

COURSE OVERVIEW

Course Title: Level 3: Data Structures & Algorithms

Course Number:

Number of Units: 4.0 units

Total Hours of Instruction: 40

Course Description

During this course, students study and practice data structures and algorithms. All serious programmers need to be data virtuosos. In Level 3 students learn to master lists, stacks & hashmaps and to implement basic sorting and searching algorithms that are fundamental to computer science. This course is designed for intermediate programmers aged 13-16 years old.

Content and Evaluation

The curriculum for this course is a proprietary set of coding challenges that build knowledge of data structures and algorithms in a way that appeals to kids. For example, students build an iPod Shuffle using ArrayLists, and a Hangman game that uses Stacks and HashMaps. Many of the challenges are visual and engage the student creatively, with many references to pop culture, and using stories that make complex ideas sticky for young minds.

In this level, we begin to solve problems from previous College Board Advanced Placement Computer Science exams, and hone skills using assignments from Stanford University CS 106A. We use the Java programming language exclusively.

To complete this course, students take a 1.5 hour exam that consists of a short written portion, and two substantial coding exercises that must be completed independently.

Extra Credit

n/a

Methods of Instruction

- | | | | |
|-------------------------------------|------------------------------------|-------------------------------------|--------------------------------|
| <input checked="" type="checkbox"/> | CLASS DISCUSSION/DISCUSSION BOARDS | <input checked="" type="checkbox"/> | LECTURES |
| <input type="checkbox"/> | FIELD TRIPS | <input type="checkbox"/> | CASE STUDIES |
| <input checked="" type="checkbox"/> | GROUP WORK | <input checked="" type="checkbox"/> | OTHER: PROGRAMMING ASSIGNMENTS |

Out of Class Assignments

Total hours expected to complete assignments: n/a

- | | | | |
|--------------------------|--------------------|--------------------------|-----------------------------|
| <input type="checkbox"/> | TEXTBOOK EXERCISES | <input type="checkbox"/> | READINGS |
| <input type="checkbox"/> | GROUP WORK | <input type="checkbox"/> | WRITTEN ASSIGNMENT/ESSAY(S) |

STUDENT PROJECT

X OTHER: EXPLORATION OF CONCEPTS BY PROGRAMMING AT HOME

Evaluation/ Grading

X EXAM(S)

WRITTEN ASSIGNMENT/ESSAY(S)

X OTHER: TWO PROGRAMMING EXERCISES

CLASS PARTICIPATION/DISCUSSION BOARDS

CLASS PROJECT(S)

Topical Outline

Rather than follow a traditional lecture-style/textbook form of teaching, this course is driven by a set of practical learning objectives that are practiced until the student gains mastery over that skill. In the same way a shoemaker might make a lot of shoes before becoming a master, our students continually write code to solve a particular challenge in the theme of the skill they are learning. As a shoemaker would not learn to make shoes by spending time listening to someone describe the theory of lasting and hammering, a young programmer does not learn by listening to an adult explain the theory of computer science.

At the end of Level 3, students have mastered these skills;

1. Data Structures / Java Collections API

- ArrayList
- Stack
- HashMap
- enhanced for-each loop

2. Linear Algorithms

- basic sorting (swapping)
- find minimum/maximum
- searching
- data processing using web services (Twitter, Flickr)

3. Intermediate Software Engineering

- Unit Testing with JUnit
- Using the Eclipse debugger
- Naming conventions
- Refactoring