Introduction to:
Computers & Programming:
Exercises Using:

(the print function
and
Mathematical Operators
and
Data Types)

Adam Meyers
New York University
Exercise 1 to 3

1. What are some simple ways to print the same thing 3 times, with a space in between each instance? Let's assume one variable as input to the function `print_3_times`.

2. What is a simple way to print the same thing 3 times, each on a newline?

3. How would you convert an arbitrary temperature from Fahrenheit to Celsius?
How to Solve a Complex Problem

• Break down big problem into small problems
  – For each small problem either:
    • Solve the problem if you know how
    • Break it down further if you don't know how to solve it and:
      – Try to solve each of the smaller problems
      – Etc.
Until we find problems easy enough to solve.
Exercise 4

• Given
  – Today's date:
    • Month, day, year, and day of the week
  – An arbitrary number of days in the future (e.g., 1500)

• How should we go about identifying that day in the future?
  – Month, day, year and day of the week
How Can We Solve This Problem?

1. Some Pieces of the Problem
   a) Find the year
   b) Find the month
   c) Find the day
   d) Find the day of the week

2. Are all calculation dependent on each other or are any independent of the others, i.e., is there an order?

3. Write Out Pseudo Code for Solving This Problem
   • We can implement this late in the term
Finding the Year

• Find Year based on Days in Future from January 1\(^{st}\) of the current year
  – Find out how many days so far in this year
    • Add the number of days in the previous months plus the number of days in the current month
      – Define the number of days in each preceding month
      – Make an account for the leap year when calculating the number of days in February
        » Leap year iff: year \(\%\) 4 = 0
    • \(\text{New\_year} = \left(\frac{(\text{days\_in\_future} - \text{day\_of\_year})}{365}\right) + \text{current\_year}\)
Find new month

- \( \text{day\_offset} = ((\text{days\_in\_future} - \text{day\_of\_year}) \mod 365) \)

- Subtract number of days in each consecutive month and stop when there are more days in month X than there are days left (allow for February to be varied due to whether new\_year is a leap year or not). That month is the new month.

- Days left is the new day
Day of the Week

• Assign numbers to the days of the week
  – Sunday = 0, .... Saturday = 6
• New Day of Week = (current_day + days_in_future) % 7
• Look up day in table.
Summary

• A Few Simple Problems
  – Implemented in Python

• One Larger Problem
  – Broken Down into Small Problems
  – The Problems were Ordered
  – Then Solved Using Pseudo Code

• Limitations of Our Current Python Programming:
  – Variables
  – If/Then Structures
  – Repetition Structures