CSCI 102-001: Data Structures, Summer 2017 Course Information and Syllabus

## 05/22/2017 - 08/13/2017 (Summer Session One through Summer Session Two)

TR 5:45-7:50PM

## Room 102 WWH

## Prof. Douglas Moody

Adjunct Assistant Professor,  
Computer Science Department,  
Courant Institute of Mathematical Sciences,  
New York University

**Email address**

dlm9@nyu.edu

**Mailing address**  
251 Mercer Street,   
Room 305  
New York City, 10012

**Office hours – Summer 2017**  
Room 328 WWH   
Tuesdays / Thursdays 4:30 - 5:30 pm.  
before class, and by appt.

**Textbook(s)**

|  |
| --- |
| (REQUIRED)  Object-Oriented Data Structures Using Java, Fourth Edition By Nell Dale, Daniel T. Joyce, Chip Weems ISBN-13: 9781284089097 Jones & Bartlett Learning [Publisher's website](http://www.jblearning.com/catalog/9781284089097/) (If you want to use the third edition, that's fine, but you may need to cross reference the chapter numbers and problem numbers with the 4th edition. ) |
| [Open DSA - online eBook](http://algoviz.org/OpenDSA/Books/OpenDSA/html/)  OpenDSA is an open source project whose goal is to provide online course materials for a wide range of Data Structures and Algorithms courses.  It is currently funded by two National Science Foundation grants: TUES program grant DUE-1139861 and EAGER program grant IIS-1258571 . |
| **DO NOT SELL YOUR CS101 BOOK JUST YET!**  (OPTIONAL)  Introduction to Java Programming, Brief Version, 10/E By Y. Daniel Liang |
| (OPTIONAL)  Think Java: How To Think Like a Computer Scientist, By Chris Mayfield and Allen Downey Open book, can be downloaded from [here](https://open.umn.edu/opentextbooks/BookDetail.aspx?bookId=285) |
| (OPTIONAL)  Java, Java, Java: Object Oriented Problem Solving,  By Ralph Morelli and Ralph Walde Open book, can be downloaded from [here](https://open.umn.edu/opentextbooks/BookDetail.aspx?bookId=218) |

**Prerequisites**

Passing CSCI.UA.0101 with a grade of C or better.

You are expected to know and remember the material from CSCI.UA.0101 course. If you took the course a few semesters ago and/or do not remember parts of the material, start reviewing it during the summer break.

**If you took an equivalent of this course at a different school, you need to make sure that you are familiar with Java. We assume you know enough Java to write fairly large programs right at the beginning of the semester.**

**Grading**

Your grade will be based on:

* exams (65%): one midterm exam (25%) and the final (40%),
* programming projects (30% total) - part of the programming projects grade may be based on in-class quizz given during the lecture/recitation immediately after the project due date (those quizzes are based on the programming assignment)
* quizzes, attendance, other homeworks (5% total)
  + short (usually non-programming) homework assignments might be given throughout the semester

Grades will be determined using the following scale:

A 95-100

A- 90-95

B+ 87-90

B 84-87

B- 80-84

C+ 76-80

C 72-76

D 65-72

F less than 65

The grade of Incomplete is reserved for students who, for legitimate and documented reason, miss the final exam. The grade of Incomplete **will not be given** to student who started falling behind in class. Those students should withdraw from the class or switch to Pass/Fail option.

**Assignments**

There will be three different types of assignments in this course: 1) programming projects, 2) short (usually non-programming) homeworks, and 3) do not hand-in homeworks (DNHI).

**Programming projects (30% of your final grade)** will be given on a regular basis. In general, they will be due one-two weeks after they are assigned. They will require you to write and, often, read significant amount of code.   
No programming projects can be accepted after the last day of classes.

Late and missed programming projects:   
The late project submissions lose 10% of their value for each day they are late, up to 3 days.

Broken programming projects:  
If you hand in a program that does not compile or crashes when it is run, you will get a grade of zero on it. As you are working on your code, make sure that it compiles and does what you expect it to do. Test frequently, not only after you write hundreds of lines of code.

**Homeworks (part of the 5% of your grade together with quizzes and activities)** might be given several times during the semester. In general, they will be due four-five days after they are assigned.   
Late homeworks will be accepted with a deduction in points , but only for 2 days lateness.  
No homeworks can be accepted after the last day of classes.

**Do not hand-in homeworks (DNHI)** will be given to encourage you to practice the material that we discuss in class. The problem sets will be posted as separate homeworks or inicated in the lecture notes. They will serve also as review questions for exams.   
Recitations are the good place to have them discussed!

**Academic Integrity Policy**

I follow the department's [academic integrity rules](http://cs.nyu.edu/webapps/content/academic/undergrad/academic_integrity). In short, it is fine to talk to other students about your ideas and your programs, but it is not fine to work together on assignments or copy someone else's assignment. You cannot copy other people's work without giving them a proper credit (and part of your grade).  
Any sharing or copying of assignments will be considered cheating. By the rules of the College of Arts and Science, I am required to report any incidents of cheating to the department.   
If you have any doubt if something that you are doing qualifies as academic dishonesty, talk to me!

So what is cheating?

* Sharing code/solutions: by copying, retyping, looking at, or supplying a file (this includes posting partial or complete solutions on the course discussion board, a public repository - think GitHub, or any place where others can access it)
* Coaching: helping your friend to write their code line by line (send them to the tutors or office hours, if you want to help them)
* Copying code from previous course or from elsewhere on WWW. You are only allowed to use code from the textbook and what we supply on the course website.
* Hiring someone to complete your assignments for you.

What is NOT cheating?

* Explaining how to use systems or tools
* Helping others with high-level design issue

**Exams**

There will be a midterm and a final exam. All exams are cumulative.

**Missing an exam:** There will be no make-up exams. Failure to take an exam counts as a zero grade on that exam. The only exception to this rule is for students who have a legitimate serious medical or personal emergency (documented). These students need to talk to me as soon as possible (trying to excuse an exam absence three weeks after it happened will not work).

**Topics Covered**

* Java features: abstract classes, interfaces, inheritance, polymorphism, generics, iterators, exception handling, file I/O
* Recursion
* Worst case asymptotic running time analysis
* Abstract data types
* Lists, stacks, queues
* Trees (binary and binary search trees, AVL trees)
* Sorting algortihms (merge sort, quicksort)
* Hashing and hash tables
* Priority Queues

Additional topics (time permitting):

* Graphs
* Other sorting algorithms

**Academic Email Etiquette**

* Check the school email address on a regular basis. You can simply forward its content to another email account that you use regularly.
* Use your school's email account to send emails to professors, instructors, TA's, graders, administrators, etc. OR make sure that your email address contains your true name, not "frabjous@gmail.com", "BabyGurl@yahoo.com" or some other cool alias.
* Start your email with proper salutations! Use the correct titles (Professor, Dr., etc.) and spell first and last names correctly. If you are on the first name basis with your instructors, use their names, not "Hey". For example: "Dear Professor Drummer" or "Dear Robert", not "Hey Bob".
* Sign your name under the body of your email, otherwise you expect people to read emails from anonymous.
* Do not write everything in upper-case letters. Do not write everything in lower-case letters.
* Make sure you included everything you wanted before hitting send. Don't send three emails one after another because you forgot something in the first one.
* Proofread the text in your email before sending it. Most of the email clients check for typos, but they cannot tell if your email makes much sense. Read it, before you send it.