**COMP 1800 – Fall 2016**

Study Guide for Midterm Exam

* Major computer hardware components and what they do (CPU, RAM, storage, GPU, PSU, etc.)
* How to count in different number bases
* Number conversions:
  + Converting a number in any base into base 10
  + Converting a base 10 number into any base
  + Converting between bases 2 and 16
* Hexadecimal color codes – how to produce them, how to extract color information from them
* Boolean expressions:
  + Boolean operators: **AND** (&&), **OR** (||), **NOT** (!)
  + Evaluating Boolean expressions (order of operations: parentheses, NOT, AND, OR)
  + Constructing basic truth tables
* Basics behind computer software: a programmer writes source code in a high-level programming language, which has to be translated by a compiler or interpreter into a CPU’s native machine language before it can run.
* Know what an algorithm is
* Basics of the Python programming language:
  + Interactive vs. script mode
  + Using the **print** command to display things on the screen
  + Using **#** to write comments in a program
* Python expressions
  + Data types: **int**, **float**, **str**
  + Basic mathematical operators (+, -, \*, /, \*\*)
  + Integer division and mod (//, %)
  + Order of operations in numerical expressions (parentheses, exponents, multiplication/division/mod from L to R, addition/subtraction from L to R)
  + Concatenation (+) and repetition (\*) in string expressions
  + How expressions work when **int**, **float**, and **str** are mixed
* Variables
  + Assigning values to a variable using the assignment operator (=). Assignment does NOT mean “is equal to.” It means “evaluate the expression on the right, and assign that result to the variable on the left.”
  + Allowable names for variables – can’t contain spaces, can’t begin with a number
  + Remember that a variable’s value DOES NOT CHANGE until you assign it something else!
  + Be able to trace a sequence of assignment statements and keep track of what values are being stored in each variable
* Reading user input: be able to use Python’s **input** command to allow the user to enter input as the program is running. Remember that **input** returns a **str** value by default; you must use **int** or **float** around the **input** to convert it if necessary.
* There will be questions involving reading as well as writing Python code. Be sure you’re able to look at a piece of code and trace the actions that the code is performing. The key here is to go line by line (write things like variable values down if you have to) – don’t just try to glance at the code and figure it out all at once!