

Wright State University

CORE Scholar

Computer Science & Engineering Syllabi

College of Engineering & Computer Science

Winter 2011

CS 400/600-02: Data Structures and Algorithms

Sarah Gothard

Wright State University - Main Campus

Follow this and additional works at: https://corescholar.libraries.wright.edu/cecs_syllabi



Part of the [Computer Engineering Commons](#), and the [Computer Sciences Commons](#)

Repository Citation

Gothard, S. (2011). CS 400/600-02: Data Structures and Algorithms. .

https://corescholar.libraries.wright.edu/cecs_syllabi/818

This Syllabus is brought to you for free and open access by the College of Engineering & Computer Science at CORE Scholar. It has been accepted for inclusion in Computer Science & Engineering Syllabi by an authorized administrator of CORE Scholar. For more information, please contact library-corescholar@wright.edu.

CS 400/600 Section 2 Syllabus

Course Information

- Term: Winter 2011
- Title: Data Structures and Algorithms
- Description: Study of the implementation of data structures and control structures in professional computer programs. Introduction to the fundamentals of complexity and analysis. Study of common standard problems and solutions (e.g., transitive closure and critical path). Emphasis on high-level language software design.
- Location: Russ Engineer Center 145
- Meetings times: MWF 1:30 p.m. - 2:20 p.m.
- Prerequisites: Undergraduate level CS 242 Minimum Grade of D and Undergraduate level CEG 233 Minimum Grade of D.

Instructor Information

- Name: Dr. Sarah Gothard
- Email: sarah.gothard@wright.edu
- Office location: Russ Engineering Center 437
- Office hours:
 - MWF 8:30 a.m. - 10:30 a.m.
 - MWF 12:15 p.m. - 1:15 p.m.
 - F 9:30 a.m. - 10:30 a.m.
 - F 2:30 p.m. - 4:00 p.m.
- 937-321-5167 (during office hours or emergencies only)
- Biography:

Dr. Gothard received her Ph.D. in Computer Science from Clemson University in 2007.

Dr. Gothard taught the following lecture courses at Clemson University from 2002 to 2008: Computer Science I and II, Algorithms and Data Structures, and Tools and Techniques for Software Development. Courses were taught with various formats as part of her research, including the use of computer graphics in C, C++, and Java.

Dr. Gothard began teaching at WSU Spring 2009 and has taught CS 240 (Computer Programming I), CS 241 (Computer Programming II), CS 242 (Computer Programming III), CS 400/600 (Data Structures and Algorithms), CEG 233 (Linux and Windows), and CEG 433/633 (Operating Systems).

Dr. Gothard was awarded the 2009-2010 College of Engineering and Computer Science Excellence in Teaching Award for Adjunct faculty.

Textbook

Data Structures and Algorithm Analysis in C++ (3rd ed.) - or any similar data structures book, Mark Allen Weiss, Addison-Wesley, 2007, 0-321-44146-X

Course Requirements

There will be four (4) programming assignments, a midterm exam, a final exam, and a couple homework assignments.

Language

C++ in Unix

Points

- Programming assignments: 40%
- Homework: 10%
- Midterm Exam: 20%
- Final Exam: 30%

Grading

A: 90-100
B: 80-89
C: 70-79
D: 60-69
F: 0-59

Policies

Deadlines

- Work is due at the specified deadline. Late work will not be accepted. If submitting the assignment appears to fail, email a copy of the assignment to me before the deadline.
- If a project is only partially completed, you should submit what you have before the deadline.

Missing Grades

- If something is wrong with your grade on Pilot, it is your responsibility to notify the instructor within a couple weeks of posting.
- If you miss an exam, inform the instructor before the solution is covered in the next class. While notification may not prevent you from getting a zero (depending on the circumstances), waiting until after the solution has been covered guarantees a zero on the exam.

Instructor Late

If the instructor is late for class, students are expected to wait for 15 minutes after the class period starts before leaving.

Academic Integrity

General

- Be honest at all times.
- Act fairly towards others. For example, do not seek an unfair advantage over others by cheating with or by looking at other individual's work during examinations or laboratory assignments.

- Passing off other people's work as your own is unethical in any setting. In an academic setting, it is a breach of the university's policies.
- *All* cases of plagiarism, cheating, or academic dishonesty will be reported to the Community Standards and Student Conduct Office. Penalties will be handled on a case by case basis, ranging from a zero on the assignment for all involved students to a failing grade in the course for all involved students.
- Those who are complicit in academic misconduct will receive the same penalties as the primary offenders.

What Is Allowed

- Students are allowed to discuss the general requirements of assignments to make certain that they understand the problem and its goal.
- A student is allowed to ask another student (who has submitted the assignment) for help with a syntax error or other minor problem that does not require extensive exploration of the solution. (Students are never to show other students their solutions until after the submission deadline is passed.)
- Students may receive direction from their TA's, the CS help room, the textbook, the provided materials, and the instructor without documenting that the help took place.
- Any outside help should be documented in the comments for the assignment. This will allow the instructor to comment on and correct the degree of collaboration if necessary. Unacknowledged collaboration will be considered a violation of course policy.

What Is Not Allowed

- Students who have not submitted their assignments may not look at other students' programs.
- Students who have looked at other students' code may not re-submit unless they first consult the instructor.
- Students may not look at any other solutions to any assignments that can still be turned in.
- Students may not use code from other classes or other quarters without first consulting the instructor.

Special Team Rules

- If a project is to be done as a team, **you may not discuss the project with anyone else in the class other than your project partners.**