Ceng 111 – Fall 2021
Week 6b

Credit: Some slides are from the “Invitation to Computer Science” book by G. M. Schneider, J. L. Gersting and some from the “Digital Design” book by M. M. Mano and M. D. Ciletti.

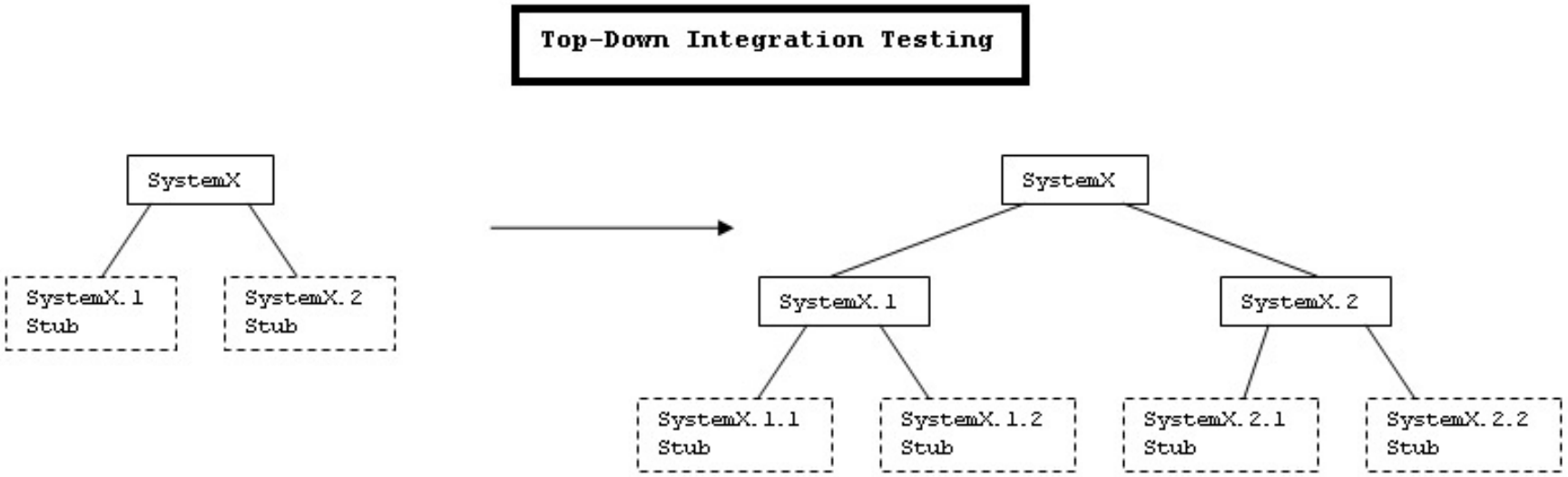
How is a program written?

- Modular & Functional Breakdown
- For example:
  - User interface module
  - Database module
  - Control module

Previously on CENG111!

