the best yield that week and identify any cows with a yield of less than 12 litres of milk for four or more days that week.

Write and test a program or programs for the farmer.

- Your program or programs must include appropriate prompts for the entry of data.
- Error messages and other output need to be set out clearly and understandably.
- All variables, constants and other identifiers must have meaningful names.

You will need to complete these **three** tasks. Each task must be fully tested.
TASK 1 - Record the yield.
Write a program for TASK 1 to record the milk yields for a week. The program records and stores the identity code number and the yield every time a cow is milked.

TASK 2 - Calculate the statistics.
Using your recorded data from TASK 1, calculate and display the total weekly volume of milk for the herd to the nearest whole litre. Calculate and display the average yield per cow in a week to the nearest whole litre.

TASK 3 - Identify the most productive cow and cows that are producing a low volume of milk.
Extend TASK 2 to identify and display the identity code number and weekly yield of the cow that has produced the most milk. Also identify and display the identity code numbers of any cows with a yield of less than 12 litres of milk for four days or more in the week.

1 (a) All variables, constants and other identifiers should have meaningful names.

(i) In Task 2 & 3, you had to store the Highest Yield and Total Yield in variables.
Write suitable declarations for two variables in pseudocode or program code and initialize them too.

.....
.....
.....
..... **[4]**

(ii) It has been decided to record the Identity Code and Cow total yield input by the user in an array of size 5.
Write the new array `IdentityCode` and `CowTotalYield` declaration that you would use.

.....
.....
.....
..... **[2]**

(iii) Declare a constant that you have used in Task 1 and state what you used it for.
Constant .....

Use ..... **[2]**

(b) Task 1 uses an array `IdentityCode` you have already declared in 1(a)(ii). The array can accept any combination of identity code. However it is required to have a 3-digit Identity code. This situation requires a validation check to be applied over the input of identity code.

Write a validation check using pseudocode, programming code or flowchart that you would have used in solving Task 1 using an array `IdentityCode`; possibly using a condition based loop.

.....
.....
.....
.....
.....
.....
.....
.....
.....
..... **[4]**

(c) Complete the algorithm below to complete Task 2, using pseudocode.

```
FOR cntcows ← 1 TO herdsize
      cowavgyield ← ..........
      temp ← cowavgyield - INT(cowavgyield)
      IF temp = ..... THEN
            cowavgyield ← INT(cowavgyield)
      ELSE
            ..........
      END IF
      PRINT "Average yield of cow ID: ", IdentityCode[cntcows], "is", cowavgyield,"ltr"
      weeklyvol ← ..........
NEXT cntcows
```

[4]

(d) Explain how you completed Task 3. You should assume that Task 2 has been completed using an array `IdentityCode`.

You can include pseudocode or programming statements as part of your explanation.

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....[4]

Section B

2 Read this section of program code that should input 50 numbers and then output the average of the digits per numbers input, maximum number of digits entered as number and number of digits entered per number.

```
A. FOR Count ← 1 to 500
B.   INPUT Num
C.   Digits ← INT(LOG(Num))+1
D.   CASE OF Digits
E.     1: OUTPUT "One Digit"
F.     2: OUTPUT "Twenty Digits"
G.     3: OUTPUT "Three Digits"
H.     4: OUTPUT "Four Digits"
I.   END CASE
J.   OTHERWISE: OUTPUT "Too Many Digits."
K.   Total ← Total + Digits
L.   IF Digits > Highest Then Highest = Highest +1
M. END FOR
N. AVG ← Total/50
O. OUTPUT "Average number of digits entered per turn was ", AVG
P. OUTPUT "Biggest number entered had ", Highest, " Digits."
```

There are **four** errors in this code.
Locate these errors and suggest code corrections to remove each error.

1 .....
.....

2 .....
.....

3 .....
.....

4 .....
..... **[4]**

3 ZAK Autos gives discounts based on the number of visitors in a group.
The pseudo-code for an algorithm that determines group discounts is shown.

```
A. IF ((numAdults > 1) AND (numChildren > 0)) THEN
B.   OUTPUT "Family discount"
C. ELSE
D.   IF (numAdults >= 10) THEN
E.     OUTPUT "Large group discount"
F.   ELSE
G.     IF (numAdults >= 5) THEN
H.       OUTPUT "small group discount"
I.     ELSE
J.       OUTPUT "Regular pricing"
K.     ENDIF
L.   ENDIF
M. ENDIF
```

(a) Complete the table to show the output of the pseudo-code algorithm, based on the given inputs.

INPUT		Output Displayed
numAdults	**numChildren**	
8	0	
2	2	
12	0	

[3]

(b) The pseudo-code algorithm needs to be tested more thoroughly.
Construct test data to meet the requirements set out in the table.

Requirements	INPUT	
	numAdults	**numChildren**
A condition generating 'regular pricing'		0
Smallest group qualifying for 'family discount'		

[3]

(c) Complete the table to give the appropriate data type of a variable to store each item.

Item	Data Type
Gender of individual staff member	
Whether an individual car is still under manufacturer's warranty	
Mean number of hours needed to recharge the battery in each car	
The number on the individual car	

[4]

(d) State a validation check that you can perform on each of these fields. Each validation check must be different.

Item	Validation Check
Gender of individual staff member	
Whether an individual car is still under manufacturer's warranty	
Mean number of hours needed to recharge the battery in each car	
The number on the individual car	

[4]

4 Complete the query-by-example grid below to select and show the `StudentName`, `Class` and `Percentage` of students Age above 14 in a table `StuTable`. Results must be in decreasing order of `Percentage`.

Field:					
Table:					
Sort:					
Show:	☐	☐	☐	☐	☐
Criteria:					
or:					

[5]

5 A routine checks the age and height of children who are allowed to enter a play area. The children must be less than 7 years of age and under 1.2 metre in height.

Provide two additional sets of test data. For each, give

- the type of each set of test data

Each type of test data and reason for use must be different.

Set 1

.....

Type

.....

Set 2

.....

Type

.....

[4]

6 What is an identifier and state two rules for proper identifier naming.

Identifier:

.....

.....

Naming Rule 1:

.....

Rule 2: .....

[3]