CS 114 Introduction to Computer Science II
Spring 2023

Instructor: Dr. Ravi Varadarajan
Email: ravi.varadarajan@njit.edu
Office Hours: Will be posted in Canvas

Prerequisite : CS 113

Course Goals
The course introduces all basic and some advanced data structures and presents efficient algorithms for solving problems in some areas using these data structures. Students will implement these algorithms in Java and understand the computing resource bounds of an algorithm.

Course learning outcomes

1. Learn asymptotic notations to analyze the growth rate of time and space complexities of algorithms in bounding the resources required by a program.
2. Learn commonly used data structures and their use in solving efficient algorithms for solving problems in various domains such as sorting, searching and common graph problems.
3. Learn some algorithm design strategies for solving problems efficiently.
4. Demonstrate practical applications of data structures and algorithms discussed in the course through implementation of algorithms in Java.

Course Text:
Data Structures & Algorithm Analysis in Java, Edition 3.2, by Clifford A. Shaffer, Dover, 2011. ISBN: 0486485811. You can also access online version of the text book from the author's web page : https://people.cs.vt.edu/shaffer/Book/JAVA3elatest.pdf

Lecture slides from the class will be the primary source of material for the course.

Other references:

1. Algorithm Design: Foundations, analysis, and internet examples by M.T. Goodrich and R. Tamassia, Addison Wesley, 2001, ISBN-0-471-38365-1
2. Introduction to Algorithms, 3rd Edition, MIT Press, Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein, ISBN-13: 978-0262033848

Course Topics:
Week 1 (Jan 16) : Introduction, ADT, Recursion
Week 2 (Jan 23) : Backtracking search
Week 3 (Jan 30): Recursion for optimization problems
Week 4 (Feb 6): Mathematical preliminaries, Proof techniques (Induction, By contradiction etc.)
Week 5 (Feb 13): Algorithm complexity and asymptotic analysis
Week 6 (Feb 20): Lists, Stacks
Week 7 (Feb 27): Queues, Dictionaries
Week 8 (Mar 6): No Lab. Midterm review, **Midterm exam**

Mar 13-19: **Spring Recess**
Week 9 (Mar 20): Binary trees, Binary Search Trees, Balanced binary search trees
Week 10 (Mar 27): Priority Queues, Heaps
Week 11 (Apr 3): Sorting lower bound, Comparison based sorting algorithms (insertion sort, heap sort)
Week 12 (Apr 10): Recursive sorts (merge sort and quick sort), Distribution based sorting (radix sort)
Week 13 (Apr 17): Order statistics and selection, Graphs, Graph search
Week 14 (Apr 24): Graph algorithms, Final exam review

There might be slight deviations of this schedule depending on the pace of the lectures.

Grading Criteria:
Lab assignments (15%), Homework assignments (30%), In-class quizzes (10%), Midterm Exam (15%), Final Exam (30%). Lowest three scores will be dropped for quizzes and homework assignments in computing the final grade.

Tuesday classes are devoted to lab sessions. Lab attendance is mandatory for acceptance of lab submissions and assignments are expected to be completed during the lab session or by the end of the day.

HomeWorks which are mainly programming assignments will be assigned every week on Wednesday and will be due the following Thursday. Late penalty of 10% per day will be assessed on a homework submission beyond the dead line up to 3 days after which no credit will be given to the assignments.

Online quizzes will be given weekly at the beginning of every lab session. Quizzes will be based on the lecture materials covered during the previous week. There are no make-up quizzes.

Online exams or quizzes offered will require Lockdown Browser in Canvas and are proctored in-class.

Exceptions for missing any component of the final grade (e.g. quizzes, exams) will be considered on a case-by-case basis provided you submit documentation for the reason to the Dean of Students Office who in turn will contact me.

Grading Scale:
Will be posted in Canvas.

Academic honesty policy:

Read the following University Code on Academic Integrity:

"Academic Integrity is the cornerstone of higher education and is central to the ideals of this course and the university. Cheating is strictly prohibited and devalues the degree that you are working on. As a member of the NJIT community, it is your responsibility to protect your educational investment by knowing and following the academic code of integrity policy that is found at: http://www5.njit.edu/policies/sites/policies/files/academic-integrity-code.pdf.

All work that you submit as your own must, in fact be your own. Penalty for the most serious violations of academic integrity may be an F grade.

There should be no collaboration on programming assignments.