Discrete Structures: Programming Constructs CMPSC 102

Oliver BONHAM-CARTER

Fall 2022

Week 3

Slides 01

Let's Discuss

Key Questions

How do I use **iteration** and **conditional logic** in a Python program to perform computational tasks like processing a file's contents and mathematical tasks like using Newton's method to approximate the square root of a number?

Learning Objectives

To **remember** and **understand** some discrete mathematics and Python programming concepts, setting the stage for exploring of discrete structures.

Python Programming Retrospective

- Python code is to be **intuitive**
- Key components of Python programming include:
  - Function calls
  - Assignment statements
  - Iteration constructs
  - Conditional logic
  - Variable creation
  - Variable computations
  - Variable output

Python Programming Retrospective

We are *gonna* ...

- Investigate the **syntax** and **semantics** of these components
- Understand how to **connect** these components together in a program
- **Implement** Python functions to **understand** mathematical functions

A program is a sequence of statements

To be philosophical for a moment ...

Going back to this program ...

```
file = open("emails")
for line in file:
  name, email = line.split(",")
  if name == "John Davis":
    print(email)
```

Programming parallels cooking ...

- A Python program is a sequence of statements about mixing things with the rest of the ingredients ... like a *recipe*
- There is a list of ingredients
- There is a sequence of events about when to use each ingredient
- Timing (run time) is important
- (Chef, waiter, guests) == (programmers, instructions, users)

Programs contain simple and compound statements

Going back to this program ...

```
sum = 0
count = 0
file = open("observations")
for line in file:
  n = int(line)
  sum += n
  count += 1
print(sum/count)
```

Looking closer ...

- Programs contain both **simple** and **compound** statements (i.e., steps having multiple processes on one line)
- Which of these statements are simple?
- Which of these statements are compound?

Industry Standard Python

Be like the Python professionals!

- Please, please, please, use Python 3 for all of your programs!
- Python2 is no longer supported...
- Add **docstrings** to your Python programs (i.e., informed comments to help others follow reasoning behind the code, including)
  - Modules
  - Classes
  - Functions
- Add comments to enhance understanding of important lines of code
- Create command-line interfaces using **Typer** and **Poetry** when shipping code

Industry Standard Python

Be like the Python professionals!

- Add **comments** for **important blocks** of your program
- Use **descriptive** variable and function **names**
- The book does not always adhere to industry standards!
- All course projects will enforce these standards in GitHub Actions

Let's Look a the *"Coolest Thing"* ever!!

BTW: This graphic was found as a result of searching, *The Coolest Thing* on the Internet. Apologies.

Quadratic Root Calculation

Quadratic Equation

$$ax^2 + bx + c = 0 \tag{1}$$

Quadratic Formula

$$x_{1,2} = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a} \tag{2}$$

Special Note

Note the $x_{1,2}$ to imply that there are **two** solutions (i.e., x_1 and x_2) to find for a second degree equation as observed from the x^2.

Quadratic Root Calculation

The Problem Defined

To solve

$$x^2 + 3x - 4 = 0 \quad (3)$$

Want to have roots

$$x_1 =? \text{ and } x_2 =?$$

Finding Roots of an Equation

Let's work an example by hand ...

Quadratic Formula

$$x_{1,2} = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a} \tag{4}$$

Two solutions: x_1 and x_2

Knowns: a = 1, b = 3, c = -4

$$x_{1,2} = \frac{-3 \pm \sqrt{(-3)^2 - 4(1)(-4)}}{2(1)} \tag{5}$$

$$x_{1,2} = \frac{-3 \pm \sqrt{25}}{2} \tag{6}$$

1. **Root 1** using addition of $\pm$: $x_1 = \frac{2}{2} = 1$,
2. **Root 2** using subtraction of $\pm$: $x_2 = \frac{-8}{2} = -4$

Creating Solutions