* 2012-2013 (High School)
  + Java Programming Honors
    - First exposure to hands-on coding
* 2014-2015
  + Intro To Computer Science I
    - Unit Conversion Web App
    - Python Flask App
      * Hosted on PythonAnywhere.com
      * HTML in Python
    - Picture Puzzle Minigame
      * Python Kivy
      * Interactive Game
    - Stock Analysis
      * Retrieving data from Internet services using urllib2
      * CSV File parsing
  + Intro To Computer Science II
    - Java round two
    - Graphical User Interfaces
      * Photo Library Slideshow
        + Displays all photos within a file directory in an applet
    - Blackjack
      * Allows players to interactively play the game of blackjack against a java program
    - Basic sorting and search algorithms
    - Class dependency
* 2015-2016
  + Intro To Software Development
    - Complex Regexes
    - Very basic SQL
    - Multi-Threading and thread-safety
    - Design of complex software
    - Version Control
    - Search Engine
      * Back-end
        + Over the course of a semester I worked on a complex search engine designed to scan and analyze files and webpages for word entries \* and store them in an inverted index structure that kept track of word location and frequencies of each word. When parsing webpage code, this project stripped out all HTML code but saved nested, valid links (including fragments and #\*) to be parsed as well
      * Front-end
        + I also added abilities for my search engine to be deployable to local host and be usable from a standard web browser, prompting for web addresses to scan and displaying an index of words found from that provided link
  + Intro To Parallel Programming
    - Basic C and pointers
    - Message Passing Information (MPI), PThreads, CUDA/GPU
    - Memory/Cache
    - For this class I became familiar with writing parallelized programs using either MPI or PThreads. I had three large scale projects:
      * Dijkstra’s Algorithm
      * Bitonic Sort
        + Merge Sort
      * Conway’s Game of Life
        + Game states
        + Cell states
  + Data Structures and Algorithms
    - In this class we covered a wide range of different Abstract-Data-Types (ADTs)
      * Movie Recommender
        + Linked-Lists
        + Using an informational database of users->movie->rating, was able to create recommendations/anti-recommendations for a user
      * Huffman Encoding
        + File compression
        + Huffman Trees
      * Persistent Data Types
        + Binary Search Tree
        + Stack
      * Dijkstra’s Algorithm
        + Hashing
        + Adjacency Lists/Matrixes
        + Binomial Queue
      * Network The Game
        + Board-game-like design
        + Artificial Intelligence
    - Algorithm Time-Space Complexity
    - Spanning Tree/Minimal Cost Spanning Trees
  + Linear Algebra and Probability
    - LaTex Documents
    - Sage.Math Fractal design
    - Processing 3 Fractal-based movie
  + Flash Animations