Credits: 3

Description: In this course, students will survey fundamental data structures and many associated algorithms. Emphasis will be placed on matching the appropriate data structures and algorithms to application problems. Analysis of algorithms is crucial to making proper selections, so analysis is important in the course. This course assumes that students are already familiar with advanced programming techniques including the denition of classes, and use of dynamic memory and linked data structures, including lists and trees. Even though the treatment of algorithms and data structures is mostly conceptual, students are expected to be able to transform these algorithms and data structures into programs with proper approaches of software module development.

Time and Location: Tue& Thu 3pm-4:20pm, 2400 EB

Instructor: Hu Ding, huding@msu.edu, office 2140 EB, office hours 1:30-3:30pm Mon

TA: Manni Liu, liumanni@msu.edu, office 3100 EB, office hours 3-5pm Wed

Prerequisites: Knowledge comparable to that taught in:

1. CSE 260 Discrete Structures in Computer Science
2. CSE 232 Introduction to Programming II


Also recommend to read Introduction to Algorithms 3rd edition, Cormen, Leiserson, Rivest and Stein, MIT Press.

Class notes: The class notes will be posted on the course web site.

Content:

- Asymptotic analysis (chapter 2).
- Sorting: insertion sort; merge sort (divide and conquer) (chapter 7).
- Heap: heap sort (chapter 6).
- Tree: binary search tree; AVL tree (chapter 4).
- Hashing (chapter 5).
- Graph: DFS; BFS; topological sort (chapter 9).
• Spanning trees (chapter 9).
• Greedy algorithms (chapter 10).
• Dynamic programming (chapter 10).
• Shortest path algorithms (chapter 9).
• Complexity.
• Approximation algorithms.

Graded work:
• homework 56% = 7 × 8% (all the homeworks are paperwork including algorithms design in pseudo code, analyses, and math proofs. Several optional coding tasks may be released but not required; the purpose is to help students to understand the algorithms better and get more coding exercise).
• midterm exam 20%
• final exam 20%
• class participation 4%

Grading: The final grades will be assigned based on the following scale:
1. ≥ 90%: 4.0
2. 85%: 3.5
3. 75%: 3.0
4. 70%: 2.5
5. 60%: 2.0

The instructor reserves the right to make changes to the grading scale. Specifically, the score required to obtain each mark may be lowered.

Midterm Exam: Thursday, Oct 19 2017, 3:00pm - 4:20pm in 2400 Engineering Building.

Final Exam: Wednesday, Dec 13 2017, 10:00am - 12:00pm in 2400 Engineering Building.

Homework: Homework assignments typically will have multiple problems and may require mathematical analysis, analysis of an existing programs or functions.

Re-grading policy: All requests for re-grading must be submitted as a written document detailing the request. The written request must be received by the course instructor no later than one week after the graded item was returned to the student. Since graded items will typically be returned during class, students are advised to pick up their graded items in a timely manner. Any requests for re-grading that do not follow these guidelines will not be considered.

Homework deadlines: All homework assignments are due at the start of class on the deadline date specified on the assignment handout. A homework assignment submitted after the start of class but before the end of class will be accepted with a late penalty of 10% off of the final grade. Homework will not be accepted after the end of class on the deadline date. In case of a documented crisis, such as illness,
the student should submit an official document to arrange for alternate grading. Advance notification is required for late submission unless this is impossible.

**Exam schedule:** Documented university-related or health-related issues that prevent a student from taking an exam at the regularly scheduled time will be reviewed for accommodation. For conflicts that are known in advance, students are required to submit written documentation at least three business days before the exam is offered. Any emergency issue for which written documentation cannot be given in advance must be supported by appropriate documentation. For example, a serious medical emergency must be supported by documentation from a physician. The course instructor reserves the right to confirm any written documentation via verification with third parties.

**Academic Integrity:** As scholars and scientists, academic integrity is of the utmost importance. CSE 331 will adhere to the Michigan State University policies of academic integrity as set forth in the General Student Regulations, the All-University Policy on Integrity of Scholarship and Grades, and the Department of Computer Science and Engineering Graduate Student Handbook. Students violating the policies and regulations regarding academic dishonesty will be penalized accordingly. Furthermore, additional penalties may be imposed at the discretion of the instructor.

**Students with disabilities:** The course instructor is committed to accommodating students with disabilities according to the policy of the Michigan State University Resource Center for Personals with Disabilities (RCPD), stated as follows (reproduced from the RCPD Model Statement): “Michigan State University is committed to providing equal opportunity for participation in all programs, services and activities. Requests for accommodations by persons with disabilities may be made by contacting the Resource Center for Persons with Disabilities at 517-884-RCPD or on the web at rcpd.msu.edu. Once your eligibility for an accommodation has been determined, you will be issued a verified individual services accommodation (“VISA”) form. Please present this form to me at the start of the term and/or two weeks prior to the accommodation date (test, project, etc). Requests received after this date will be honored whenever possible.”

**Acknowledgments:** Some course materials have been adapted from others including Yanni Sun, Jeremy Buhler, George Stockman, John Weng, Li Xiao, Laura Dillon, and Kevin Liu.