Topic 2a – Python Review

Sample Python Scripts
• Python 2 vs. Python 3
• example1.py: simple types, expressions, I/O
• example2.py: simple functions, control structures
• example3.py: lists, dictionaries, files
• Example4.py: functions and parameters

References
• Appendices A of course text.
• Supplementary materials – installing python and using python on school servers
A Tale of Two Pythons

• Python changed between version 2.5 and 3.0
• Version 2.6 and 2.7 are “intermediate versions” that support (some) features of both
• For example

    print "Hello world!"  # old format of print
    print("Hello world!")  # new format of print

• Python 2.7.3 supports both
• We’ll use version available 3.2.3 and higher
A Tale of Two Pythons

Python 2.7.3 and 3
• `print("hi")`
• `input('prompt')`
• `open("in.txt", "r")`
• `9 // 2 => 4`

Python 2.x
• `print "hi"
• `raw_input('prompt')`
• `file("in.txt", "r")`
• `9/2 => 4`

Only in Python 3
• `9 / 2 => 4.5`
• `range(n)`

xrange(n), range(n)
example1.py

Part 1.1

• no semi-colons to mark the end of a statement
• quotes can be 'single' or "double" for a string
• backslash for special characters
• print evaluates to None but output is always a side effect
Part 1.2

• the type() commands tells you want type or class a variable or constant is
• int, float, str, and bool are primitive types
• floats are stored inexacty
• int(), float(), str() convert variables and constants
• the operator '+' means different things to different types
• for strings it means concatenation (i.e. join together)
example1.py

Part 1.3
• precedence rules (when calculating totalPoints)
• no type declarations
• console i/o using input()
• can use a string for the prompts with input()
• long lines can be split with a backslash '\'
• running as a script versus running in the shell
• in shell we still have access to global variables
Flow of control

• `in guessNumber(): while ... :`
• `in printDiamond(): for i in range(...):`
• `in num5s(): if ... : elif ... : else:`
• don't forget colons in flow of control structures
• indentation matters

Boolean (bool) expressions

• can contain variables
• evaluates to False: `[], "", 0, None, False`
example2.py

Return types
• no explicit return type for guessNumber() and printDiamond()
• they return None

Parameters
• Discussed later
Lists, Dictionaries and Files

• indexing, slicing and concatenating lists
• empty lists, dictionaries
• tuples do not support mutation (modification)
• many useful methods exist (e.g. append, extend, sort)