Function Machine — Design Your Functions

Name: _____ **Date:** _____

Part 1: The Mystery Machine

Record the teacher's inputs and outputs:

Input	Output

My guess for the rule: _____

Actual function: `double_and_add_3(x) = 2 × x + 3`

Verify: `double_and_add_3(7)` = 2 × 7 + 3 = ___

Part 2: Your Group's Secret Function

Keep this hidden until after other groups have guessed!

Function name: _____

Parameter(s): _____

Rule: _____

```
DEFINE ____________________ ( ____________________ ):
    RETURN ____________________
```

Input	Expected Output

Part 3: Design Two More Functions

Function 2 (one parameter):

Name: _____ **Parameter:** _____

Rule: _____

```
DEFINE ____________________ ( ____________________ ):
    RETURN ____________________
```

Input	Output
3	
10	
0	

Function 3 (two parameters):

Name: _____ **Parameters:** _____, _____

Rule: _____

```
DEFINE ____________________ ( _________ , _________ ):
    RETURN ____________________
```

Input A	Input B	Output
4	6	
2	8	
5	5	

Part 4: Function Chaining

`double_and_add_3(double_and_add_3(2))`

Step 1 — inner function: `double_and_add_3(2)` = 2 × 2 + 3 = ___

Step 2 — outer function: `double_and_add_3(___)` = 2 × ___ + 3 = ___

Your own chain: _____ (_____ (___)) = ___

Part 5: Reflection

Difference between DEFINING and CALLING a function:

Why use descriptive names like `double_and_add_3` rather than just `f`?

How is a function like a recipe? Use this analogy to explain parameters and return values.
