Algorithm_1

Total Marks: 18

Question 1:

Read this section of program code that should input 10 positive numbers and then output the smallest number input.

```
1   Small = 0
2   Counter = 0
3   REPEAT
4     INPUT Num
5     IF Num < Small THEN Num = Small
6     Counter = Counter + 1
7     PRINT Small
8   UNTIL Counter < 10
```

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

Locate these errors and suggest a corrected piece of code for each error.

1 .....

.....

2 .....

.....

3 .....

.....

4 .....

.....[4]

Answer:

1 mark for each error identified + suggested correction

Line 1 or `Small = 0`: this should read **`Small = 999`**

line 5 or `IF...`: this should read **`IF Num < Small THEN Small = Num`**

line 8 or `UNTIL`: this should read **`UNTIL Counter = 10`** or **`UNTIL Counter > = 10`** or **`UNTIL Counter > 9`**

line 7 or `PRINT...`: **`PRINT Small`** should come after the end of the repeat loop

or

line 8 or `UNTIL`: this should come before line 7 [4]

Question 2:

Read this section of program code that should input 30 positive numbers and then output the largest number input.

```
1   Large = 9999
2   Counter = 0
3   WHILE Counter > 30
4   DO
5     INPUT Num
6     IF Num < Large THEN Large = Num
7     Counter = Counter - 1
8   ENDWHILE
9   PRINT Large
```

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

Locate these errors and suggest a corrected piece of code for each error.

1 .....

.....

2 .....

.....

3 .....

.....

4 .....

.....[4]

Answer:

1 mark for each error identified + suggested correction

Line 1 or `Large =9999`: this should read **`Large = 0`**

Line 3 or `WHILE`: this should read **`WHILE Counter < 30`**

line 6 or `IF`: this should read **`IF Num > Large THEN Large = Num`**

line 7 or `Counter =...`: this should read **`Counter = Counter + 1`** [4]

Question 3:

Four programming concepts and four examples of programming code are shown below.

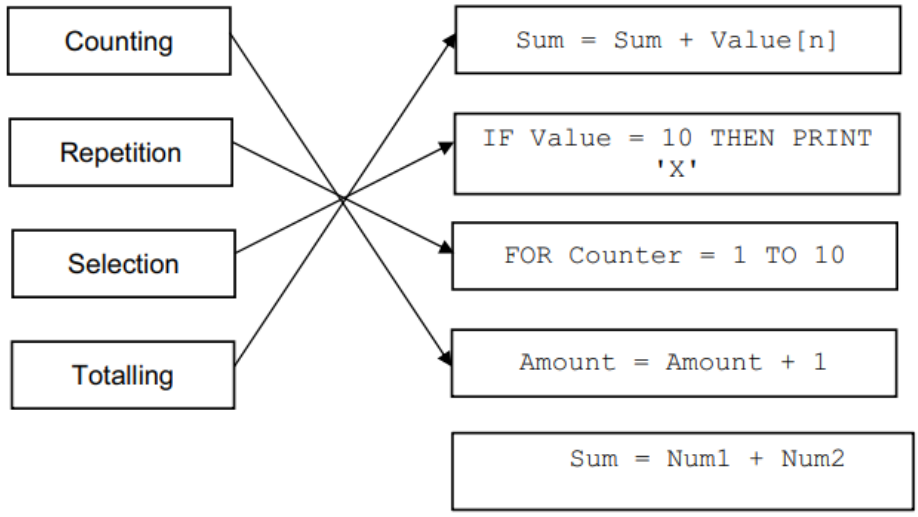
Draw a line to link each programming concept to the correct example of programming code.

Programming concept	Example of programming code
Counting	`Sum = Sum + Value[n]`
Repetition	`IF Value = 10 THEN PRINT 'X'`
Selection	`FOR Counter = 1 TO 10`
Totalling	`Amount = Amount + 1`
	`Sum = Num1 + Num2`

[4]

Answer:

1 mark for each correct line, two lines from one box not allowed



[4]

Question 4:

Write an algorithm, using pseudocode and a `FOR ... TO ... NEXT` loop structure, to input 1000 numbers into an array.

.....

.....

.....

.....

.....

.....[2]

Answer:

1 mark for `FOR ... TO ... NEXT` **1** mark for `INPUT`

```
FOR Count ← 1 TO 1000
  INPUT A[Count]
NEXT (Count)
```

[2]

Question 4:

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

```
1   Total = 0
2   For Counter = 1 TO 50
3     INPUT Num
4     Total = Total + 1
5     Counter = Counter + 1
6     Average = Total/Counter
7   NEXT Counter
8   PRINT Average
```

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

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

1 .....

.....

2 .....

.....

3 .....

.....

4 .....

.....[4]

Answer:

One mark for each error identified + suggested correction
line 4 or (`Total =) Total + 1`: this should read (`Total =) Total + Num`
line 5 or `Counter = Counter + 1`: delete this line
line 6 or (`Average = )Total / Counter`: swap lines 6 and 7
line 6 or (`Average = )Total / Counter` : this should read (`Average =) Total / 50`

[4]