A course on algorithms and data-structures. Advanced topics in data structures and algorithms, involving sequences, sets, and graphs such as searching, sorting, order statistics, balanced search tree operations, hash tables, graph traversals, graph connectivity and path problems. Algebraic and numeric algorithms. Performance measures, analysis techniques, and complexity of such algorithms. Greedy algorithms and dynamic programming-based techniques. String matching algorithms. Introduction to NP-completeness.

1.1 Contact Information

Instructor:	Alex Gerbessiotis	E-mail:	alexg+cs435@njit.edu
Office:	GITC 4213, 4th floor	Tel:	(973)-596-3244
Office Hours:	Tue 4:00-5:30pm and Wed 4:30-5:30pm	Else, by	appointment Mon/Thu
Assistant:	TBA on WEB (Room: See recitation)		
Recitation:	W 1-2pm @ CKB 217; Room might change		
Class Hours:	W 18:00-21:05, CKB 219		

Course Web Page: http://www.cs.njit.edu/~alexg/courses/cs435/index.html

Course Web Page [alternate] : https://web.njit.edu/~alexg/courses/cs435/index.html

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

1.2 Course Administration

Prerequisites	CS 241, CS 288.
Textbook	T.C.Cormen, C.E.Leiserson, R.L.Rivest, and C. Stein. "Introduction to Algorithms", 3rd edition, MIT Press, ISBN-10 : 0262033844 (ISBN-13: 978-0262033848). *We abbreviate it in class as CLRS.*
CourseWork:	**3 exams (including the final)**; **MiniProject (aka MP)** ; **Homeworks (HW)**.
Grading scheme:	1000 points = MP(160) + Ex1(100) + Ex2(345) + Ex3(345) + HW(50)
MP	A programming mini-project (MP) with 3 options each one worth 120 points. A student may submit one or two options; all files of one or two options in ONE archive per Handout 2 **AND RECEIVED in MOODLE 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 thereafter. A 0-60 grade for the resulting grade is accounted as 0; an over-60 grade is cut-off at 160 points. Over-160 points are excess points.
HW,PS	Approximately eight problem sets PS1-8 will be periodically posted along with their solutions. 5 practice Homeworks (HW1-HW5) with one or two problems will be given out (submission by moodle). Exams 1 and 2 may draw from these.
Exams	Dates in Course Calendar; all exams in classroom unless otherwise noted. Exam1 is closed-everything; if snow day it becomes Exam 2.5. Exam2 and Exam3 are open-textbook only; you may bring a hard-copy of the textbook but you are not allowed to borrow one during the exam. For the final, you may also bring in class a clean copy of Handout 4 on red-black trees in addition to the textbook. Exam1 is on **Wed Feb 07**, 45min, 100 points. Exam2 is on **Wed Feb 28**, 120min, 345 points. Exam3 is on **Wed May 09**, 120min, 345 points.
Exam Conflicts	This is a high-numbered required course. **In case of multiple exams on a same day, this exam has priority even if it is the last exam of the day**.

2.1 Course Objectives

A1. Learn how and be able to asymptotically compare functions using $o, O, \omega, \Omega, \Theta$, and be able to solve recurrences using the master, the iteration/recursion tree, and the substitution method.

A2. Learn how and be able to describe the asymptotic performance of algorithms and data structure operations.

A3. Learn how and be able to understand fundamental algorithms and data structures and be able to trace their operations for problems such as sorting, searching, selection, operations on numbers, polynomials and matrices, and graphs.

B1. Learn how and be able to understand the input and output specifications of a problem, the relationship between input and output and identify, define, and describe the algorithmic requirements that formulate a solution for those problems.

C1. Learn and be able to identify the performance characteristics of algorithms and data structures for problems such as sorting, searching, selection, operations on numbers, polynomials and matrices, and graphs; be able to select among multiple available solutions to meet desired needs.

C2. Learn how fundamental algorithm design techniques work and be able to understand how to use them to design, implement and evaluate a variety of algorithmic problems.

F1. Learn how and be able to describe in writing algorithm operations and reason about their behavior and performance succinctly.

I1. Learn how and be able to effectively apply the knowledge gained in the course in dealing and interacting with software libraries that offer same or alternative implementations of algorithms and data structures.