Testing

- Top-down Testing

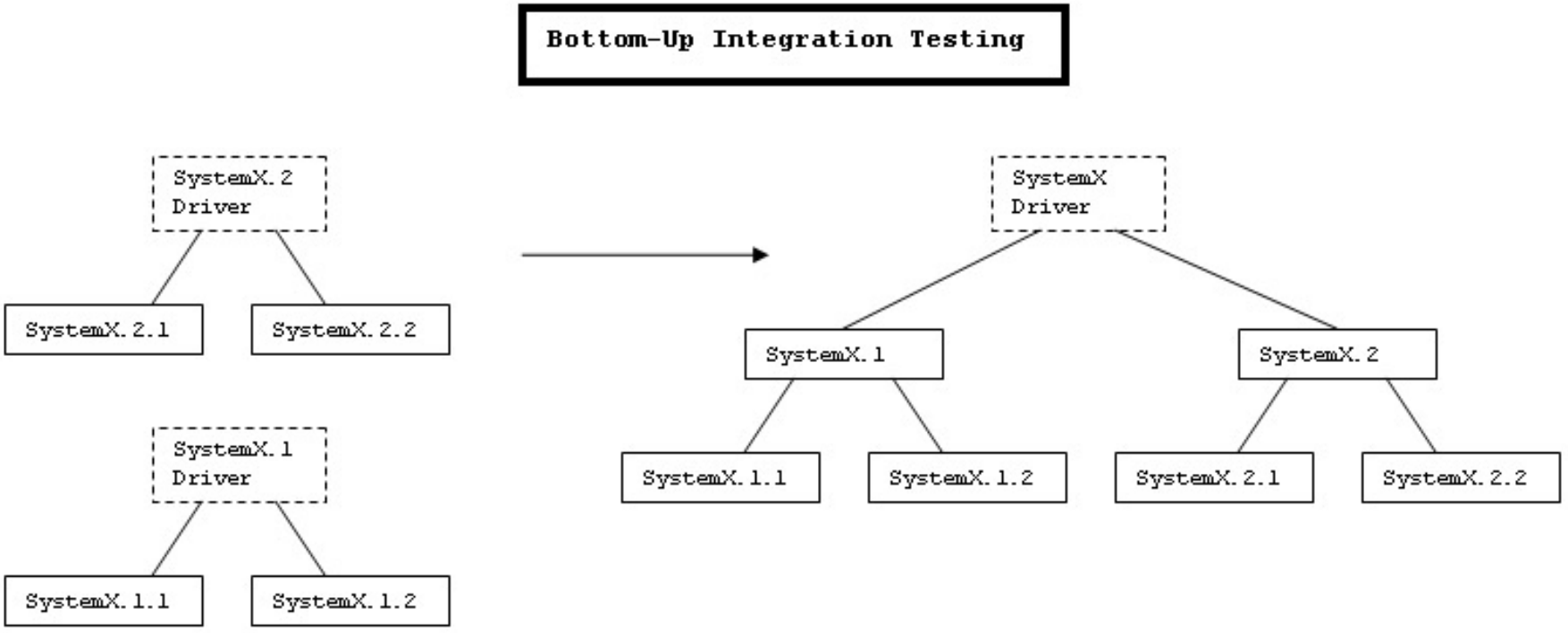


http://sce.uhcl.edu/whiteta/sdp/subSystemIntegrationTesting.html

Previously on CENG111!

