

Wright State University

CORE Scholar

Computer Science & Engineering Syllabi

College of Engineering & Computer Science

Spring 2008

CS 241-02: Computer Programming II

Michael R. Peterson

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

Peterson, M. R. (2008). CS 241-02: Computer Programming II. .

https://corescholar.libraries.wright.edu/cecs_syllabi/574

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 Programming II CS 241-02 – Spring 2008

Instructor: Michael R. Peterson
Office: 390 Joshi
Office Hours: 7:30 – 8:30 T Th (Other times via email appointment)
Email: (preferred contact) peterson.7@wright.edu
Office Phone: (937)775-5173

Room & Time: Section 02: 6:05 – 7:20 PM, A330 Creative Arts

Course Description: A continuation of CS240. The emphasis is on data abstraction and object-oriented programming. Prerequisite: CS240.

Textbook (Required): Gaddis, Tony (2008). "Starting out with JAVA, 3e", Addison Wesley, ISBN 978-0-321-47927-3.

It is neither possible, nor desirable, to discuss every nuance of the material covered in this course during our limited class time. Students should be aware that although we will discuss the most important materials in class, the textbook contains important facts that may not be discussed in class. Students should not only be able to discuss course concepts in detail, but they should also be able to demonstrate their mastery by applying these concepts on examinations to related problems with which they have no previous experience.

Grading: A student's demonstration of their ability to discuss issues, solve problems, and demonstrate mastery of programming and introductory computer science will be the underlying metric for the determination of a student's overall grade in this course. Students will be provided the opportunity to demonstrate their mastery through examinations, weekly laboratory assignments, and several programming projects. The overall course grade will be determined as follows:

Programming Projects: 400 Points [4 @ 100 pts]

Laboratory Assignments: 160 Points [8 @ 20 pts]

Midterm Examination: 200 Points

Final Examination: 300 Points

Total: 1060 Points

Grades will be assigned on a standard A/90%, B/80%, C/70%, D/60%, F/60%- scale. Clustering of grades may cause the thresholds to be lowered; they will not be raised. *The instructor reserves the right to fail any student who does not attain an overall passing grade (70%+) in the programming projects.*

Programming Projects and Laboratory Assignments: The instructor will provide a number of opportunities for students to develop their mastery of the subject throughout the course through graded assignments. Laboratory assignments are subject to changes specified by the TA during the laboratory period. All students are required to attend their scheduled laboratory each week. *Take your textbook to the laboratory sessions.* Lab assignments will be provided at the lab session, by the TA. Assignments must compile to receive credit. *Programs that do not compile will not be graded and will receive a grade of 0.* All programs must have comments at the top that identify the student, the course, and the project type/number. Points will be deducted for projects submitted late. The deduction will be 10% of the total possible points per 24 hours (or portion thereof) elapsed from the moment that the project was due. No points will be awarded for projects that are more than one week late. Begin your projects immediately to guarantee that you have time to get help if necessary and complete them on-time. Deadlines will only be extended for documented emergencies. Poor time management, corrupt files, or network outages will not be considered a sufficient excuse to extend this deadline. Important note: Murphy's Law dictates that computers go down, networks fail, and data gets destroyed on the day that a project is due. Plan ahead. Back up your work. Start early!

Examinations: Examinations will occur at the normally scheduled class time and location unless announced otherwise in class. The final examination is cumulative and will take place during the university scheduled time period in the normally scheduled class location unless announced otherwise in class. Students may use one (two-sided) 8.5"x11" page of *hand-written* notes on the examinations.

Academic Integrity: Student-teacher relationships are built on trust. For example, students must trust that teachers have made appropriate decisions about the structure and content of the courses that they teach, and teachers must trust that the assignments which students turn in are their own. Acts which undermine this trust undermine the educational process. It is the policy of Wright State University to uphold and support standards of personal honesty and integrity for all students consistent with the goals of a community of scholars and students seeking knowledge and truth. Furthermore, it is the policy of the university to enforce these standards. The following recommendations are made for students:

- 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.
- Take group as well as individual responsibility for honorable behavior. Collectively, as well as individually, make every effort to prevent and avoid academic misconduct, and reports acts of misconduct that you witness.
- Know the policy -- ignorance is no defense. Read the policy contained in the student handbook. If you have any questions regarding academic misconduct, contact your instructor.

