LIFE SKILL: CODE REUSE

CSC111: Introduction to CS through Programming

Adapted from R. Jordan Crouser

Assistant Professor of Computer Science

Smith College

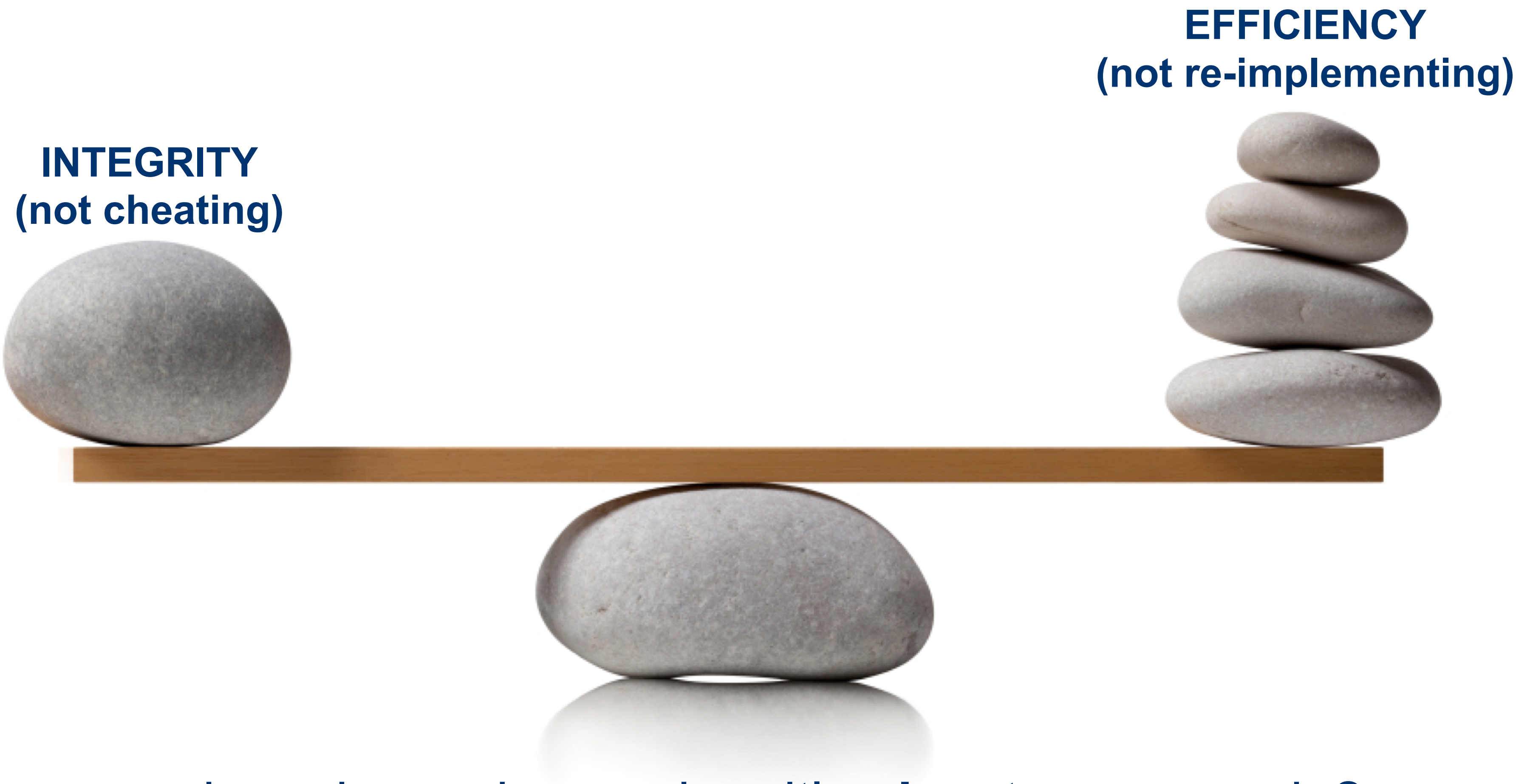
Discussion

We've talked about the programming process:

- "S^4": start small | slow | simple
- Organizing your code so it's easy to reuse pieces
- Documenting your code so it's easy to come back to it

Can you think of any **ethical concerns** about this?