Testing

- Bottom-up Testing

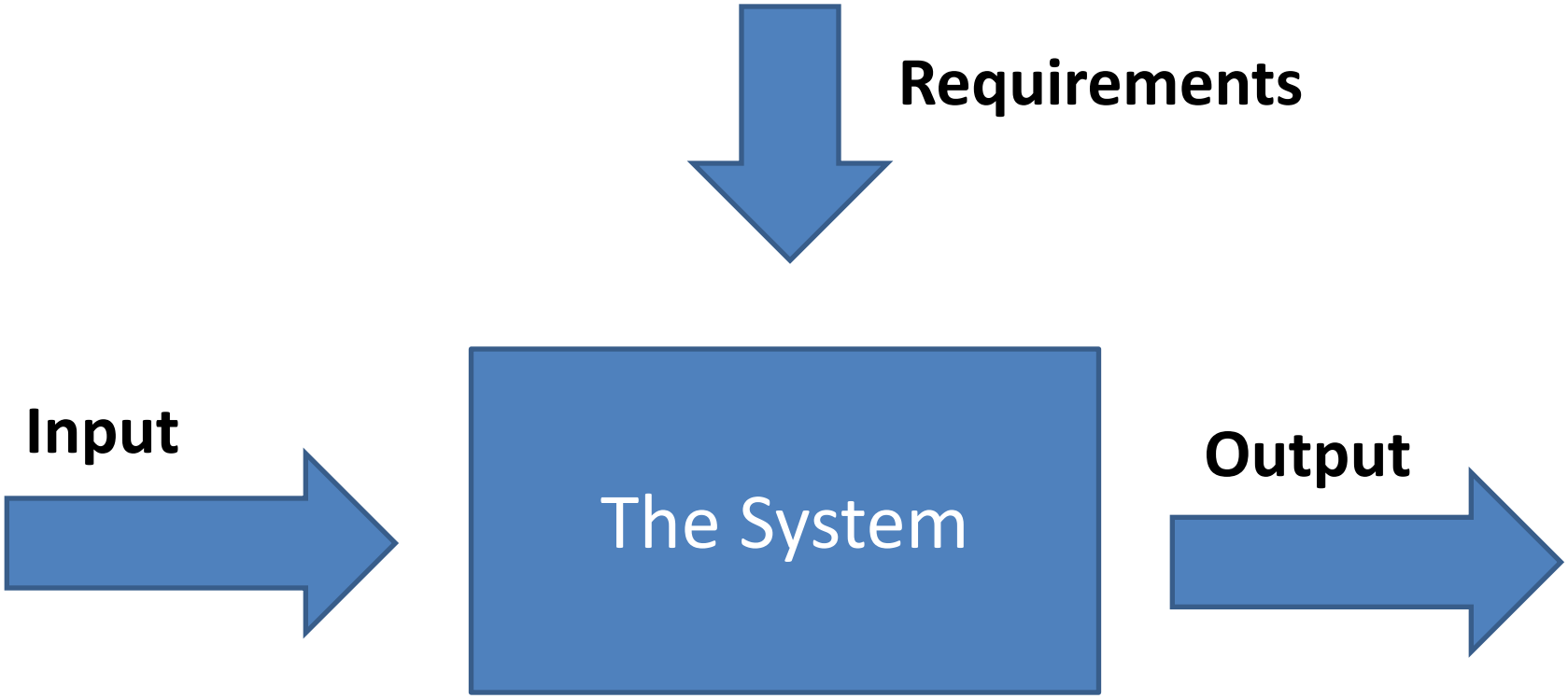


http://sce.uhcl.edu/whiteta/sdp/subSystemIntegrationTesting.html

Previously on CENG111!

Testing

- Black-box Testing



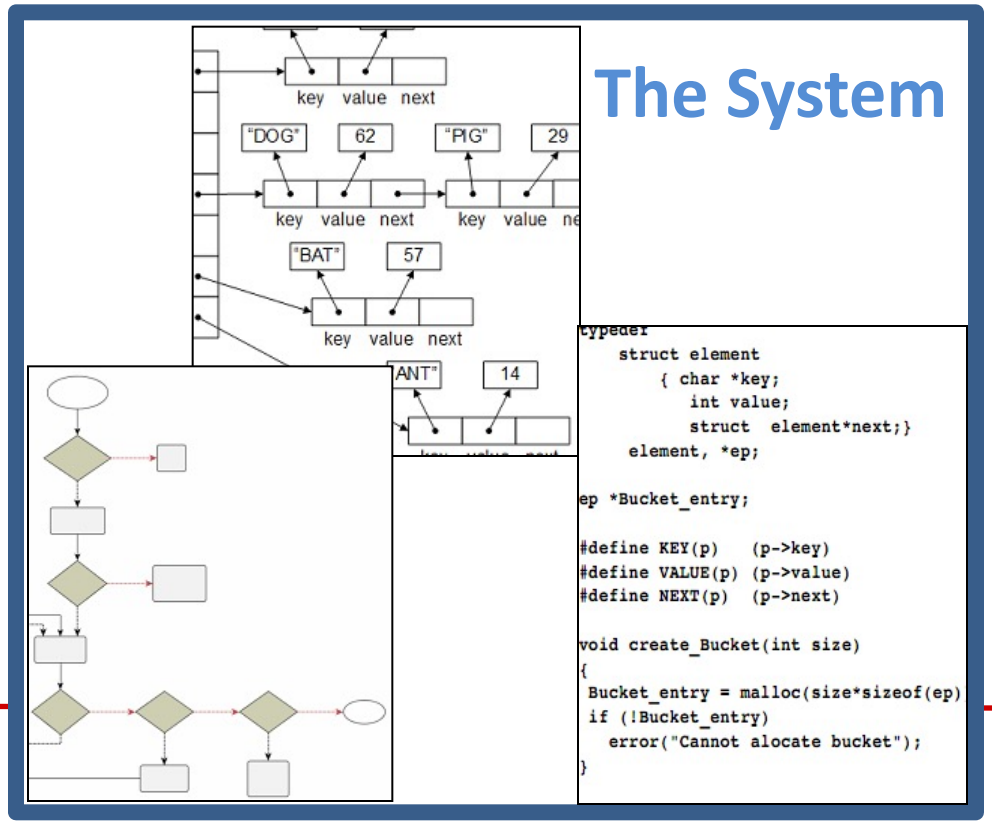
Previously on CENG111!

Testing

- White-box Testing

Requirements

Input



Output

Previously on CENG111!

