CSC 108H: Introduction to Computer Programming

Summer 2012

Marek Janicki

Administration

- Final is Thurs. Aug 16, 7-10 in SF 3201
  - Office hours have moved to BA2230
  - They will be F2-4 this week, M4-6 and W2-4 next week
  - Need 40% to pass
  - Old Finals posted.
- Assignment 1 autotesting is out.

Quick Correction

- To get access to the specific object that caused an exception one uses:

```
try:
    block
except Exception as e:
    #block can now reference e
    block
```

Course Review

- Types
  - int, float, bool, str
  - list, dict.
    - be wary of aliasing.
- Control flow syntax
  - If statements
  - loops
  - return vs. print

Course Review

- Structure of Programming.
  - Functions, modules, classes.
  - reusing code, making it extendable.
- Meta-Programming
  - Testing
  - Code Design
  - Documentation
  - Complexity
  - Algorithm Design

Immutable Type Review

- Ints and Floats.
  - represent numbers
  - can use +, -, *, /, **.
  - can be compared with ==, <, >, etc.
    - returns a boolean.
  - If there is a float in one operation, the result gets cast to a float.

Immutable Type Review

- Bool
  - can be True or False.
  - has the following operators: and, or, not.
  - Used for control flow structures.
    - So is generated by ==, <, >=, not, etc.
- str
  - called strings.
  - Used to represent sequences of characters.
  - comes with lots of methods.

Mutable Type Review

- Lists
  - Is a list of items.
  - Can be looped over.
  - Can check if elements are inside of the list.
  - Each list element can be changed
  - Have aliasing problems.
  - Often used to store related data.
  - Size can change.
  - Has lots of methods.

Mutable Type Review

- Dictionaries
  - Should be viewed as a set of key:value pairs.
    - The keys must be immutable and unique
    - The values can be anything.
  - Can easily (and quickly) check if a dictionary contains a key.
  - Checking if it contains a value is much slower
    - linear time, not constant.
  - Has many useful methods.

Control flow

- If statements allow one to selectively execute blocks of code.
  - else statement can allow for binary decisions.
  - elif statements should be used when choosing between lots of different possible exclusive choices.
- Loops allow one to repeat a bit of code.
  - For a fixed number of times in the case of a for loop.
  - Until some condition is true for a while loop.

Structure of Programming

- Functions
  - Used to chuck code into easy to understand bits.
  - Used to avoid repeating code.
  - Can return a value if it is specified.
  - Otherwise return none.
- Modules
  - Used to group related functions together.
  - Names can be reused across modules.

Structure of Programming

- Classes
  - Essentially user-made types.
  - Classes have their own variables and methods.
  - Inheritance allows for easy extendability of code.
  - The class methods should be like a user manual for the class.

Meta-Programming

- Testing and Code Design
  - Tied together.
  - Thinking of one should influence the other.
  - At all points you should be thinking about how to test your code, and what those tests mean for your code design.
  - Similarly, knowing your code design should help you think about tests.
  - In general want to test 'generic behaviour' as well as 'corner cases'.
  - Useful to design tests adversarially.

Meta Programming

- Documentation.
  - If one is writing code that will be used constantly, one should treat it as if one is writing code for someone else.
  - Docstrings should contain type information.
  - They should describe what a function/method does, but not how it does it.
  - Modules/Classes should also have generic docstrings

Meta Programming

- Algorithm Design and Complexity
  - Algorithm design is the first step of solving a problem.
  - Algorithms are usually fairly general, and so are judged by 'complexity'
    - a high-level sense of how the time to solve the problem scales with the input.
- Accurate time-measurements are used more for refinement of solutions than for generating them.

Things that were covered at aren't on the final.

- Media
- Inheritance
- Exceptions.

Where to go from here.

- Graphical User Interface(GUI)
  - Really finicky.
  - TKinter is a python module used to design guis.
- Choose a project
  - Probably the best way to learn programming.
  - You should have enough fundamentals to be able to look up modules online and use them.
- Default values
  - Can make code cleaner.

Where to go from here.

- Programming Courses.
  - These will cover meta-programming skills
    - commonly used templates
    - commonly used data structures
    - common approaches to working within a group/organising large amounts of code.
- Theory Courses.
  - Show the math of computer science.
  - Give high level templates to generic problems.
    - These can be instantiated.

Where to go from here.

- Online Resources
  - Khan Academy
    - lectures
  - Software Carpentry
    - lectures
  - Codecademy
    - small problems that build off of eachother.

Break

Exam Overview