J1. Learn how and be able to model, design, and construct algorithms and data structure operations and analyze and derive their asymptotic performance and provide tradeoff solutions (eg. space vs time).

J2. Learn how and be able to understand the operations of fundamental algorithms and data-structures, their characteristics, and be able to choose among a variety of similar ones based on problem/program specification and requirements in building more complex algorithms and data structures, and deciding trade-offs between space and time requirements.

K1. Learn how and be able to compose more complex algorithms and data structures using as building blocks the fundamental algorithms and data structures introduced in class.

K2. Learn how and be able to compose more complex algorithms using the algorithmic design techniques introduced in class.

2.2 Course Topics

```
T1 : AL1(2)/AL2(1)/AL3(1): Introduction, Algorithm Design Techniques (Incremental, Divide-and-Conquer)
T2 : AL1(2)/AL2(1)   : Sorting Algorithms (Insertion, Selection, BubbleSort, MergeSort) Asymptotic growth of functions
T3 : AL1(2),DS3(1)   : Recurrences
T4 : AL3(1),DS5(1)   : Brief Review on elementary data structures (Stacks,Queues,Trees,Lists)
T5 : AL2(2),AL7(1)   : HeapSort,PriorityQueues,Huffman Coding,and QuickSort(Worst-case and Average-case analysis)
T6 : AL3(2)          : Distribution-based sorting(Count/Radix/Bucket-Sort).Lower bounds on comparison-based sorting.
T7 : AL3(2)          : Selection. Selection in Linear Time.
T8 : AL3(4),AL7(3)   : Hashing, Balanced Binary Search Trees ( Red-Black Trees).
T9 : AL2(2),AL3(1)   : Dynamic Programming and Chained Matrix Multiplication,Arithmetic problems
T10: AL3(2),AL7(1)   : Union Find Algorithms; Introduction to Graph Algorithms
T11: AL3(3)AL7(2)    : Depth First Search,Breadth First Search,Minimum Spanning Trees.
T12: AL3(3)          : Shortest path Algorithms (Dijkstra and Floyd-Warshall)
T13: AL4(2)          : NP-completeness.
```

Section 2.2 of the previous page contains a tentative list of topics that is intended to be covered in class. The code Ti refers to a topic. A topic may spread over one or more lectures. The code ALi refers to the ACM Computing Curricula 2013 topic description code. In parentheses, we provide an approximate number of hours per topic. Hour coverage may change depending on circumstances (eg. class pace, weather). Minimum time requirements of the topics covered are. AL1 (basic analysis):4, AL2 (algorithmic strategies): 6, AL3 (fundamental data structures and algorithms):12, AL4 (basic automata, computability and complexity) optional:2, AL7 (advanced data structures, algorithms and analysis): elective, DS3 (proof techniques):1, DS5(graphs and trees):1.

3.1 Tentative Course Calendar

Spring 2018			
Week	Wed	PS or HW or Exam	Comments
W1	01/17	PS1* HW1	MP out; PS out
W2	01/24	PS2* HW2	HW1 in before noon
W3	01/31	PS3*	HW2 in before noon
W4	02/07	**Exam1**	
W5	02/14	PS4* HW3	
W6	02/21	PS5*	HW3 in before noon
W7	02/28	**Exam2**	
W8	03/07		
W-	03/14		Spring Break
W9	03/21	PS6* HW4	
W10	03/28	Exam 2.5	HW4 in before noon
W11	04/04	PS7* HW5	MON APR 2 : Withdrawal Deadline
W12	04/11	PS8*	HW5 in before noon
W13	04/18		MP is due BEFORE NOON
W14	04/25		
W-	05/02		READING DAY; NO CLASS
W15	05/09	**Exam3**	Final Exam is Exam3

* 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 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. No MP 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.
MP-Grading	See Handout 2 for details (section Testing and Grading).
Grades	Check the marks in 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 or homework graded in moodle; for MP 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 based on a 0 to 1000 point scale. A 25% or less in the final exam 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 points or more are usually needed for an A but this threshold can vary (be lower). If you score less than 500 points the only way to avoid a D or F is to do well in the cumulative final (say more of half of the points) or have excess 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 MP. For MP, 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!