CS 4040/5040: Design and Analysis of Algorithms  
Fall 2014

Class Meetings: Mon, Wed, Fri 12:55–1:50pm, ARC 106

Instructor: Razvan Bunescu  
Office: Stocker 341  
Office Hours: Wed, Fri 2:00–3:00pm, or by email appointment  
Email: bunescu@ohio.edu

Grader: Mika Chen  
Office: Stocker 371  
Office Hours: Tue, Thu 10:00–11:30am, or by email appointment  
Email: mc277509@ohio.edu

Class Website: http://ace.cs.ohio.edu/~razvan/courses/cs4040

Prerequisites: CS 3610 (Data Structures) and basic mathematical dexterity.

Textbook:  
Introduction to Algorithms by T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein,  

Course Description:  
This course provides an introduction to the modern study of computer algorithms.  
Through this course students should be able to:  
1) Analyze algorithm performance using complexity measurement.  
2) Master major algorithms design techniques such as divide and conquer, greedy and  
dynamic programming.  
3) Apply the above approaches to solve a variety of practical problems such as sorting  
and selection, graph problems, and other optimization problems.  
4) Understand the theory of NP-completeness.

Course Outcomes:  
A: Ability to apply knowledge of Computing and Mathematics appropriate to the discipline.  
Students will be able to:  
1. Use mathematical induction to prove asymptotic bounds for time complexity.  
2. Prove the correctness of algorithms using loop invariants or more general types of  
proofs.  
3. Use asymptotic notation to formulate the time and space requirements of algorithms.  
4. Prove the tight asymptotic lower bound for the running time of any comparison-based  
sorting algorithm.  
5. Prove that a problem is P, NP, or NP-Complete.
B: Ability to analyze a problem, and identify and define the computing requirements appropriate to its solution. Students will be able to:

1. Use the Master Theorem to analyze the asymptotic time complexity of divide and conquer algorithms.
2. Use the theory of NP-completeness to determine whether a computational problem can be solved efficiently.

C: Ability to design, implement, and evaluate a computer-based system, process, component or program to meet desired needs. Students will be able to:

1. Design, implement, and test an efficient algorithmic solution for a given computational problem.

I: Ability to use current techniques, skills, and tools necessary for computing practices. Students will be able to:

1. Apply the divide-and-conquer, greedy, and dynamic programming techniques to the design and analysis of algorithms.

J: Ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer-based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices.

1. Comparatively evaluate sorting algorithms.
2. Apply algorithmic principles to determine whether a given set of requirements for a computational problem can be met.
3. Compare the implementation choices of specific data types, such as priority queues or graphs, and justify which is the most appropriate one for a given application.
4. Produce an algorithmic approach that meets a given set of requirements for a computer-based system.

Grading:
24%: 6 HW Assignments
8%: Project
8%: 2 Quizzes
30%: 2 Midterm Exams
30%: Final Exam

Grading Scale:
A (> 92%) A− (> 90%) B+ (> 87%) B (> 83%) B− (> 80%)
C+ (> 77%) C (> 73%) C− (> 70%) D+ (> 67%) D (> 63%) D− (> 60%)

Exam Dates:
Midterm 1: Monday, Oct. 6, in class
Midterm 2: Monday, Nov. 10, in class
Final: Friday, Dec. 12, 10:10am – 12:10pm

Other Important Dates:
Monday, Sep. 1: Labor Day, no class.
Friday, Oct. 3: Reading Day, no class.
Friday, Oct. 31: Last day to drop class.
Friday, Nov. 7: EECS faculty retreat, no class.
Wednesday, Nov. 26: Thanksgiving break, no class.
Friday, Nov. 28: Thanksgiving break, no class.
Friday, Dec. 5: Last day of this class.

Course and Attendance policies:
Assignments: All homework assignments are due before class. **No late submissions will be accepted.**
Attendance: It is in your best interest to attend all the lectures. Some of the material will not be found in the textbook or on the slides. Extra credit will be awarded for class activity. Also, be sure to check your OU email on a regular basis for important announcements.

Academic Dishonesty Policy:
All work must be the student’s own. All external references used in reports must be properly cited. No credit will be given for duplicate or plagiarized work. Additional measures may be imposed by the Office of Community Standards and Student Responsibility, when conditions warrant. Students may appeal academic sanctions through the grade appeal process. The OU Student Code of Conduct Policy is available online at:
http://www.ohio.edu/communitystandards/academic/students.cfm

Disability-based Accommodation:
Any student who suspects s/he may need an accommodation based on the impact of a disability should contact the class instructor privately to discuss the students specific needs and provide written documentation from the Office of Student Accessibility Services. If the student is not yet registered as a student with a disability, s/he should contact the Office of Student Accessibility Services.

Other Policies:
Be sure to notify the professor of any exam conflicts or other extenuating circumstances well in advance. No missed exams will be made up without prior approval.