

Wright State University

CORE Scholar

Computer Science & Engineering Syllabi

College of Engineering & Computer Science

Summer 2007

CS 141: Computer Programming I

Eric Maston

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

Maston, E. (2007). CS 141: Computer Programming I. .
https://corescholar.libraries.wright.edu/cecs_syllabi/661

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.

Computer Science 141 - Computer Programming I

**Summer 2007
Wright State University**

Course Description

This course provides a general introduction to the fundamentals of computer programming. Examples from and applications to a broad range of problems are given. No prior knowledge of programming is assumed. The concepts covered will be applied to the Java programming language.

Goals

There are several goals to accomplish in CS 141:

1. Learning basic programming skills.
2. Conceptualize how computer programs are logically developed.
3. Develop an appreciation for systems and programming.
4. Learn how to solve real, complex problems
5. Have some fun!

Lecturer

Eric Matson

Office: 336 Russ Engineering Center

Phone: 937-775-5108

Office Hours: Tuesday/Thursday 3:00 – 4:10 Russ 336 or by appt.

Email: eric.matson@wright.edu

Class: Tuesday/Thursday 12:20 – 1:35 Russ Engineering Center 150

Text

Starting Out With Java (3rd Ed.). Tony Gaddis, ISBN: 0-321-47927-0.

Lab

You must be enrolled in one of the lab sections. Lab is a crucial element of the course. This is a “hands on” technical class, so lab will provide lots of practice. Take advantage of lab time to enhance your programming skills.

Prerequisites

There are no official prerequisites, but knowledge of algebra is useful.

Grading

Programming Lab Assignments 50%

Midterm Exam & Quizzes 20%

Final Exam 30%

The base scale is: A: 90-100, B: 80-89, C: 70-79, D: 60-69, F: 0-59. This is the highest requirement that will be used. The scales may be lowered or revised if necessary, **unless you get less than 70% of the possible points on your programming lab assignments in which case you fail the entire course regardless of your overall course average.**

Policies and Notes

- Attendance: Attendance is not required, nor will it be taken after the first couple of lectures. If you are not a regular attendee, it will be your responsibility to seek out what material was covered in the lecture and learn it. Most of my exam questions will be taken directly from ideas covered during the lecture, so it greatly helps if you attend!
- I will utilize WebCT to post updates to the course, sample code, projects, announcements, schedule, etc. Get in the habit of checking it regularly.
- Always make back ups of all of you work. Never have just one copy of anything!
- If you are going to miss an exam, for any reason, discuss it with me in advance. If it is an emergency situation, please notify me as soon as possible.
- You can reach me a number of ways. Email is normally the best as I check it about 18 hours a day normally. You can also reach me by phone during the day at 775-5108. If you need human contact either stop in during my office hours, make an appointment, or just come by my office. If I am in and not on a deadline to get something else completed, I will normally try to help as much as possible.
- There are technologies we will use in this class that you may not already know, such as file transfer, command line, text editors, file systems, etc. We will cover some of these technologies as we go.
- The key to learning in this class will be spending time working through the problems. Don't wait until 2 hours before something is due to try to learn the concept and then write the program. This normally ends in a disaster! Stay up with the readings and try to work through some of the examples in the book. I will post what I call, "10 minute programs" which are exercises that you can work through to learn key concepts. And yes, they are programs you can write and execute in 10 minutes (unless you are a really slow typist, like me. In that case, they become "20 minute programs".)

Academic Misconduct

In this class, the only way to truly learn the concepts to is do the work yourself. I encourage working with other people on the course concepts. When you begin to write the program, complete and submit your own work.

Work that has obviously been copied or in the more extreme case, when the original author's name has not even been changed, both parties will receive a 0 grade for that assignment. Both parties will also be turned over to the Office of Judicial Affairs.

Schedule (always subject to changes)

#	Day	Date	Topic	Reading
1	T	June 12	Introduction, Algorithms, Languages, Computers, Number Systems, Web and Java	Chapter 1
2	U	June 14	Your First Java Program	2.1 – 2.2
3	T	June 19	Data Types, Operations, the Math Class	2.3 – 2.8
4	U	June 21	Input, Output Methods, Errors and Debugging, Strings	2.9 – 2.15
5	T	June 26	Decisions	3.1 – 3.7
6	U	June 28	Decisions	3.9 -3.12
7	T	July 3	Loops	4.1 – 4.4
8	U	July 5	Loops	4.5 – 4.9
9	T	July 10	Program Style, Documentation and Guidelines, Review	
10	U	July 12	Midterm Exam	
11	T	July 17	Arrays	8.1 -8.2
12	U	July 19	Arrays, Sorting	
13	T	July 24	Methods	5.1
14	U	July 26	Methods	5.2 – 5.6
15	T	July 31	Objects and Classes	6.1
16	U	Aug 2	Objects and Classes	6.2 – 6.4
17	T	Aug 7	Objects and Classes	6.5 -6.6
18	U	Aug 9	Strings	
19	T	Aug 14	Strings Concluded, Command-Line Arguments, Review	
20	U	Aug 16	Final Exam	

Always have readings scheduled for that day complete prior to the class meeting

Note: T = Tuesday
U = Thursday