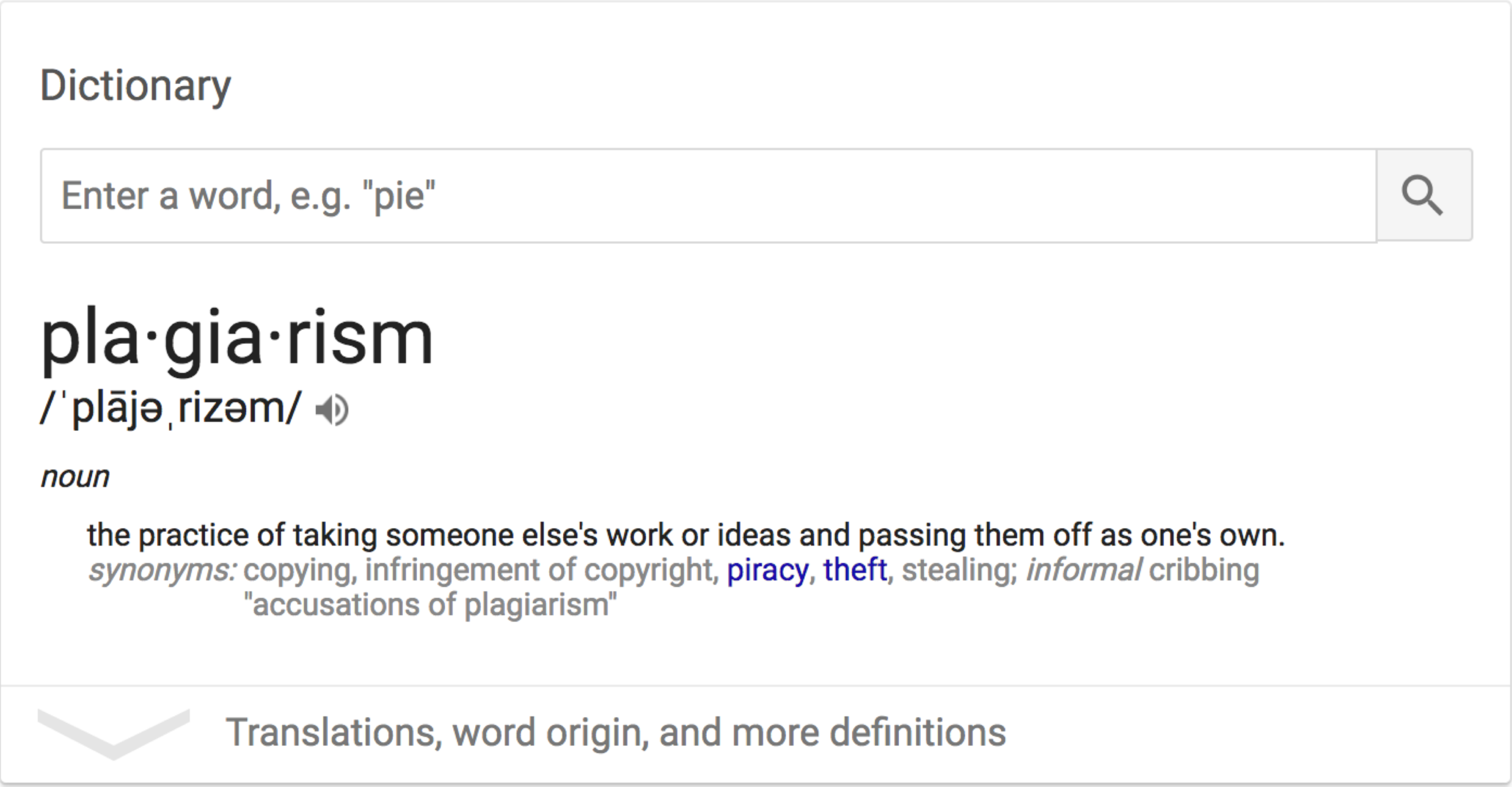
The balancing act…



…so how do you know when it's **okay** to reuse code?

Let’s consider a more familiar case…

Same or different?



