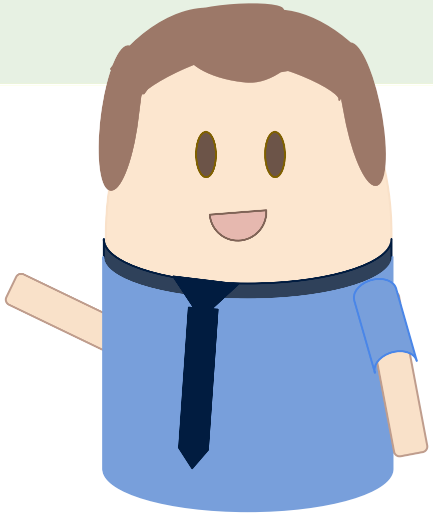
CMPUT 174

Introduction to Testing

Testing

What is it NOT?	What is it?
• A "proof of correctness" *Testing can <u>never completely identify all the defects</u> within software!*	• A ***<u>process</u>*** <u>of validating</u> and verifying that a program: ○ Meets the requirements ○ Works as expected



Testing: What is it?

Process of executing a program with the intent of finding errors

Glenford Myers [The Art of Software Testing, 1979]

This makes it a challenging task!

- It is not easy to find errors in programs (especially, large!)
- It is a destructive activity - your purpose is to find ***faults***
- Can be demoralizing and unrewarding (if not treated positively)

Testing Takes Creativity!

- Testing is often viewed as dirty work!
- To develop an *effective* test, one must have
  - Detailed understanding of the program
  - Knowledge of the testing techniques
  - Skill to apply these techniques
- Programmers often stick to the data set that makes the program work ("*happy path*")
- A program <u>often does not work when tried by somebody else</u>
  - Don't let this be the end-user (or your marking TA :-))!

Limits of Testing

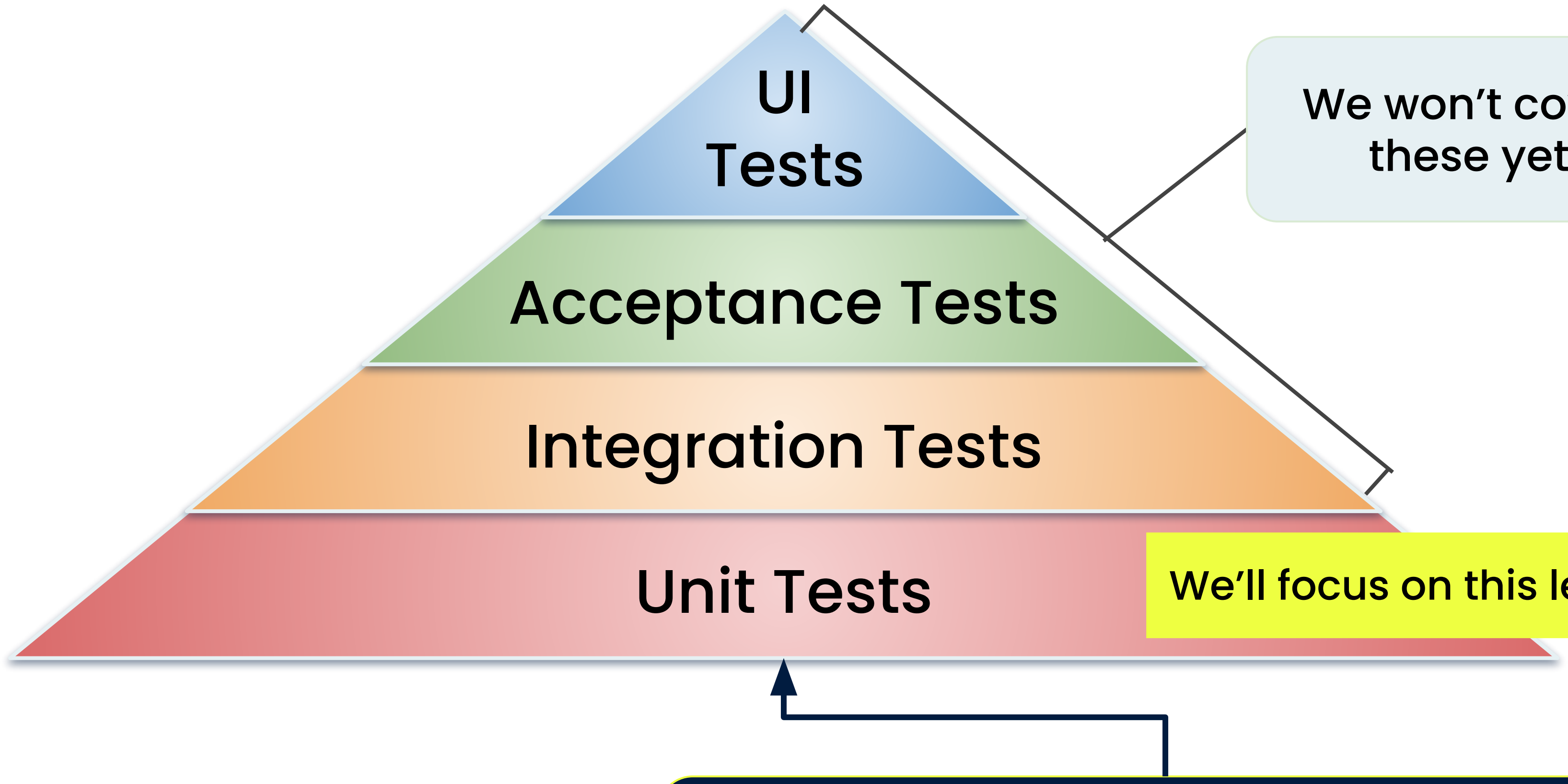
- A program can *not* be tested completely
  - Too many possible combinations to cover!
- Testing *cannot* find all defects
  - Cannot show their absence, just their presence

