Intensive study of the fundamentals of data structures and algorithms. Presents the definitions, representations, processing algorithms for data structures, general design and analysis techniques for algorithms. Covers a broad variety of data structures, algorithms and their applications including linked lists, various tree organizations, hash tables, strings, storage allocation, algorithms for searching and sorting, and a selected collection of other algorithms.

1.1 Contact Information

Instructor:	Alex Gerbessiotis	E-mail:	alexg+cs610@njit.edu
Office:	GITC 4213, 4th floor	Tel:	(973)-596-3244
Office Hours:	Mon 4:00-5:30pm and Thu 4:30-5:30pm	Else, by	appointment Mon/Wed/Thu
Assistant:	TBA on course web-page		
Class Hours:	Thu 6:00-9:05pm, Weston LEC 2		

Course Web Page: http://www.cs.njit.edu/∼alexg/courses/cs610/index.html
Course Web Page [alternate]: https://web.njit.edu/∼alexg/courses/cs610/index.html

Print Handout 1 (the PDF) from Web-page and compare the printout to this document! They must be identical.

1.2 Course Administration

Prerequisites	Per catalog: CS 114 and CS 241 or equivalent. One way to read this: CS 505 and CS 506 (Read Handout 0).
Textbook	Algorithm Design: Foundations, analysis, and internet examples. M. T. Goodrich and R. Tamassia. Wiley, 2001, ISBN 0-471-38365-1. *We abbreviate it in class as GT.*
CourseWork:	**3 exams (including the final)**. Programming project (aka PrP). Potentially one more unannounced quiz (To be determined).
Grading scheme:	1000 points = PrP(134) + Ex1 (200) + Ex2 (333) + Ex3(333).
PrP	A programming project (PrP) with 2 options, each one worth 134 points. A student may submit one or two options; both options must be submitted together (all files in one archive) in ONE email message per Handout 2 **AND RECEIVED BEFORE NOON-TIME** of the date specified in the Calendar. On the sum of the grades of the options submitted lateness points are applied. 30 lateness points deducted at noon-time of due date and every noon-time therafter. A 0-50 grade for the resulting grade is accounted as 0; an over-50 grade is cut-off at 134 points. Over-134 points are excess points.
Practice PS	Five comprehensive problem sets PS1-5 will be periodically posted along with their solutions. Exams may draw from these problem sets.
Exams	Dates in Course Calendar; all exams in classroom. Exams are open-textbook only; you may bring a hardcopy of the textbook but you may not borrow one during the exam. Exam1 (quiz) is on **Sep 28**, 60mins, 200 points. Exam2 (midterm) is on **Oct 19**, 2hours 333 points. Exam3 (final) is on **Dec 21**, 2hours, 333 points. An Exam 2.5 (quiz, 45 minutes) might be given on Nov 16; in that case Exam1's grade gets halved and Exam 2.5 will count for the cut 100 points.
Exam Conflicts	Distance learning (DL) courses might schedule final exams on the day of this class's final exam. If you are a DL student, you should not be taking live courses i.e. this section. If you mix DL and live sections, I will not accommodate DL taking; you got to talk to the DL instructor for exam conflict resolution. Otherwise, avoid taking this section of CS610.

2.1 Course Objectives

1. Learn how and be able to understand and formulate the input-output relationship of computational problems, and formulate the requirements, data and operations of abstract data types (ADT).

2. Learn how and be able to asymptotically compare functions using $o, O, \omega, \Omega, \Theta$, and be able to solve recurrences using the master, iteration/recursion tree, and the substitution methods.

3. Learn how and be able to describe, derive and determine, the asymptotic performance of algorithms for computational problems and operations on elementary and more advanced data structures.

4. Learn how they operate and be able to understand fundamental algorithms and data-structures, and understand their characteristics for problems related to searching, sorting, selection, operations on numbers and polynmials and matrices and graphs. Be able to choose among a variety of similar ones based on problem/program specification and requirements.

5. Learn how and be able to compose more complex algorithms using as building blocks the fundamental algorithms introduced in class.

6. Learn how and be able to compose more complex algorithms using the algorithmic design techniques introduced in class.

7. Learn how and be able to compose advanced data structures using as building blocks the elementary data structures introduced in class.

8. Learn how and be able to implement in a high-level imperative language some of the algorithms and data structures introduced in class in the form of a programming project of considerable complexity.

9. Learn how and be able to understand and possibly identify that some problems are complex and are not susceptible to 'easy' solutions. Learn how and be able to understand the benefits and complexities of using randomness in computation.

2.2 Course Topics

The following describes a tentative list of topics that is intended to be covered in class with indicative chapter pointers to GT. Order of class presentation might change (slightly)!

```
T1 :Ch1,5.2     : Introduction (insertion-sort, fibonacci sequences),
                  Algorithm Analysis (Asymptotic Growth of functions, recurrences)
T2 :Ch1,4.1,5.2 : Algorithm Design Techniques (Incremental, Divide-and-Conquer)
                  Sorting(selection-sort,bubble-sort,merge-sort).
T3 :Ch2.1-2.3   : Elementary Data Structures and Trees. Tree traverals.
T4 :Ch2.4,9.3   : Heaps and Priority Queues. Greedy Algorithms and Huffman coding.
T5 :Ch4         : Quick sort. Complexity of sorting. Sorting in linear time
                  (radix-sort, bucket/count-sort). Selection.
                  Randomization in computation (Randomized QuickSort and QuickSelect).
T6 :Ch6         : Graphs and their representation. Graph traversals (DFS,BFS).
                  Strongly connectivity. Topological sorting.
   Ch4.2          Union-find operations.
T7 :Ch7         : Weighted graph problems. Shortest-path problems (Dijkstra's).
                  All-pairs shortest paths and transitive closure (Floyd-Warshall).
                  Spanning trees (Prim's and Kruskal's algorithms).
T8 :Not in GT   : Graphs and Web-page Ranking: Google's PageRank, Kleinberg's HITS algorithm.
T9 :Ch2.5-2.7   : Hashing.
T10:Ch5         : Integer operations (addition and multiplication). The WORD and BIT models.
                  Straight line programs.
                  Matrix operations (addition and multiplication). Strassen's method.
                  Dynamic Programming and chained matrix multiplication.
T11:Ch3         : Binary Search Trees and Balanced Binary Search trees.
T12:Ch9         : String and Pattern matching algorithms**.
T13:Ch13        : The complexity classes P and NP. NP-completeness.
```

