Week 9

Diana Gage

www-personal.umich.edu/~drgage

Python + Exam Review & Questions

Any general questions? Topics you want to go over?

Python Review

- Main Differences:
  - True and False are capitals
  - Python floors with int division (matter with negatives: -3 / 2 = -2
  - No variable declarations (automatically interprets based on what you assign it to)
    - my_string = "hello"
  - Python has no ++ operator
  - Review lecture slides for more details – it is fairly straightforward
  - **Use str[-1] to access last element in string**
  - What about **str[-2:]**?

Python Review

- Main Differences:
  - True and False are capitals
  - Python floors with int division (matter with negatives: -3 / 2 = -2
  - No variable declarations (automatically interprets based on what you assign it to)
    - my_string = "hello"
  - Python has no ++ operator
  - Review lecture slides for more details – it is fairly straightforward
  - **Use str[-1] to access last element in string**
  - What about **str[-2:]**? → *lists from 2nd to last til end*

Python Review

- raw_input() is how you extract from the input stream
  - Put prompt inside parentheses! → returns a string
  - Works like getline() → grabs until newline
- Raise to a power using **
- Concatenation using +
- Multiplication using *
  - Works on strings (see lecture slides!)
- **print** is like cout but no '<<' needed
  - Each new line that you have print is a new line in output; no 'endl' needed
  - Add a comma to force output on the same line
    - This comma adds a space between the two outputs

Python Review

- You can think of a **list** as an array that can also:
  - Contain different data types
  - Change its size dynamically (grow)
  - Start out with any number of elements (no need to declare or decide on a size)
- You can access elements with brackets [ ] just like with an array

Python Review

- If, elif, else – very similar to C++ (logic is the same, see lecture slides for syntax examples)
- Loops – **while** and **for** loops

```
while condition:
        #do something
```

```
for <variable> in <container>:
        #do something
```

  - variable can be anything (make a new one)
    - Refers to each successive member in the container
  - Container is any type that holds other values (str, list)
  - You ***can't modify*** the elements of the container

Python Review

- Another variation of **for** loop

```
n = 5
for i in range(n):
    #do something
```

- Range(n) creates a list of values from 0 to n-1
  - It can also accept 2 parameters as a range
- Use range() and len() to loop over a list when you want to **change values**

```
for k in range(len(my_str)):
        my_str[k] = "x"
        print my_str[k]
        #will change every element in string to 'x'
```

Exam Review (new topics)

- **2D arrays! (C++)**
  - Iterating through them
  - Accessing elements (loops still very important)
- **File streams (C++)**
- **Classes: (C++)**
  - Header files vs. cpp files (what goes where)
  - Constructors, getters, setters
  - Public vs. private
- **Python!**
  - everything covered this week
- All EECS183 material is valid, just expect the focus to be on the newer material

How to Study Python:

- CodeAcademy course (if you have time…)
- Make a side-by-side chart with C++ and Python and fill in syntax for different operations
  - Declarations, comments, functions, types, conditionals, loops, etc.

File Stream Practice

- What do you need to #include?
- Write code to:
  - Create a filestream named myFile
  - Open a txt file called “notes.txt”
  - Read in the first line of text into a variable named note1
  - Print that line to standard output
  - Close the file

File Stream Solution

```
#include <fstream>
#include <string>
#include <iostream>
using namespace std;

int main(){
        ifstream myFile;
        myFile.open("notes.txt");
        string note1;
        getline(myFile, note1);
        cout << note1 << endl;
        myFile.close();
}
```

Exam Review Packet

- Another IA (Erin) made a packet of practice exam problems (from 183 study and old exams)

- You may not finish it in class, but give it your best shot!
  - You can work together
  - Solutions are on 183 study/eecs183.org
    - Do as many practice exams as you can!
    - eecs183.org → resources → exams