Bug Hunters — Find and Fix the Broken Algorithms

Name: ________________________________ **Date:** ____________

Bug Type Reference

Bug Type	Description	Clue
Wrong Order	All steps present but in the wrong sequence	The result almost works, but something happens too early or too late
Missing Step	An essential step has been left out	The algorithm seems too short; something that should happen never does
Logic Error	A condition or value is incorrect	The algorithm runs without crashing but produces the wrong result

Algorithm 1: Making Toast

```
Step 1: Put bread in the toaster
Step 2: Eat the toast
Step 3: Wait for the toaster to pop
Step 4: Spread butter on the toast
Step 5: Take the toast out of the toaster
```

Trace — what actually happens if you follow this exactly?

Bug type: Wrong Order / Missing Step / Logic Error *(circle one)*

Which step(s) are in the wrong position? Step(s) ___

Write the FIXED algorithm:

1. ____________________
2. ____________________
3. ____________________
4. ____________________
5. ____________________

Algorithm 2: Opening a Combination Lock

A combination lock has three numbers. You must turn the dial right to the first number, then left to the second, then right to the third, then pull.

```
Step 1: Turn dial RIGHT to first number (e.g., 15)
Step 2: Turn dial LEFT to second number (e.g., 30)
Step 3: Pull the lock open
```

Trace — what actually happens?

Bug type: Wrong Order / Missing Step / Logic Error *(circle one)*

What is missing?

Write the FIXED algorithm:

1.
2.
3.
4.

Algorithm 3: Finding the Largest Number

```
numbers = [3, 7, 2, 9, 1]
biggest = 0

FOR EACH number IN numbers:
    IF number < biggest:
        biggest = number

PRINT biggest
```

Trace — run this with the list [3, 7, 2, 9, 1]. Fill in the table:

Step	number	biggest (before)	Condition: number < biggest?	biggest (after)
Start	—	0	—	0
1st item	3	0	3 < 0?	
2nd item	7		7 < ?	

Step	number	biggest (before)	Condition: number < biggest?	biggest (after)
3rd item	2			
4th item	9			
5th item	1			

What does the algorithm PRINT? ___

What SHOULD it print? ___

Bug type: Wrong Order / Missing Step / Logic Error *(circle one)*

What exactly is wrong? ____

Write the fix (change only the minimum needed):

```
numbers = [3, 7, 2, 9, 1]
biggest = 0

FOR EACH number IN numbers:
    IF number ___ biggest:
        biggest = number

PRINT biggest
```

Describe a test case that would catch this bug:

Describe a test case that would NOT catch this bug (the buggy algorithm would give the right answer):