

Bronx Community College
Department of Mathematics and Computer Science
Course Outline FALL 2019

Course: CSI33-E01 Data Structures

Prerequisite: CSI 32 and CSI 35 and ENG 02 and RLD 02, if required

Time and location: Monday and Wednesday 6:00-7:50pm in CP320.

TEXT: Data Structures and Algorithms Using Python and C++ , by David M. Reed and John Zelle, Franklin Beedle and Assoc.

Instructor: Dr. Sharon Persinger

Office: CPH 306

Office hours: Wednesday 5:00-6:00pm in CP306 **E-mail:** sharon.persinger@bcc.cuny.edu

Course web page : <https://fsw01.bcc.cuny.edu/sharon.persinger/>

Course overview and goals: Students in this course will learn a variety of ways to organize large collections of data in order to solve problems more efficiently. Students will learn some methods for analyzing the efficiency of algorithms. These data structures will be implemented using object-oriented programming techniques in the programming languages Python and C++.

Course objectives: A student who successfully completes this course will be able to

1. write a specification of a function, design a function, and implement it in Python and C++
2. write a specification of a class, design a class, and implement it in Python and C++,
3. understand and use the basic data structures of stacks, queues, linked lists, binary trees
4. implement a linked list and functions operating on linked lists,
5. understand the use of pointers and dynamic memory allocation in implementing data structures,
6. use big-O analysis to compare the running time of data structure operations with different implementations,
7. solve searching and sorting problems efficiently using appropriate data structures.

Schedule of exams:

There will be an in-class midterm exam on October 23. There will also be a comprehensive final exam whose date will be announced later.

Make-ups:

If you miss an exam, you will be permitted to make up that exam only if you can provide a medical excuse or other significant written reason for your absence.

Programming exercises:

You learn to program by writing programs. The programming exercises are a necessary part of this course. There will be 8-10 programming exercises, some small ones and some larger ones. Programming exercises in total will be worth 200 points.

Grading:

Your total grade is based on 400 points: Midterm exam 100 points, Final Exam 100 points, and Programming exercises 200 points.

I use the College's recommended grading scale:

Academic Grade	A+	A	A-	B+	B	B-	C+	C	C-	D+	D
Numerical Average	97-100	93-96.9	90-92.9	87-89.9	83-86.9	80-82.9	77-79.9	73-76.9	70-72.9	67-69.9	63-66.9
Points out of 400	388-400	372-387	360-371	348-359	332-347	320-331	308-319	292-307	280-291	268-279	252-267

Academic integrity:

Giving or receiving help on an exam is cheating. Using unauthorized notes, books, or other aids during an exam is cheating. Any student who cheats on an exam will receive a grade of 0 for that exam

Copying another person's program is cheating. Any student who submits another person's program as his or her own work will receive a 0 for that assignment; any student who allows his or her work to be copied will receive a grade of 0 for that assignment

Assistance:

If you have questions, please send me an email, come to my office hours, or make an appointment to see me at another time.

Holidays and other important dates:

9/2/2019	Monday	College Closed
9/5/2019	Thursday	Classes follow Monday schedule
9/30/2019-10/1/2019	Monday-Tuesday	No classes scheduled
10/8/2019-10/9/2019	Tuesday-Wednesday	No classes scheduled
10/14/2019	Monday	College Closed
10/16/2019	Wednesday	Classes follow Monday schedule
11/28/2019-12/1/2019	Thursday-Sunday	College Closed
12/13/2019	Friday	Reading Day
12/14/2019-12/20/2019	Saturday-Friday	Final Examinations

Schedule of classes		
Monday	Wednesday	
	August 28, 2019 Day 1 Introduction, review of syllabus Chapter 1: Abstraction and Analysis 1.2 Functional Abstraction 1.3 Algorithm analysis	
September 2, 2019 Labor Day No classes, college closed	September 4, 2019 Day 2 1.3 Algorithm analysis Chapter 2: Data Abstraction 2.2 Abstract Data Types 2.3 ADTS and Objects	Thursday, September 5, 2019 Monday schedule, Day 3 2.4 An Example ADT: Datasets 2.5 An Example ADT: Rational Programming Exercise

Schedule of classes	
Monday	Wednesday
September 9, 2019 Day 4 Chapter 3: Container Classes 3.2 Python Lists 3.3 A Sequential Collection: A Deck of Cards 3.4 A Sorted Collection: Hand	September 11, 2019 Day 5 3.6 Python Dictionaries Programming exercise
September 16, 2019 Day 6 4.3 The Python Memory model 4.3 A Linked Implementation of Lists 4.4 Linked Implementation of a List ADT	September 18, 2019 Day 7 4.4 Linked Implementation of a List ADT 4.5 Iterators 4.7 Lists vs. Arrays
September 23, 2019 Day 8 Programming exercise	September 25, 2019 Day 9 5.2 Stacks 5.3 Queues
September 30, 2019 No classes	October 2, 2019 Day 10 5.4 Queue Implementation 5.5 An Example Application: Queueing Simulations Programming exercise
October 7, 2019 Day 11 6.2 Recursive Definitions 6.3 Simple Recursive Examples 6.4 Analyzing Recursion	October 9, 2019 No classes
October 14, 2019 No classes, college closed	October 16, 2019 Day 12 6.5 Sorting 6.6 A “Hard” Problem: The Tower of Hanoi Programming exercise
October 21, 2019 Day 13 7.2 Tree Terminology 7.3 An Example Application: Expression Trees Midterm exam review	October 23, 2019 Day 14 Midterm Exam on Chapters 1, 2, 3, 4, 5, 6
October 28, 2019 Day 15 7.4 Tree Representations 7.5 An Application: A Binary Search Tree Programming Exercise	October 30, 2019 Day 16 8.2 C++ History and Background 8.3 Comment, Blocks of Code, Identifiers, and Keywords 8.4 Data Types and variable declarations 8.5 Include Statements, Namespaces, and Input/Output

	8.6 Compiling
November 4, 2019 Day 17 8.7 Expressions and Operator Precedence 8.8 Decision Statements 8.9 Type Conversion 8.10 Looping Statements	November 6, 2019 Day 18 8.11 Arrays 8.12 Function Details 8.13 Header Files and Inline Functions 8.14 Assert Statements and Testing 8.15 The Scope and Lifetime of Variables
November 11, 2019 Day 19 8.16 Common C++ Mistakes by Python Programmers Programming exercise	November 13, 2019 Day 20 9.1 Basic Syntax and Semantics 9.2 Strings 9.3 File Input and Output 9.4 Operator Overloading 9.5 Class Variables and Methods
November 18, 2019 Day 21 10.2 C++ Pointers 10.3 Dynamic Arrays	November 20, 2019 Day 22 10.4 Dynamic Memory Classes 10.5 Dynamic Memory Errors

Schedule of classes	
Monday	Wednesday
November 25, 2019 Day 23 11.2 A C++ Linked Structure Class 11.3 A C++ Linked List 11.4 C++ Linked Dynamic Memory Errors	November 27, 2019 Day 24 12.2 Template Functions 12.3 Template Classes
December 2, 2019 Day 25 13.2 Priority Queues and Heaps 13.5 Hash Tables	December 4, 2019 Day 26 15.2 Divide and Conquer
December 9, 2019 Day 27 15.3 Greedy Algorithm	December 11, 2019 Day 28 Programming lab Review for final exam