Dictionary

pla·gia·rism

/ˈplājəˌrizəm/

noun

the practice of taking someone else's work or ideas and passing them off as one's own.

synonyms: copying, infringement of copyright, piracy, theft, stealing; *informal* cribbing

"accusations of plagiarism"

Scenario 0: self-reuse, not in a class

- You wrote a program that **solved a particular problem** as part of a project you're working on for fun
- Later on for **a different project** you're working on for fun, you need to solve the same problem
- Questions:
  - Can you reuse the code?
  - Does it matter what the code **does**?
  - Do you need to **attribute** the code?

Scenario 1: self-reuse, in-class work

- You wrote a program that **solved a particular problem** for a previous assignment in a course
- Later on for **a project you're working on for fun**, you need to solve the same problem
- Questions:
  - Can you reuse the code?
  - Does it matter what the code **does**?
  - Do you need to **attribute** the code?

Scenario 1: self-reuse, same course

- You wrote a program that **solved a particular problem** for a previous assignment in a course
- In a **later assignment for that same course**, you need to solve the same problem as part of a larger process
- Questions:
  - Can you reuse the code?
  - Does it matter what the code **does**?
  - Do you need to **attribute** the code?
  - Does it matter if you copy/paste or **`import`** it?

Scenario 2: self-reuse, different course

- You wrote a program that **solved a particular problem** for a previous assignment in a course
- In **an assignment for a different course**, you need to solve the same problem
- Questions:
  - Can you reuse the code?
  - Does it matter what the code **does**?
  - Do you need to **attribute** the code?
  - Does it matter if it’s the **whole assignment**, or just one part?

Scenario 3: self-reuse, academic work →

- You wrote a program that **solved a particular problem** for an assignment in a course
- You later get a **job as a software engineer**, and you need to solve the same problem
- Questions:
  - Can you reuse the code?
  - Does it matter what the code **does**?
  - Do you need to **attribute** the code?
  - Does it matter if it’s the **whole assignment**, or just one part?

Scenario 4: professors and TAs

- You are trying to **solve a particular problem** for an assignment in a course, but you are stuck
- You ask the professor or TA for advice, they walk you through **how to implement one of the functions**
- Questions:
  - Can you use the code?
  - Does it matter what the code **does**?
  - Do you need to **attribute** the code?

Scenario 5: peers

- You are trying to **solve a particular problem** for an assignment in a course, but you are stuck
- You ask a friend who took the class last year, they walk you through **how to implement one of the functions**
- Questions:
  - Can you use the code?
  - Does it matter what the code **does**?
  - Do you need to **attribute** the code?

Scenario 6: online sources

- You are trying to **solve a particular problem** for an assignment in a course, but you are stuck
- You look online to try to understand a concept, someone walks through **how to implement one of the functions**
- Questions:
  - Can you use the code?
  - Does it matter what the code **does**?
  - Do you need to **attribute** the code?

Common online Q&A resources

Google
Quora
stackoverflow
StackExchange
reddit

How to attribute online code

```
# ------- START ATTRIBUTED CODE SECTION -------
# Code created with the help of Stack Overflow
# https://stackoverflow.com/questions/49581417
#
# Question by Alden:
# https://stackoverflow.com/users/9378177/alden
#
# Answer by CD Lane:
# https://stackoverflow.com/users/5771269/cdlane

# ...THE ACTUAL CODE GOES HERE...

# --------- END ATTRIBUTED CODE SECTION --------
```

Discussion

For educational purposes, does copyright matter?

Isn’t everything covered by “**fair use**”?

Rules for code reuse

- Always **attribute**
- Only use code you **actually understand**
- If it’s for a course, **talk to the professor first**
- Understand **the license,** e.g. for StackOverflow

Attribution-ShareAlike 3.0 Unported (CC BY-SA 3.0)

Information you should review for Test 1

-
- Data types (`int`, `float`, `string`, `bool`, `list`, `dict`) and the operations they support
- Familiar functions: `print`, `range`, `eval`, `input`
- Exceptions and how to debug them
- Conditional statements
- Loops (both `for` and `while`)
- *Defining / calling functions*

Stuff you should be able to do for Test 1

- Define terms
- Describe what a function or line of code does
- Read a snippet of code and find / correct **common errors**
- **Trace the execution** of a short program, including functions that take in parameters and return values
- Write a **function** from a given description