

CSE 260, Computer Science B

Course Information

Fall 2020

Stony Brook University

Instructor: Dr. Paul Fodor

<http://www.cs.stonybrook.edu/~cse260>

Course Description

- *“Second part of a two-semester sequence, CSE 160 and CSE 260. Applies object-oriented programming methodology to data storage and manipulation on complex data sets, such as, binary trees, heaps, priority queues, balanced trees and graphs. Recursive programming is heavily utilized. Fundamental sorting and searching algorithms are examined along with informal efficiency comparisons. Intermediate-level programming language concepts and paradigms, including functional programming, basics of type systems, program and data abstractions, and modularity.”*

(<https://www.cs.stonybrook.edu/students/Undergraduate-Studies/courses/CSE260>)

- Prerequisites: CSE 160
- Corequisite: CSE 261 (1 credit)
- 3 credits

Course Outcomes

- The following are the official course goals agreed upon by the faculty for this course

<https://www.cs.stonybrook.edu/students/Undergraduate-Studies/courses/CSE260>:

- *An ability to define and use complex object-oriented data structures and use them in the context of large-scale software development.*
- *An understanding of the importance of time and memory efficiency in algorithm design.*
- *An understanding of programming language paradigms and tradeoffs.*
- *An understanding of functional techniques to identify, formulate and solve problems.*

Instructor Information

- Dr. Paul Fodor
214 New Computer Science Building
- Office hours: Mondays and Wednesdays 2:30-4pm, on Google Meet <http://meet.google.com/xyu-jhqc-bdx>
- Phone: 1 (631) 632-9820
- Email: paul (dot) fodor (at) stonybrook (dot) edu
 - Please include “CSE 260” in the email subject and your name in your email correspondence
- TAs: see course Web page:
<http://www.cs.stonybrook.edu/~cse260>
- **Blackboard** will be used for assignments, grades and course material.

Class Time and Place

- CSE260 Lectures: MoWe 4:25PM - 5:45PM, on Google Meet
<http://meet.google.com/xyu-jhqc-bdx>
- CSE261 Labs: MoWe 6:05PM - 7:25PM, on Google Meet
<http://meet.google.com/xyu-jhqc-bdx>

Textbooks

- Introduction To Java Programming, Comprehensive Version, Author: Daniel Liang, Publisher: Pearson , Edition: 11th, 2017. ISBN-13: 978-0134670942.
- Scott, Michael L., Programming Language Pragmatics (3rd ed.), Morgan Kaufmann, 2009, ISBN-10: 0123745144.

Grading Schema

- Grades will be based on homework and exams according to the following formula:
 - Homework assignments = 25%
 - Quizzes = 5%
 - Labs = 10%
 - Midterm exam 1 = 20%
 - Midterm exam 2 = 20%
 - Final exam = 20%

Examinations

- Midterm 1: We. 9/30, during classtime (120 minutes) with Lockdown Browser and Monitoring.
- Midterm 2: We. 11/4, during classtime (120 minutes) with Lockdown Browser and Monitoring.
- Final Exam: Wednesday, December 9, 2020, 8:30-10:30pm (120 minutes) with Lockdown Browser and Monitoring.
 - See Final Exams University Schedule here:

<https://www.stonybrook.edu/commcms/registrar/registration/exams.php>

The exams will be like the problems that we solve in the class!

- **The Pass/No Credit (P/NC) option is not available for this course**
 - This policy applies to *all* CSE/ISE undergraduate courses used to satisfy the graduation requirements for the major

Grading

- **Grade Cutoffs:** A [95-100], A- [90-95), B+ [87-90), B [83-87), B- [80-83), C+ [77-80), C [73-77), C- [70-73), D+ [65-70), D [60-65), F [0-60)
- **SPECIAL RULE:** If all your grades, including homework assignments, quizzes, recitation and your three exam grades are above the respective class averages, you're guaranteed to receive a grade of C or higher for this class.
- There will be extra credit problems as a part of quizzes and homework assignments which values to an increase of less than 4% in the final grade.
- There will be in-class quizzes / brief assessments used to practice the class material and measure growth in knowledge, abilities, and skills.
- They will be solved in class and they are valued 2 points each.

Grading

- The final grade you receive in this class will reflect, as far as possible, the extent to which you have mastered the concepts and their applications.
 - How much someone needs a grade, or how close they are to the next higher grade, will have no effect on grade.
 - As the instructor, I want everyone to do well in this course, and will make every reasonable effort to help you understand the material taught.
 - However, the grades provided at the end of the semester are final, except for rare situations involving grading errors.
 - They will not be altered for any reason, so please do not ask me to do so.

