

Week 11 & 12 Overview

QUICK LINKS[Study Guide](#)[Objectives](#)[Instructions](#)

Study Guide

Message	These two weeks will be a mix of theory followed by actual implementation of abstract data types. Please pay attention to the programming and try to recreate the experience yourself. Noting beats typing yourself when you want to learn how to program.
Reminders	
Due Dates	<ul style="list-style-type: none">• DQ 10<ul style="list-style-type: none">○ Your initial post is due by 11:59 PM ET on Thursday of Week 11.○ Your replies are due by 11:59 PM ET on Monday of Week 11.• DQ 11<ul style="list-style-type: none">○ Your initial post is due by 11:59 PM ET on Thursday of Week 12.○ Your replies are due by 11:59 PM ET on Monday of Week 12.• Assignment 4 is due by 11:59 PM ET on Monday of Week 12.

[Top Menu](#)

Objectives

We will look at more abstract data types and implement them in both Python and Java. These two languages behave quite differently on the surface, but you should see patterns emerge. We will learn about

- Stacks
- Queues
- Deques and
- Linked Lists

With each ADT we will attempt a suitable implementation. In Python we will mostly fall back onto lists, whereas in Java we will try to use basic types as much as possible. This will reinforce our programming skills.

[Top Menu](#)

Instructions

Topics:

- More Linear Data Structures – Stacks, Queues, and Deques

- Even More Linear Data Structures – Linked Lists

Lesson 11

1. **Read** the following lecture slide shows.
 - Lecture 9-1 Stacks, Queues, and Deques in Python
 - Lecture 9-2 Stacks, Queues, and Deques in Java
2. **View** the following lecture videos via the Media Gallery.
 - Lecture 10-1 Stacks, Queues, and Deques in Python
 - Lecture 10-2 Stacks, Queues, and Deques in Java
3. **DQ (Discussion Question) 10.** Create your initial post on the DQ 10 Discussion Board in response to the following questions:
Initial Post due by Thursday 11:59 pm:
 - We have seen the abstract data types stack, queue, deque, and we will look at lists next. These are not the only ADTs. Research and describe an ADT that we have not dealt with. Discuss implementation ideas as an actual data structure in either Python or Java.

For your Response to at least **2 classmates due by Monday, 11:59 pm:**

- Comment on their findings and discuss possible implementations of the ADT they chose.

After posting, return to the board and read over the posts of your fellow classmates. Choose at least two classmates and create a post responding to their initial post on the Discussion Board.

As you read and respond to others' posts, keep in mind some basic rules for netiquette:

- Be kind and respectful to others
- Use full sentences
- Do not use too much jargon
- Treat others online as you wish to be treated
- Use language that supports others

Lesson 12

1. **Read** the following lecture slide shows.
 - Lecture 11-1 Linked Lists in Python
 - Lecture 11-2 Linked Lists in Java
2. **View** the following lecture videos via the Media Gallery.
 - Lecture 11-1 Linked Lists in Python
 - Lecture 11-2 Linked Lists in Java

3. **DQ (Discussion Question) 11.** Create your initial post on the DQ 11 Discussion Board in response to the following questions:

Initial Post due by Thursday 11:59 pm:

- We implemented the lists interface using singly linked lists. A more complex form of lists are doubly linked lists. Research and report on some aspects of doubly linked lists.

For your Response to at least **2 classmates due by Monday, 11:59 pm:**

- Comment on their findings and discuss possible implementations of the ADT they chose.

After posting, return to the board and read over the posts of your fellow classmates. Choose at least two classmates and create a post responding to their initial post on the Discussion Board.

As you read and respond to others' posts, keep in mind some basic rules for netiquette:

- Be kind and respectful to others
- Use full sentences
- Do not use too much jargon
- Treat others online as you wish to be treated
- Use language that supports others

4. **Programming Assignment Week 12 (Assignment 4)**

This is the fourth assignment where you will turn in programs. This time there are two different implementations.

ADT Deque in Python

You are to reimplement the ADT deque, using the Python deque from the collections module, but with the interface of lecture 10-2.

You may use the resources available in our course.

- Lecture videos and slides

This exercise is graded on five measures

- Correct interface definition (implicitly in the class methods)
- Correct class and methods definition
- Correct use of deque methods
- Parameter validation in methods
- Testing (meaning you should submit a program that works)

Doubly Linked List in Python

You are to reimplement the ADT list as a doubly linked list class in Python with the interface from Lecture 11-1. Recall that None can be used in place of Java's null.

You may use the resources available in our course.

- Lecture videos and slides

This exercise is graded on four measures

- Correct interface definition (implicitly in the class methods)
- Correct class and methods definition
- Parameter validation in methods
- Testing (meaning you should submit a program that works)

Each of these measures is graded according to the following rubric.

Score	Problem Solving
4	Response gives evidence of a complete understanding of the problem; is fully developed; is clearly communicated.
3	Response gives the evidence of a clear understanding of the problem but contains minor errors or is not fully communicated.
2	Response gives evidence of a reasonable approach but indicates gaps in conceptual understanding. Explanations are incomplete, vague, or muddled.
1	Response gives some evidence of problem understanding but contains major programming or reasoning errors.
0	No response or response is completely incorrect or irrelevant.

This makes for 20 points and 16 points, respectively. Please make sure your programs run without external dependencies. Upload your files as program files (.py). The files must be named in the following format: *Lastname_Assignment4-[Deque/List/class/main].py*. Within your submitted files make a comment as to the naming you used on your own computer, so as to make error tracing easier. Submit your completed document to *Assignment 4*, listed on the Modules and Assignments pages, as file attachments.

[Top Menu](#)