I can *only* partially test the program, and I can *only* find a defect's presence!

my_program.py

so many things to test!

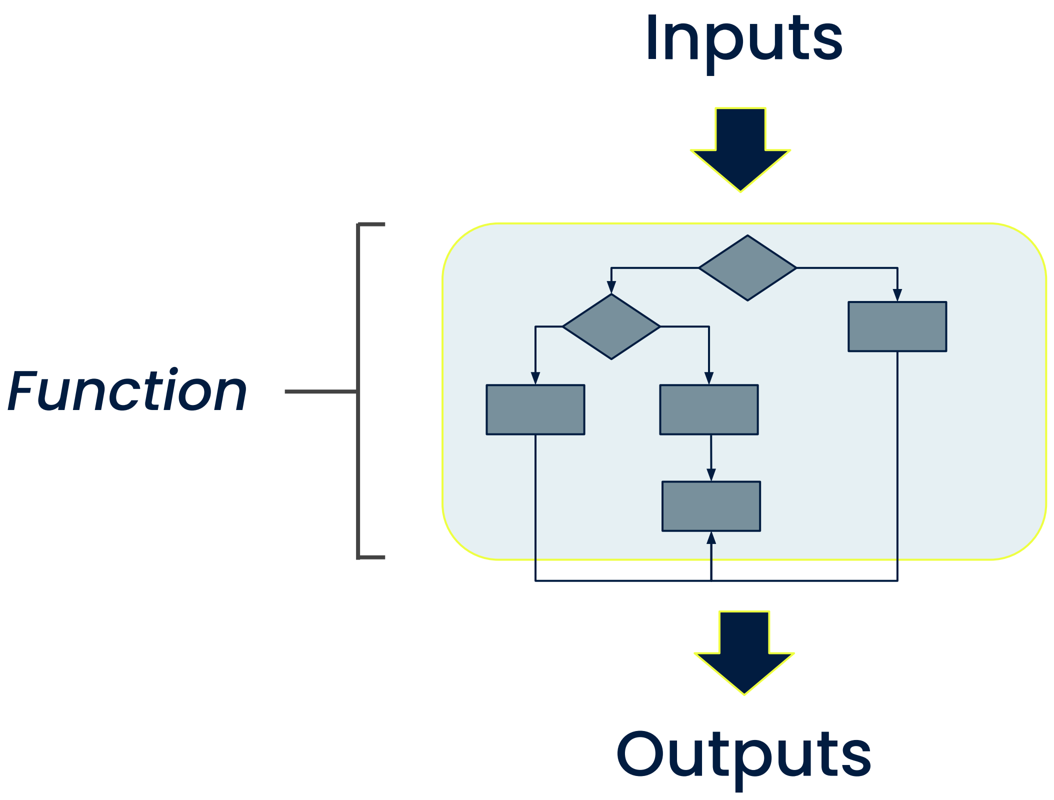
Testing Hierarchy



Unit tests are small tests that checks that a *single* component operates in the right way