** Time permitting.

3.1 Tentative Course Calendar

Fall 2017				
Week	Thu	Exams	PrP or PS	Comments
W01	09/07		PrP out,PS1*	
W02	09/14			
W03	09/21		PS2*	
W04	09/28	**Exam1**		
W05	10/05		PS3*	
W06	10/12			
W07	10/19	**Exam2**		**aka Midterm**
W08	10/26		PS4*	
W09	11/02			
W10	11/09		PS5*	Mon Nov 06: Drop Date
W11	11/16			
W12	11/21	Class-Day		It is a TUESDAY!!!
W–	11/23			Thanksgiving Day; No class
W13	11/30		PrP in	
W14	12/07			
W–	12/14		No-class	**It's a Reading Day**
W15	12/21	**Exam3**		Final Exam

* Problem Sets (PS) with solutions are not for credit.

Any modifications or deviations from these dates or information, will be done in consultation with the attending students and will be posted on the course Web-page. It is imperative that students check the Course Web-page regularly and frequently.

3.2 Absenteeism

MISSING If you miss a class, it's up to you to make up for lost time. If you miss an exam you MUST CONTACT the Dean of Students (DOS) within 2 working days from the day the reason for the absence is lifted with all necessary documentation. The maximum accommodation period will be the number of missing days to the exam date: it is imperative then that you contact DOS even before the 2 working day period has expired if the accommodation period would be shorter. For Exam1, a DOS approval will get you a scaled Exam2 grade for Exam1. For the unannounced quiz and a DOS approval, Exam1 stays as is. No PrP extensions for any reason, medical or otherwise; you have 3 months to submit it: SUBMIT EARLY.

Grading Written work will be graded for conciseness and correctness. Use formal arguments. Be brief and to the point and write clearly. Material covered in class and appearing in the relevant notes and chapters of the designated textbook can be used without proof. DO NOT USE PENCILS to write down your solutions; If you do use a pencil do not complain about grading after an exam. Read Handout 2 re grading of the PrP.

PrP-grading You are responsible for archiving and making sure dearchiving, compilation, and execution takes place properly in any one of the test platforms (see Handout 2). It is thus imperative to test your archive (and code) in those platforms. If dearchiving, compilation or execution fails, you will not be contacted. Do not expect partial credit without a DETAILED BUG REPORT. See Handout 2 for details.

Grades Check the marks in a written work and report errors promptly. **Resolve any issues WITHIN 2 CALENDAR WEEKS and definitely before the first Reading Day** from the day an exam is returned in class; for PrP or Exam3, within 5 calendar days from the day grades were emailed or posted on Banner respectively. Talk to the grader first and then to the instructor (if different). The final grade is decided on a 0 to 1000 point scale. A 25% or less in the final will get you an F. If you get more than 25% in the final and collect at least 500 points you should expect a C or better. 850-900 points or more are usually needed for an A including robust programming work but this varies and may be lower. If you score less than 500 points the only way to avoid an F is to do well in the cumulative final (say more than half of the points) or have leftover programming points. (All these assuming no violation of the Collaboration policy.)

Incomplete A grade of I(incomplete) is given in rare cases where work cannot be completed during the semester due to documented long-term illness or absence (e.g. unexpected national guard duty). A student needs to be in good standing (i.e. passing the course before the absence) and receives a provisional I if there is no time to makeup for the documented lost time; a letter (or email) with a timeline of what is needed to be done will be sent to the student. Note that for most cases an I would be resolved within few days, not months and not the following semester! Not showing up in the final will probably get you an F rather than an I.

Collaboration **Collaboration of any kind is PROHIBITED in the in-class exams and the PrP. For PrP, a student must turn in code that has fully been written by him/her and no-one else. Any submitted code (even few lines) obtained through the Internet or otherwise, or is product of someone else's work or is common with another student submission, in the same or other section/course, risks severe punishment, as outlined by the University; all parties of such interaction receive automatically 0 and grade is lowered by one or two levels. The work you submit must be the result of your own mental effort and you must safeguard it from other parties; if you can't protect your home computer, use a Lab (AFS) machine.**

Mobile Devices Switch off (not just silence) mobile devices before class. **IF A STUDENT GETS CAUGHT HAVING A PHONE/MOBILE DEVICE (on or off) ON HIM/HER DURING an exam it is graded 0. DEVICES MUST BE OFF AND in A BAG/BACKPACK, NOT ON YOU.**

Email/SPAM Use an NJIT email address or your email might not reach us. Send email to the designated course email address per Handout 0 instructions! ■.

The NJIT Academic Integrity (Honor) Code will be upheld; violations will be reported to the Dean of Students (DOS). Read this handout carefully!