METU Computer Engineering

Bugs, Errors

- Syntax Errors

  Area = 3.1415 * R * R

  Area = 3.1415 x R x R

- Run-time Errors

```
>>> def SqrtDelta(a,b,c):
>>>       return sqrt(b*b - 4*a*c)
>>>
>>> print SqrtDelta(1,3,1)
2.236067977497898
>>> print SqrtDelta(1,1,1)
ValueError: math domain error
```

Previously on CENG111!

Bugs, Errors

- Logical Errors

$$root_1 = \frac{-b + \sqrt{b^2 - 4ac}}{2a}$$

```
>>> root1 = (- b + sqrt(b*b - 4*a*c)) / 2*a
```

- Design Errors

$$x^3 + ax^2 + bx + c = 0$$

$$root_1 = \frac{-b + \sqrt{b^2 - 4ac}}{2a}$$

Today

- Meet Python
- Data Representation

Administrative Notes

- Scratch assignment
- Midterm date:
  - 22 December, Wednesday, 18:00

MEET python™

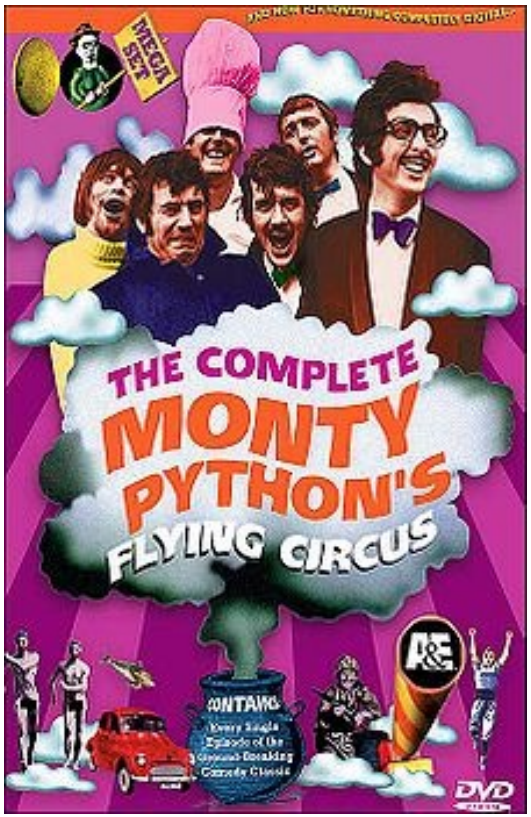
Guido van Rossum (1956 -)

- An interpretive/scripting PL that:
  - Longs for code readability
  - Ease of use, clear syntax
  - Wide range of applications, libraries, tools
- Zen of Python [https://en.wikipedia.org/wiki/Zen_of_Python]
  - Beautiful is better than ugly.
  - Explicit is better than implicit.
  - Simple is better than complex.
  - Complex is better than complicated.
  - Flat is better than nested.
  - Sparse is better than dense.
  - Readability counts.
  - …

