

PRINTABLE WORKSHEETS

Underneath the Syntax

Student handouts to accompany the Instructor Guide

By Edison Mooers

Classroom reproduction permitted: educators may copy and distribute these worksheets to learners in classes, tutoring programs, library programs, workforce programs, paid bootcamps, employer-run training, and other educational settings they directly lead. The worksheets may not be sold separately, reposted as a complete public collection, incorporated into materials offered for resale, or redistributed as editable source files.

What's included

1. Program-Tracing Worksheet — *Chapters 2, 5, 9*
2. Boxes-and-Arrows Reference Diagram — *Chapters 13, 17*
3. Conditionals Worksheet — *Chapter 6*
4. Loop-Tracing Table — *Chapter 7*
5. Recursion Call-Stack Template — *Chapter 15*
6. Collection-Selection Worksheet — *Chapter 8*
7. Capstone Planning Sheet — *Chapter 25 / capstone*
8. AI-Generated-Code Review Checklist — *AI-Code-Literacy unit*

1. Program-Tracing Worksheet

Chapters 2, 5, 9 — What a Program Actually Is · Operators & Expressions · Strings

Name: _____ Date: _____ Refers to: Chapters 2, 5, 9

Your instructor will give you a short piece of pseudocode. Go through it one line at a time. For each line, write down what changes and what, if anything, gets displayed.

Line #	Instruction	Variables After This Line	Output (if any)

2. Boxes-and-Arrows Reference Diagram

Chapters 13, 17 — Value Types, Reference Types & Typing Systems · Pointers & References

Name: _____ Date: _____ Refers to: Chapters 13, 17

Draw a labeled box for each variable your instructor gives you. For a value type, draw a separate box holding its own copy of the value. For a reference type, draw the variable's box with an arrow pointing to a separate box holding the actual data, and draw a second arrow from any other variable that refers to the same data.

3. Conditionals Worksheet

Chapter 6 — Conditional Logic

Name: _____ Date: _____ Refers to: Chapter 6

Turn the policy or scenario your instructor gives you into a chain of conditions. Write one branch per row, in the order they should be checked.

Condition to Check	If True, Then...	If False, Then...

4. Loop-Tracing Table

Chapter 7 — Loops & Iteration

Name: _____ Date: _____ Refers to: Chapter 7

Trace the loop your instructor gives you one pass at a time. Add a row for every iteration, and stop only when the loop's condition is actually false.

Iteration #	Loop Variable Value	Condition Still True?	Output / Action

5. Recursion Call-Stack Template

Chapter 15 — Recursion

Name: _____ Date: _____ Refers to: Chapter 15

Work down from the first call to the base case, writing one row per call. Once you reach the base case, work back up, filling in each call's return value in the order the calls actually finish.

Call (e.g., factorial(4))	Argument	Waiting On	Return Value

6. Collection-Selection Worksheet

Chapter 8 — Arrays & Collections

Name: _____ Date: _____ Refers to: Chapter 8

For each scenario, choose Array, Map/Dictionary, or Set, and explain your choice in one sentence. The first two rows are examples; fill in the rest yourself or with scenarios from your instructor.

Scenario	Collection Chosen	Why
A class roster in seating order	Array	Order matters and every seat holds one name
A phone book (name → number)	Map / Dictionary	You look things up by name, not by position

7. Capstone Planning Sheet

Chapter 25 and the Capstone Project

Name: _____ Date: _____ Refers to: Chapter 25 / capstone

Use this sheet to plan your capstone project before you start writing pseudocode or code. Fill in each section completely; a little extra time here saves time later.

The problem I'm solving

Data I need to store (name and type for each)

Functions I'll need (name and what each one does)

Loops or conditionals I'll need, and why

Errors that could happen, and how I'll handle each one

How I'll test that it actually works

8. AI-Generated-Code Review Checklist

AI-Code-Literacy unit (see Instructor Guide)

Name: _____ Date: _____ Refers to: AI-Code-Literacy unit

Use this checklist while reviewing a piece of AI-generated code, whether you generated it yourself or your instructor provided it. Check off each item only once you can actually point to it in the code.

- ☐ I can identify every variable in the code and what it stores.
- ☐ I can identify every conditional and what decision it makes.
- ☐ I can identify every loop and what it repeats.
- ☐ I can identify every function's parameters and its return value.
- ☐ I identified at least one assumption the code makes about its input.
- ☐ I identified what error handling is missing, if any.
- ☐ I can explain this code in my own words, without relying on the AI's own explanation of it.
- ☐ I tested, or would test, at least one edge case not covered by the original prompt.

Notes