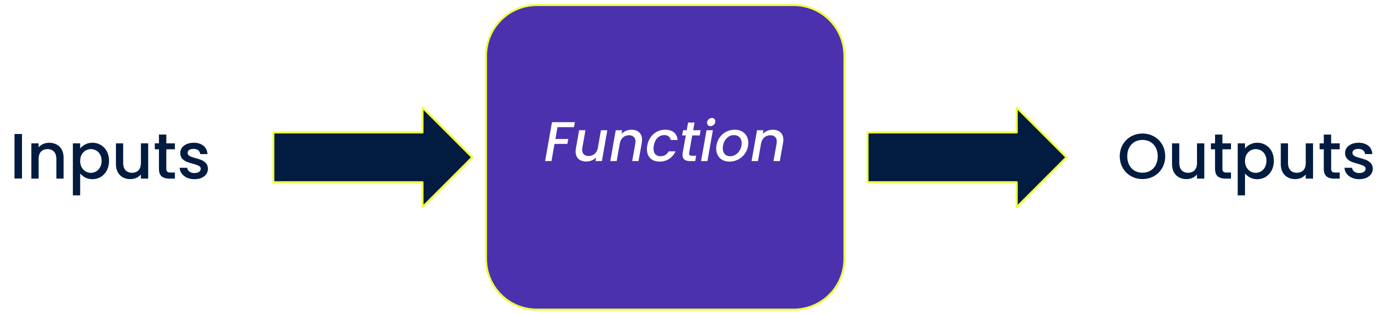
Types of Unit Testing

- White Box testing
  - Logical, structural, or program-based testing
  - Looks "under the cover"



Types of Unit Testing

- **Black Box testing**
  - Functional, specification-based testing
  - Inspects the function from the outside

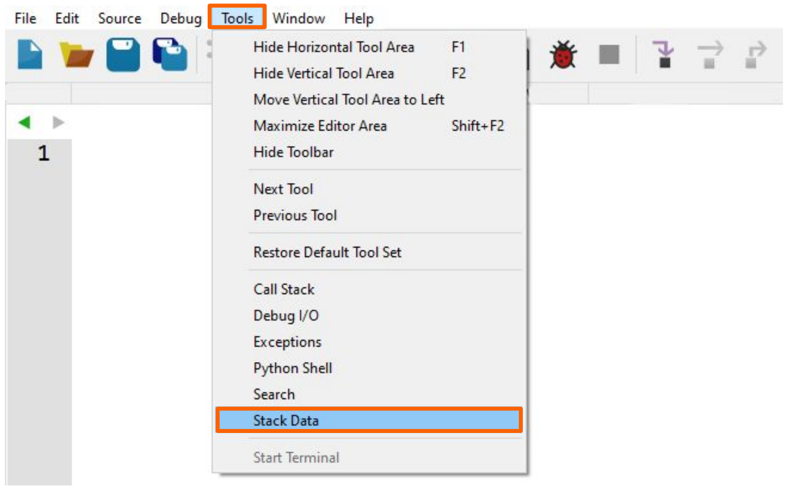


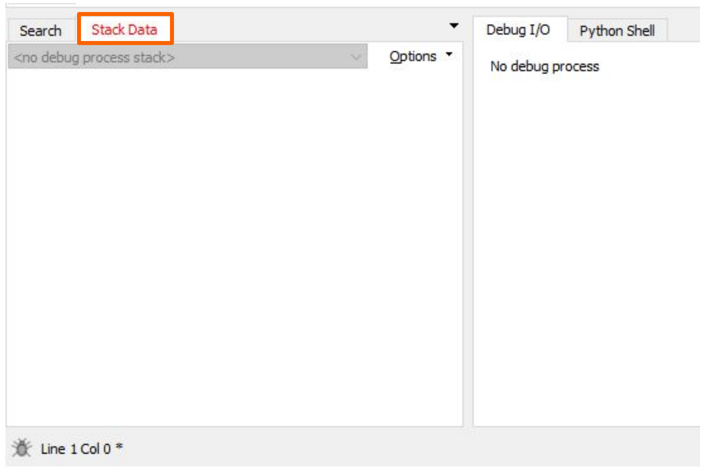
★ *We will focus on Black Box testing*

Debugging in Wing IDE

- **Debug** menu
  - Start / Continue (F5)
  - Step into (F7) — *used to run your program <u>line by line</u>*
- To see how the variables change:

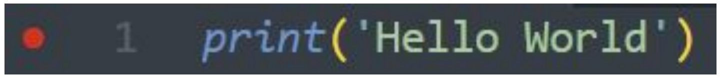
  Tools -> Stack Data





Debugging in VS Code

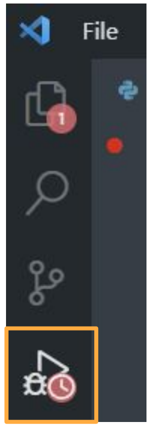
- Add a breakpoint to the line you'd like to **start** debugging with
    - Press F9 ●
    - Or, click on the line



```
print('Hello World')
```

