1. ­­­­­Objects and Classes and Git (Week 1)
   1. Expectations
      1. This class will not progress if you guys don’t work or if I don’t work. There is no grade here, no incentive. The only incentive is that you work hard to save time, money, and learn about the field of computer science.
      2. Don’t give up your long term goals for your short term pleasures
      3. I know that you may be unsure if Computer Science really is the field for you—and because of that you probably might be wondering if this class would really be beneficial. But this is the perfect opportunity to find out if you want to pursue this in college or not.
      4. Computer Science is a field that is not only lucrative and demanding (lot of jobs opening in the near future), but also it helps you develop a way of thinking that is unique for problem solving that will help you in anything.
   2. What are objects and what are classes
      1. <http://www.tutorialspoint.com/java/java_object_classes.htm>
      2. <http://www.wideskills.com/java-tutorial/java-classes-and-objects>
      3. <http://www.leepoint.net/JavaBasics/oop/oop-10-intro.html>
      4. <https://www.youtube.com/watch?v=pFrFhA42cXs>
      5. <https://www.youtube.com/watch?v=DivykEY4z1g>
      6. <https://www.youtube.com/watch?v=Gm5s_MnjYGg>
      7. <http://www.tutorialspoint.com/java/java_basic_syntax.htm>
   3. Install Git and use Git to submit homework, projects, etc
      1. Git will be used for projects
      2. Access using ssh
2. Computer Basics (Week 2) **DONE except CE**
   1. Computer ethics: piracy, copyright, intellectual property, freeware, shareware
      1. <http://www.infosectoday.com/Articles/Intro_Computer_Ethics.htm>
      2. <https://www.blackwellpublishing.com/content/BPL_Images/Content_store/Sample_chapter/9781855548442/CEAC01.pdf>
      3. <http://apcentral.collegeboard.com/apc/members/courses/teachers_corner/45296.html>
   2. Computer terminology
   3. Understanding how parts of the computer work together and separately
3. Java Basics + Using Compiler (Week 3-4)
   1. Editing and compiling simple program
   2. Running java on IDE (from command line Windows?)
   3. How and where memory is stored
      1. <http://www.cs.usfca.edu/~parrt/doc/java/JavaBasics-notes.pdf>
      2. <http://tutorials.jenkov.com/java-concurrency/java-memory-model.html>
   4. Initializers and Constants
   5. Expressions + return statements
      1. <https://docs.oracle.com/javase/tutorial/java/nutsandbolts/expressions.html>
      2. <http://www.tutorialspoint.com/java/java_regular_expressions.htm>
      3. <https://www.artima.com/objectsandjava/webuscript/ExpressionsStatements1.html>
      4. <https://www.d.umn.edu/~gshute/java/expressions.html>
4. Input/Output (week 5)
   1. Reading in with buffered reader
   2. Using system.println to print out information
5. Variables and Comments (Week 6) **DONE**
   1. Primitive Types
      1. <http://www.cs.usfca.edu/~parrt/doc/java/JavaBasics-notes.pdf>
      2. <https://docs.oracle.com/javase/tutorial/java/nutsandbolts/variables.html>
   2. Commenting
      1. Benefits of commenting regularly and effectively
   3. Naming of variables
      1. Useful and meaningful names that would help read and utilize code
6. Looping and conditionals [control flow](Week 7-8)
   1. while , do-while, and for loops
      1. <http://www.tutorialspoint.com/java/java_loop_control.htm>
      2. <http://www.homeandlearn.co.uk/java/java_for_loops.html>
      3. <http://www.learnjavaonline.org/en/Loops>
      4. <http://www.landofcode.com/java-tutorials/java-loops.php>
   2. Switch statements (maybe)
      1. <https://docs.oracle.com/javase/tutorial/java/nutsandbolts/switch.html>
      2. <http://www.homeandlearn.co.uk/java/java_switch_statements.html>
      3. <http://www.dummies.com/how-to/content/switch-statements-in-java.html>
      4. <http://www.tutorialspoint.com/java/java_decision_making.htm>
   3. If, else-if, else
      1. <https://docs.oracle.com/javase/tutorial/java/nutsandbolts/if.html>
      2. <http://www.homeandlearn.co.uk/java/java_if_statements.html>
      3. <https://www.youtube.com/watch?v=C0YRYVn_BeI>
7. Arithmetic Expressions (Week 9)
   1. Addition, subtraction, multiplication, division, modulus,
      1. <https://docs.oracle.com/javase/tutorial/java/nutsandbolts/opsummary.html>
      2. <https://docs.oracle.com/javase/tutorial/java/nutsandbolts/op1.html>
      3. <https://docs.oracle.com/javase/tutorial/java/nutsandbolts/op2.html>
      4. <http://www.w3resource.com/java-tutorial/java-arithmetic-operators.php>
      5. <http://www.tutorialspoint.com/java/java_basic_operators.htm>
8. Representing numbers in different bases (Week 10) **DONE**
   1. Interchanging from binary, hex, and decimal values
9. Creating and using classes (Week 11-12)
10. Strings (Week 13-14)
    1. <https://docs.oracle.com/javase/tutorial/java/data/strings.html>
    2. <http://www.tutorialspoint.com/java/java_strings.htm>
    3. <https://www3.ntu.edu.sg/home/ehchua/programming/java/J3d_String.html>
    4. <http://beginnersbook.com/2013/12/java-strings/>
11. Arrays (Week 15-16)
    1. Intro to Data Structures
       1. <https://www.youtube.com/watch?v=92S4zgXN17o>
    2. <http://www.tutorialspoint.com/java/java_arrays.htm>
    3. <http://introcs.cs.princeton.edu/java/14array/>
    4. <http://stackoverflow.com/questions/1200621/declare-array-in-java>
    5. <http://tutorials.jenkov.com/java/arrays.html>
    6. <https://chortle.ccsu.edu/java5/Notes/chap49C/ch49C_3.html>
    7. <http://www.homeandlearn.co.uk/java/multi-dimensional_arrays.html>
12. ArrayLists (Week 17)
13. Searching/Sorting Arrays (Week 18-20)
14. GridWorld (Parts 1-3) (Week 21-23) NOT SURE WHAT THIS IS
15. Classes, Inheritance, and Interfaces (Week 24-26)
16. Recursion and Merge Sort (Week 27-29)
    1. Understand what recursion is
    2. Understand what Merge Sort is
    3. How is recursion useful + how does it save time
17. REVIEW