Assignments

- Homework assignments due on fixed dates and times.
 - no late submission is permitted
- All assignments should be submitted electronically on Blackboard

Lab exercises

- You will be given problems that require a programmed solution during lab hours. You will have only the lab period to edit, compile, and execute your solution.
- Attendance is mandatory, if you want credit
 - you can leave early only if you showed all your assigned work
 - if you come later than 15 minutes from the start of the lab, then you don't get the credit for the lab
- Demonstrate your work to Lab-TA before you leave for lab credit
 - 0 –3 points:
 - 0 - Student did not attend the lab or program does not even compile.
 - 1 - Student attended the lab, program compiles but has major problems.
 - 2 - Student attended the lab, and program partially works (with some minor errors)
 - 3 - Student attended the lab, and program is correct

Regrading of Homework/Exams

- Please meet with a grading TA or the instructor and arrange for regrading.
- **You have one week from the day grades are posted or mailed or announced!**
- Late requests will not be entertained

Class Schedule

Week	Lecture Topics
1	Generics in Java
2	Lists, Stacks, Queues, and Priority Queues in Java
3	Sets and Maps in Java
4	Developing Efficient Algorithms
5	Sorting
6	Implementing Lists, Stacks, Queues, and Priority Queues
7	Binary Search Trees
8	AVL Trees
9	Hashing
10	Graphs and Applications
11	Weighted Graphs and Applications
12	2-4 Trees and B-Trees
13	Functional Programming
14	Functional Programming
15	Functional Programming

Disability Support Services

- If you have a physical, psychological, medical or learning disability, contact the DSS office at Room 128 ECC. Phone 632-6748/TDD
- If you are planning to take an exam at DSS office, you need to tell me ahead of time for every exam.
- **All documentation of disability is confidential.**

Academic Integrity

- The following rules are posted in every course syllabus:
"Each student must pursue his or her academic goals honestly and be personally accountable for all submitted work. Representing another person's work as your own is always wrong. Any suspected instance of academic dishonesty will be reported to the Academic Judiciary. For more comprehensive information on academic integrity, including categories of academic dishonesty, please refer to the academic judiciary website at http://www.stonybrook.edu/commcms/academic_integrity/"

Academic Integrity

- You can discuss general assignment concepts with other students: explaining how to use systems or tools and helping others with high-level design issues
- You **MAY NOT share** assignments, source code or other answers by copying, retyping, looking at, or supplying a file
 - Assignments are subject to manual and automated similarity checking (We do check! and our tools for doing this are much better than cheaters think)
- If you cheat, you will be brought up on academic dishonesty charges - we follow the university policy:
 - <http://www.stonybrook.edu/uaa/academicjudiciary>

Examples of Academic Dishonesty

- Unpermitted collaboration (on a paper, homework, lab reports, etc.). Unless an instructor has explicitly approved working together, students should assume, for their own protection, that it is not permitted.
- Helping someone else to plagiarize from one's own homework (for example, by giving them a copy of yours, or doing it for them)
 - This includes having a public repository on Github that other students can copy from.
- Representing someone else's source code as one's own. If another person's code is being used, it must be properly cited.
- Buying or selling source code.
- Using source code or pieces of a paper from the internet without properly citing the source.

Academic Integrity

- The instructor makes a recommendation at the Academic Judiciary office
 - Cheating is cheating! No matter the amount of cheating or if one is the source or destination of cheating.
 - Do not cheat! You are cheating yourself.
 - Our job is to teach you the material and make sure that you learn it.
 - Our recommendation is always F for the cheaters!

Catastrophic events

- Major illness, death in family
- Formulate a plan (with your CEAS academic advisor) to get back on track
- Advice
 - Once you start running late, it's really hard to catch up

What do you need to get started?

- Blackboard account
 - <http://blackboard.stonybrook.edu>
- SINC Sites: <http://www.sinc.sunysb.edu/helpdesk/labs.shtml>
- Java JDK standard edition:
 - <http://www.oracle.com/technetwork/java/javase/downloads>
- Eclipse IDE:
 - <http://www.eclipse.org/downloads>
 - Learn to use the debugger!!!
- Liang's student Web site:
 - <http://www.cs.armstrong.edu/liang/intro11e>

Software

- Necessary Software:
 - Java Developer Kit (JDK): download from <http://java.com/en/download/index.jsp>
 - You should download JDK for your operating system (cost: free)
 - Eclipse IDE: <http://www.eclipse.org>
 - You should download the Eclipse IDE for Java Developers (cost: free)
 - Integrated Development Environment (IDE)
 - combines writing, compiling, running and debugging Java code into a single application
 - makes coding much more efficient and organized
 - Eclipse, NetBeans, etc.

Please

- Please be on time
- Please show respect for your classmates
- Please turn off (or use vibrate for) your cellphones

...

- On-topic questions are welcome

Welcome
and Enjoy!