- Press **F11** to start debugging
- Keep pressing **F11** to <u>run the program *line by line*</u>
- To see how the variables change:

  Run and Debug section (on the left)



Debugging with *print()* Statements

- Sometimes it's quicker to trace the code with `print()` statements
- Just add a `print()` statement if you want to see the value of a variable

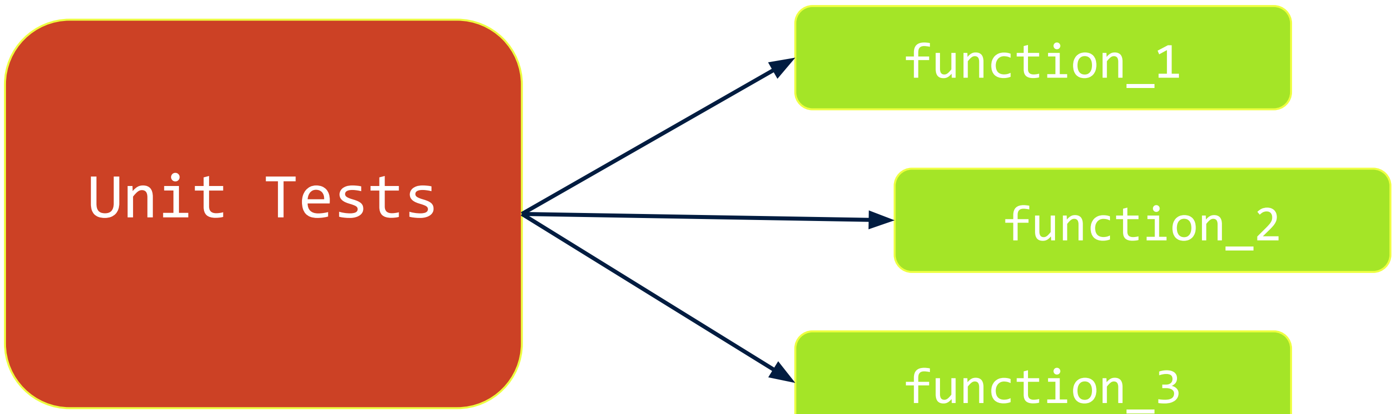
- **Trick:**

```
print(f"{words=}")
```

This format would print both **name** and **value** of a variable

Unit Testing

- Unit testing focuses on the smallest units that comprise a software system:
  - The functions that the programmers create
- True unit testing tests units **in isolation** (each function is tested *separately*)



pytest

- Python library that makes it easy to write small, readable **tests**
- Install:

```
pip install pytest
```

- Run:

```
python -m pytest
```

A Special Way to Call *main()*

```python
if __name__ == "__main__":
    main()
```

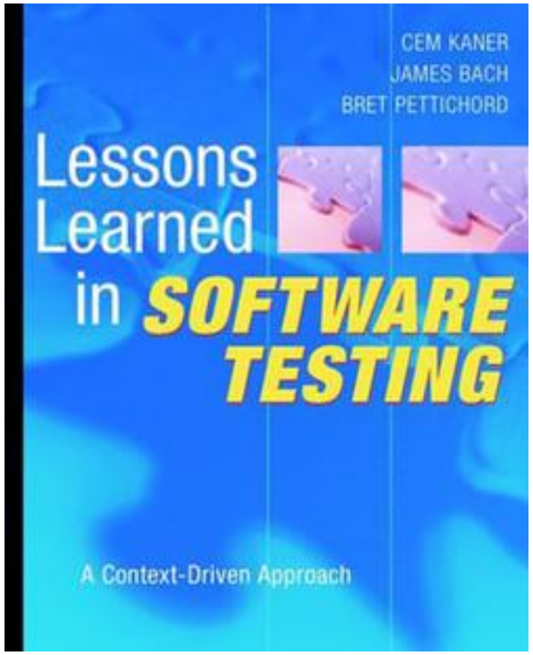
It's boilerplate code that protects users from accidentally invoking the script when they didn't intend to. Here are some common problems when the guard is omitted from a script:

If you import the guardless script in another script (e.g. import my_script_without_a_name_eq_main_guard), then the latter script will trigger the former to run at import time and using the second script's command line arguments. This is almost always a mistake.

Source: https://stackoverflow.com/a/419185/4732334

References

- *Lessons Learned in Software Testing: A Context-Driven Approach*
    - by Cem Kaner, James Bach, Bret Pettichord


CEM KANER
JAMES BACH
BRET PETTICHORD
Lessons Learned in SOFTWARE TESTING
A Context-Driven Approach

https://learning.oreilly.com/library/view/lessons-learned-in/9780471081128/