# Course Syllabus

# VE281 Data Structures and Algorithms

# Fall 2016

## Course Description:

Introduction to algorithm analysis and big-Oh notation; Fundamental data structures including priority queues, hash tables, binary trees, binary search trees, balanced trees, and graphs; Searching and sorting algorithms; Basic graph algorithms; Introduction to dynamic programming and branch-and-bound techniques.

## Instructor:

Weikang Qian
Email: qianwk@sjtu.edu.cn
Phone: 34204020
Office: Room 421, JI Building
Office hour: Monday 1:00 pm – 2:00 pm and Wednesday 1:00 pm – 2:00 pm, or by appointment

## Textbook (Recommended but not required):

1. *Data Structures and Algorithm Analysis*, by Clifford Shaffer.
Online available: http://people.cs.vt.edu/~shaffer/Book/C++3e20120605.pdf

2. *Data Structures and Algorithms with Object-Oriented Design Patterns in C++*, by Bruno Preiss.
Online available: http://www.brpreiss.com/books/opus4/html/book.html.

3. *Introduction to Algorithms, 3rd edition*, by Thomas Cormen, Charles Leiserson, Ronald Rivest, and Clifford Stein, MIT Press, 2009.

## Class Webpage:

Log into Canvas at https://sjtu-umich.instructure.com/courses/146. Announcements, lecture slides, assignments, and grades will be posted on the class webpage.

## Course Prerequisites:

Ve280 Programming and Elementary Data Structures and Ve203 Discrete Mathematics.

## Grading Policy:

There will be 4~5 written assignments, 4~5 programming assignments, one midterm exam, and one final exam. The grading distribution is:

In-class quizzes: 4%

Written assignments: 16%

Programming assignments: 30%

Midterm Exam: 20%

Final Exam: 30%

Any questions about the grading of the projects or exams must be brought to the attention of your TAs or the instructor within one week after the item is returned.

## Exam

The exams will be closed book ones. No electronic devices are allowed in the exams.

You are expected to take both exams at the scheduled times. If you miss an exam, and a medical or personal emergency is not involved, you will receive a zero for that exam. If you anticipate an exam in another course, you must notify the instructor at least one week before the exam date.

## Academic Integrity:

1. All students are expected to attend all of the lectures.
2. All programming assignments must be done by yourself independently. You may discuss the project in oral with other student. However, you may not read/copy others' solution and you may not use test cases from others. In all cases in which we have reason to believe that cheating has occurred, we will report your case to the Honor Council for evaluation.
3. Exams will be given under the JI's Honor Code and will require individual efforts.

## Teaching Schedule (Subject to Change)

| Lecture | Date | Teaching Activities (Topics and Exams) |
|---|---|---|
| 1 | Sep. 12 | Course Introduction; Asymptotic Algorithm Analysis |
| 2 | Sep. 14 | Asymptotic Algorithm Analysis |
| 3 | Sep. 19 | Analyze Program; Basic Sorting; Merge Sort |
| 4 | Sep. 21 | Quick Sort; Comparison Sort Summary |
| 5 | Sep. 23 | Non-comparison Sort |
| 6 | Sep. 26 | Linear-time Selection |
| 7 | Sep. 28 | Hashing |
| 8 | Oct. 10 | Hashing: Open Addressing |
| 9 | Oct. 12 | Universal Hashing |
| 10 | Oct. 17 | Rehashing; Bloom Filters |
| 11 | Oct. 19 | Binary Trees; Binary Tree Traversal |
| 12 | Oct. 21 | Priority Queues; Heaps |
| 13 | Oct. 24 | Heaps; Binary Search Trees |
| 14 | Oct. 26 | Binary Search Tree Time Complexity |
| | Oct. 31 | Midterm |
| 15 | Nov. 2 | Binary Search Tree Other Useful Operations; k-d Trees |
| 16 | Nov. 4 | Tries; AVL Trees |
| 17 | Nov. 7 | AVL Trees |
| 18 | Nov. 9 | Red-black Trees |
| 19 | Nov. 14 | Red-black Trees; Graphs |
| 20 | Nov. 16 | Graph Representation; Graph Search; Topological Sorting |
| 21 | Nov. 18 | Shortest Path |
| 22 | Nov. 21 | Minimum Spanning Trees |
| 23 | Nov. 23 | Dynamic Programming: Matrix-Chain Multiplication |
| 24 | Nov. 28 | Dynamic Programming: Longest Common Subsequence |
| 25 | Nov. 30 | Dynamic Programming: Knapsack Problem |
| 26 | Dec. 2 | Bellman-Ford Algorithm; All Pairs Short Paths |
| 27 | Dec. 5 | Union-Find Data Structure |
| 28 | Dec. 7 | Branch-and-bound Algorithm |