

4 This pseudocode algorithm is intended to allow data for up to 50 people to be entered and stored in a two-dimensional (2D) array. The data is their last name, first name and the city in which they live.

```
01 DECLARE People : ARRAY[1:50, 1:3] OF REAL
02 DECLARE Count : INTEGER
03 DECLARE Response : CHAR
04 DECLARE Continue : BOOLEAN
05 FOR I ← 1 TO 50
06     FOR J ← 1 TO 3
07         People[I, J] ← ""
08     NEXT J
09 NEXT I
10 Count ← 100
11 Continue ← TRUE
12 CASE OF
13     OUTPUT "Enter the last name"
14     INPUT People[Count, 1]
15     OUTPUT "Enter the first name"
16     INPUT People[Count, 2]
17     OUTPUT "Enter the city"
18     INPUT People[Count, 3]
19     OUTPUT "Do you want to enter another name (Y or N)?"
20     INPUT Response
21     IF Response = 'N'
22         THEN
23             Continue ← FALSE
24         ELSE
25             Count ← Count + 1
26     ENDIF
27 UNTIL NOT Continue
```

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

Error 1 line number .....

Correction .....

.....

Error 2 line number .....

Correction .....

.....

Error 3 line number .....

Correction .....

.....

Error 4 line number .....

Correction .....

.....

[4]

(b) Write the pseudocode that you could add to the end of this algorithm to output the contents of the array. Make sure that the output ends when the data in the array ends.

.....

.....

.....

.....

.....

.....

.....

..... [4]

(c) Explain how you could alter the original corrected algorithm to make sure that the number of elements being added to the array does not exceed the maximum size of the array (50 elements).

.....

.....

.....

.....

.....

.....

.....

..... [4]