Students are encouraged to get together in small study groups to discuss the course topics and ungraded homework problems. However, **students must work on all graded course assignments and examinations on an individual basis.**

What IS allowed: Students are allowed to discuss the general requirements of assignment to make certain that they understand the problem and its goal. Students are allowed to ask another student (who has completed the assignment) for (brief) help with a syntax error or other minor problem that does not require extensive exploration of the solution. If another student asks you for help debugging AFTER you have finished the assignment, then you may help them briefly, but you may NOT show them your solution. Students may go to their TA, the CS help room, or the instructor for more detailed help. If you work with other student in an allowed manner, you are required to acknowledge the collaboration and its extent in the assignment. This will allow the instructor to comment on and correct the degree of collaboration if necessary. Unacknowledged collaboration will be considered dishonest.

What IS NOT allowed: Students may NOT work together on assignments. Students may NOT use code created by other students. You may NOT look at code created by another student (even to debug) until after you have completed the assignment yourself. Students absolutely may NOT turn in someone else's solution with simple cosmetic changes (say, changed variable names) to the solution -- this is a gross break of academic integrity and will result in a failing grade for the course. *You are responsible for ensuring that other students do not have access to your work* - do not give another student access to your files, do not leave printouts in the recycling bin or printer, do not leave your workstation unattended, etc. If you suspect that your work has been compromised notify your instructor immediately.

Conduct for Examinations: The academic code demands that no student should have an unfair advantage over any other student during examinations. Thus, it is strictly forbidden for any student to refer to information from previous offerings of this course unless this information is provided by the instructor to all students fairly. Thus, the use of test banks of previous quizzes or asking questions about examinations or laboratory assignments to prior students is strictly forbidden.

Absences: Class attendance will not be a direct factor in your grade but will strongly affect the quality of your education. Students who miss class are responsible for the material or announcements presented. Any extenuating circumstances which impact on your participation in the course should be discussed with your instructor as soon as those circumstances are known. Make-ups for examinations may be arranged if a student's absence is caused by documented illness or personal emergency. It is the student's responsibility to provide a written explanation (including supporting evidence) to the instructor in a timely manner. Students registering after the term begins are responsible for all missed assignments and cannot expect that due dates will be altered. If you miss a lecture or plan to miss a lecture, you may be able to make arrangements to sit in on the same lecture in another concurrent offering of the course.

Additional Information: Copies of the transparencies used in lecture and additional course-related information will be made available via course web page on WebCT.

Additional Needs: Students with disabilities or any additional needs are encouraged to set up an appointment at their convenience to discuss any classroom accommodations that may be necessary.

Course Schedule: The initial course schedule may be found on the following page. This is the instructor's first offering of this class – topics and dates are likely to change (exam dates won't).

Final Exam: Tuesday, June 10 – 8:00 – 10:00 PM A330 Creative Arts Center

CS 241-03 Spring 2008 Course Schedule

Note: There are no lab sessions in weeks 1 or 10.

Date	Topic/Activity	Homework Assignment
T 4/1	Course Introduction, review of fundamentals	Review: Ch 1-5, 8
Th 4/3	Review of programming fundamentals	Review: Ch 1-5, 8
T 4/8	Introduction to objects: semantics and syntax	Ch 6 – Project 1 assigned
Th 4/10	Introduction to objects – Style	Ch. 6 & 9
T 4/15	Object-oriented software construction	Ch. 6 & 9
Th 4/17	Software construction recitation	Ch 6 & 9
T 4/22	Inheritance and polymorphism	Ch 11.1-11.5 Project 2 assigned
Th 4/24	Software construction with polymorphism	Ch 11
T 4/29	Abstract classes, interfaces & inheritance	Ch 11.8, 11.9
Th 5/1	Introductory graphics in Java	Ch 7, 13
T 5/6	Event driven programming	Ch 7, 13; Project 3 assigned
Th 5/8	Midterm examination – Includes Labs 1-4, Projects 1-2	Know: Ch 1-6, 8-9, 11
T 5/13	Event driven programming	Ch 7, 13
Th 5/15	Objects and memory	Ch 9.10, 12.3
T 5/20	Recursion	Ch 15
Th 5/22	Advanced I/O	Ch 12
T 5/27	Threads and concurrency	read lecture notes
Th 5/29	Unit testing, logging and debugging	read lecture notes
T 6/5	Bonus topics or catching up...	No lab this week!
Th 6/10	Course Evaluations, recitation and review	Study!