

## CD1 Study Guide

CD1 addresses topics from Chapters 1-2 in your textbook which were studied over weeks 1-4. It covers two topics/outcome sets in the course:

### Topic 1 - Object Oriented Code

- Identify and explain the key concepts of object-oriented programming, including classes, objects, methods, inheritance, polymorphism, encapsulation, and abstraction.
- Recognize and describe the purpose and structure of Python classes and objects in a provided code snippet.

### Topic 2 - Algorithm Analysis

- Employ appropriate vocabulary to discuss algorithmic efficiency, including terms like Big O notation, time complexity, and space complexity.
- Analyze code to determine its execution-time (big-oh notation) and storage utilization.

If you go back through the “Outcomes” sections of weeks 1-4 you will see that it addressed a fair amount of vocabulary.

### Week One

- **Algorithm**
- **Abstraction**
- Computable
- **Program**
- Programming
- Data types
- Abstract Data Types (ADTs)
- **Encapsulation**
- **Information Hiding**
- Data Structure
- **Implementation**
- Exceptions
- Pseudocode

## Week Two

- Functions
- Methods
- **Class**
- **Object**
- **Encapsulation**
- **Abstraction**
- **Inheritance**
- **Polymorphism**
- **Constructor**
- **`__init__`**
- Override(s)
- Magic Methods
- Subclass
- Superclass

Some of that is clearly repetition from prior courses and some of it is brand new. I think you will need to understand all of that vocabulary as the course goes on. But this competency demo will focus on a subset of these (in bold) as the others rely on context coming in future chapters.

Additionally, the “Outcomes” sections focused on some specific skills:

- Explain the idea of Big-Oh (Omega) notation.
- Recognize the name and symbol for each common function type discussed in Big-Oh notation. function
- Given two or more Big-Oh notation/function types put them in order from slowest growth to fastest growth. [Currently limited to  $O(1)$ ,  $O(n)$ , and  $O(n^2)$ ]
- Given an algorithm, compute it Big-Oh notation. [Currently limited to  $O(1)$ ,  $O(n)$ , and  $O(n^2)$ ]
- Given an algorithm, write an accurate 2-3 sentence description of what it will do.

You should make sure that you can meet each of these objectives/skills.