Keywords

Dry run – walking through an algorithm which sample data, running each step manually.

Trace Table – a table that follows the values of variables to check for accuracy.

Algorithms

Correcting and Completing Algorithms

Correcting Algorithms

CARR MANOR COMMUNITY SCHOOL

Objectives

ADVANCED:
Be able to find and correct errors in algorithms.

EXPERT:
Be able to complete algorithms where code is missing.

Step 1: Read what the algorithm needs to do. Read it at least twice.

Step 2: Work out what it should do in note form, e.g. draft the steps that should take place.

Step 3: Trace, or read through each step, of the algorithm.

Step 4: Compare what the algorithm does to your notes about what is needs to do.

Correcting Algorithms

CARR MANOR COMMUNITY SCHOOL

An algorithm should take two numbers as input, and then output the smallest number.

Step 1: Read the above point twice.

Step 2: Make your own notes for how it should work.

Step 3: Now trace the algorithm, use examples numbers e.g. 5 and 3.

Step 4: Compare Step 3 with Step 2.

There are **four** errors.

```
num1 = input("Enter the first number")
num1 = input("Enter the second number")

if num1 < num2 then
          print(num2)
elseif num2 < num1 then
          print(num2)
elseif
          print("Same")
stop
```

```
num1 = input("Enter the first number")
num2 = input("Enter the second number")

if num1 < num2 then
          print(num1)
elseif num2 < num1 then
          print(num2)
else
          print("Same")
endif
```

Completing Algorithms

CARR MANOR COMMUNITY SCHOOL

Objectives

ADVANCED:
Be able to find and correct errors in algorithms.

EXPERT:
Be able to complete algorithms where code is missing.

You may be given an incomplete algorithm.
You will need to complete it.
It may be some elements are missing.
Or only part of the algorithm is present.

Follow the same stages as correcting, but this time, add your own code to finish it.

Completing Algorithms

CARR MANOR COMMUNITY SCHOOL

An algorithm should ask the user to input 10 numbers, which are stored in the array.
The program then outputs the total of these numbers (added together).
The program then outputs the average number.

```
count = ......
while count <= .....
    num[count] = input("Enter number")
    ......................
endwhile
total = 0
for x = 0 to 9
          total = total + ..........
next x
average = total/10
.........................
.........................
```

```
count = 0
while count <= 9
    num[count] = input("Enter number")
    count = count + 1
endwhile
total = 0
for x = 0 to 9
          total = total + num[x]
next x
average = total/10
print ("The total is " & total)
print ("The average is " & average)
```