# CSE 2017-03 Data Structure and Lab.

# Spring 2018

This course is an introductory course in data structures. Topics include arrays, stacks, queues, linked lists, recursion and binary trees. The student will build projects using the data structures taught in class. Proper software engineering techniques are introduced, including object oriented design and programming and unit testing.

**Professor**: Eun Man Choi **Email**: [emchoi@dongguk.edu](mailto:emchoi@dongguk.edu)

**Course Web Page**: eclass.dongguk.edu

**Office Hours**: M 1:00-2:00 PM **Office Location**: Room 10119, New Engineering Bld.

**TA**: Jun Sun Whang(hwangsun12@naver.com)

**Class Hours**: T, Th 1:00-3:00 PM **Class Location**: 3183(Lab), 6114(Lecture) NEB.

**Texts**: 1. Nell Dale and David Teague: [C++ Plus Data Structures](file:///C:\Users\emcho\Documents\Teaching\DS(C++)\2017f\C++%20Data%20Structures%203rd%20ed%20-%20Nell%20Dale.pdf), fourth edition, Jones and Bartlett, 2007.

2. James Roberge: [Data Structures in C++: A Laboratory Course](file:///C:\Users\emcho\Documents\Teaching\DS(C++)\2017f\A.Laboratory.Course.In.C.Plus.Plus.Data.Structures.2nd.Editi.pdf), second edition, Jones and Bartlett, 2003.

**Prerequisites**: Knowledge of C or C++: CSE 2014 or PRI 4035.

**Grading**:

* + Project(#1~4): 15%
  + Labs(#1~12): 20%
  + Midterm: 30%
  + Final Exam: 30%
  + Attendance and Participation: 5%

**Grading Scale**: A= 85% B=70% C=60% D=50% F<50%

**Exam Failure:** The instructor reserves the right to assign a failing semester grade to any student who fails either (or both) exams.

**Lateness:**  All assignments are due at the beginning of class on the due date. Assignments turned in late will automatically have one full grade deducted. No assignments will be accepted one week or later after the due date.

**Academic Integrity**

Any indication of copying lab or project work or any behavior during exams that could be considered copying or cheating will result in an immediate zero on the assignment/exam for all parties involved. The student’s advisor/department and the Dean will be notified. Cheating on assignments is defined to be copying from someone else or providing someone else copies of your answers. Do not show your assignments to anyone else! You may answer questions on labs or project homework asked by other students.