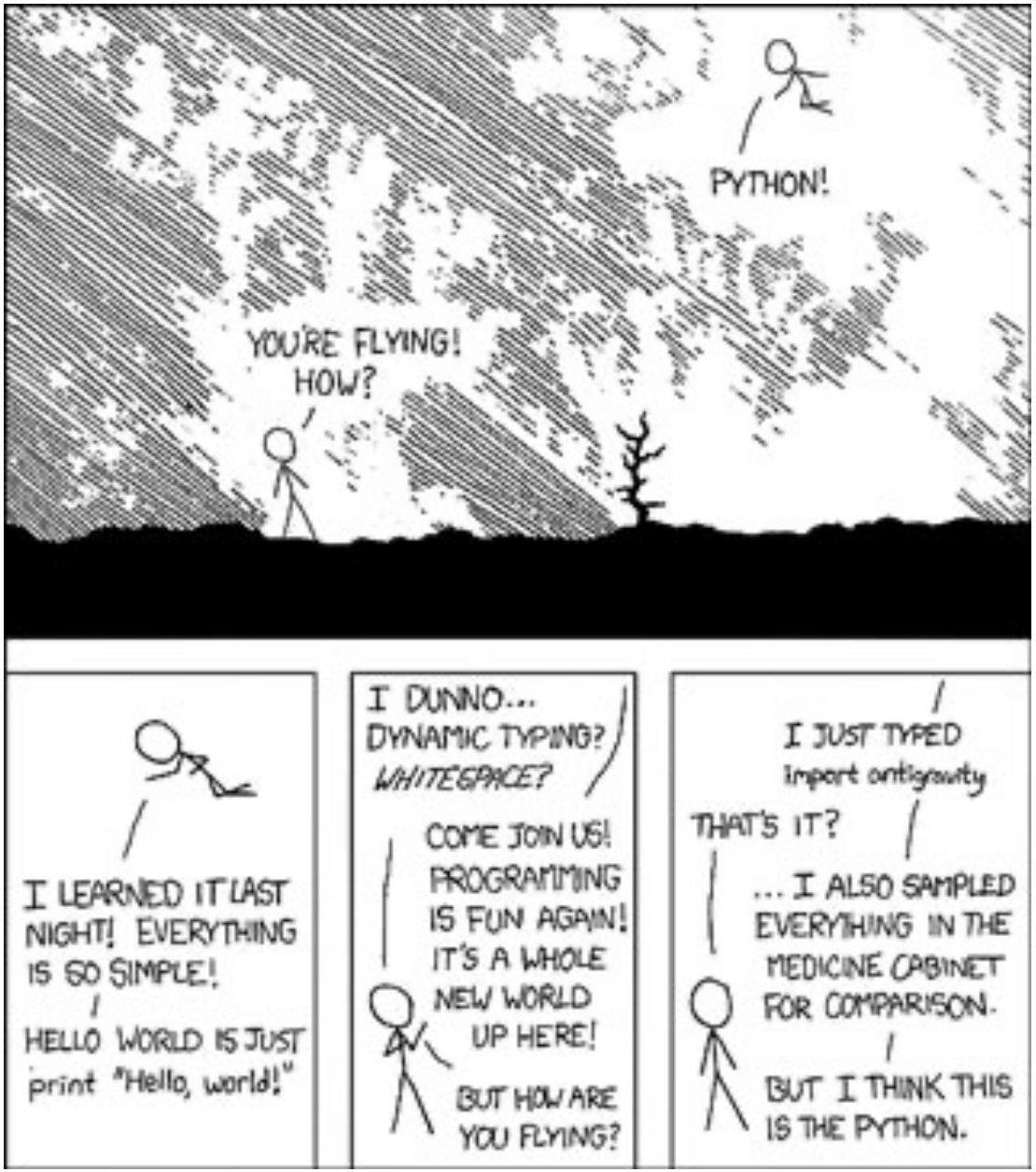
- Supports multiple paradigms:
  - Functional
  - Imperative
  - Object-oriented

- Started at the end of 1980s.
- V2.0 was released in 2000
  - With a big change in development perspective: Community-based
  - Major changes in the facilities.
- V3.0 was released in 2008
  - Backward-incompatible
  - Some of its features are put into v2.6 and v2.7.

- Where does the name come from?
  - While van Possum was developing Python, he read the scripts of Monty Python's Flying Circus and thought 'python' was "short, unique and mysterious" for the new language [1]
- One goal of Python: "fun to use"
  - The origin of the name is the comedy group "Monty Python"
  - This is reflected in sample codes that are written in Python by the original developers.



[1] https://docs.python.org/2/faq/general.html#why-is-it-called-python


PYTHON!
YOU'RE FLYING! HOW?
I LEARNED IT LAST NIGHT! EVERYTHING IS SO SIMPLE!
HELLO WORLD IS JUST print "Hello, world!"
I DUNNO... DYNAMIC TYPING? WHITESPACE?
COME JOIN US! PROGRAMMING IS FUN AGAIN! IT'S A WHOLE NEW WORLD UP HERE!
BUT HOW ARE YOU FLYING?
I JUST TYPED import antigravity
THAT'S IT?
... I ALSO SAMPLED EVERYTHING IN THE MEDICINE CABINET FOR COMPARISON.
BUT I THINK THIS IS THE PYTHON.


```
skalkan@divan:~$ python
Python 2.5.2 (r252:60911, Jan 24 2010, 17:44:40)
[GCC 4.3.2] on linux2
Type "help", "copyright", "credits" or "license" for more information.
>>>
```