## Schedule for Lecture and Laboratory

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Week** | **Class Topic & Contents** | **Chapter on Text** | **Assignment** | **Project** |
| 1 | 3/6 [Lecture 00: Orientation and Introduction to Data Structure](file:///C:\Users\emcho\Documents\Teaching\DS(C++)\2017f\00_Orientation.pdf)  3/8 [Lecture 01: Review and introduction to C++](file:///C:\Users\emcho\Documents\Teaching\DS(C++)\2017f\01_C++Introduction.pdf)(1) | Text 1: Chapter 2 |  |  |
| 2 | 3/13 Lecture 01: Review and introduction to C++(2)  3/15 [Lab 01: Practice in C++ programming and debugger](file:///C:\Users\emcho\Documents\Teaching\DS(C++)\2017f\Tutorial%20VC++.pdf) | Text 2: Chapter 1 |  |  |
| 3 | 3/20 [Lecture 02: Data Abstraction and List(array)](file:///C:\Users\emcho\Documents\Teaching\DS(C++)\2017f\02_DataAbstraction%20and%20List.pdf)  3/22 Lab 02: Array Implementation of the List ADT | Text 1: Chapter 3  Text 2: Lab 3 List ADT | Laboratory 2: Logbook In-Lab  *(pp.13)* |  |
| 4 | 3/27 [Lecture 03: Stack(Array) and Pointer](file:///C:\Users\emcho\Documents\Teaching\DS(C++)\2017f\03_Stack.pdf)  3/29 Lab 03: Stack(Array) | Text 1: Chapter 4  Text 2: Lab 4 Ordered List | Laboratory 3: In-lab Exercise #2 *(pp.60)* |  |
| 5 | 4/3 [Lecture 04: Stack(Linked)](file:///C:\Users\emcho\Documents\Teaching\DS(C++)\2017f\04_Linked%20Stack.pdf)  4/5 Lab 04: Stack(Linked) | Text 1: Chapter 4  Text 2: Lab 5 Stack ADT | Laboratory 4: In-lab Exercise #1 | Assign [Project #1](file:///C:\Users\emcho\Documents\Teaching\DS(C++)\2017f\Project_1.htm) |
| 6 | 4/10 [Lecture 05: Queue(Array, Linked)](file:///C:\Users\emcho\Documents\Teaching\DS(C++)\2017f\05_Queue.pdf)  4/11 Lab 05: Queue(Array, Linked) | Text 1: Chapter 5  Text 2: Lab 6 Queue ADT | Laboratory 5: In-lab Exercise #1 |  |
| 7 | 4/17 Lecture 06: [List Plus(Double Linked List)](file:///C:\Users\emcho\Documents\Teaching\DS(C++)\2017f\06_List%20plus.pdf)  4/19 Lab 06: Double Linked List | Text 1: Chapter 6  Text 2: Lab 9 Double Linked List ADT | Laboratory 6: In-lab Exercise #2, 3 |  |
| 8 | 4/24 Midterm exam  4/26 No Lab | Preview Exam | Laboratory 9: In-lab Exercise #2 | Assign [Project #2](file:///C:\Users\emcho\Documents\Teaching\DS(C++)\2017f\Project_2.htm) |
| 9 | 5/1 [Lecture 07: Recursion](file:///C:\Users\emcho\Documents\Teaching\DS(C++)\2017f\07_Recursion.pdf)  5/3 Lab 07: Recursion with Linked List | Text 1: Chapter 7  Text 2: Lab 10 Recursion |  |  |
| 10 | 5/8 No Class  5/10 [Lecture 08: Tree](file:///C:\Users\emcho\Documents\Teaching\DS(C++)\2017f\08_BinarySearchTree.pdf) | Text 1: Chapter 8 | Laboratory 10: In-lab Exercise 1 |  |
| 11 | 5/15 [Lecture 09: Expression Tree](file:///C:\Users\emcho\Documents\Teaching\DS(C++)\2017f\09_Expression%20Tree.pdf)  5/17 Lab 08: Binary Search Tree | Text 1: Chapter 8  Text 2: Lab 11 BST | Laboratory 11: In-lab Exercise 1 | [Assign Project #3](Project_3.htm) |
| 12 | 5/22 No Class  5/24 Lab 09: Expression Tree | Text 1: Chapter 9  Text 2: Lab 12 Expression Tree | Laboratory 12: In-lab Exercise 2 |  |
| 13 | 5/29 [Lecture 10: Heap](file:///C:\Users\emcho\Documents\Teaching\DS(C++)\2017f\10_Heap(2017).pdf)  5/31 Lab 10: Heap | Text 2: Lab B Heap | Laboratory B: In-lab Exercise 1 | Assign Project #4 |
| 14 | 6/5 [Lecture 11: Graph(1)](file:///C:\Users\emcho\Documents\Teaching\DS(C++)\2017f\11_Graph.pdf), [Graph(2)](file:///C:\Users\emcho\Documents\Teaching\DS(C++)\2017f\12_Graph2.pdf)  6/7 Lab 11: Weighted Graph | Text 1: Chapter 9  Text 2: Lab 13 Weighted Graph | Laboratory 13: In-lab Exercise 1 |  |
| 15 | 6/12 [Lecture 12: Sorting](file:///C:\Users\emcho\Documents\Teaching\DS(C++)\2017f\13_Sorting.pdf)  6/14 Lab 12: Sorting | Text 1: Chapter 10  Text 2: Lab 14 Hashing | Laboratory 14: In-lab Exercise 1 |  |
| 16 | 6/19 Final Exam |  |  |  |