Fall 2007

CS 240: Computer Programming I

Travis E. Doom
Wright State University - Main Campus, travis.doom@wright.edu

Wright State University
College of Engineering and Computer Science
Department of Computer Science and Engineering

Computer Programming I

CS 240

Fall Quarter, 2007

Professor: Travis E. Doom, Ph.D.
Professor's Office: 331 Russ Engineering Center
Office Hours: 3:00-4:00 T TH. Other office hours by appointment (via email).
Email: (Preferred contact) travis.doom@wright.edu
Office Phone: (937) 775-5105

Room & Time:
Section 01: 11:00 - 12:05 MWF 204 FH

Course Description:
Basic concepts of programming and programming languages are introduced. Emphasis is on structured programming and stepwise refinement. Prerequisite: MTH 130 or MPL 5.

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.

Synchronization between T TH and M W F sections: In most quarters, CS240 offers multiple sections. The University block schedules T TH courses to meet twice a week for 1h15min per session (a total of 2.5 lecture hours per week. The University block schedules MWF courses to meet three times a week for 1h05min per session (a total of 3.25 hours per week). In order to keep all sections of CS240 synchronized MWF classes will meet every Monday and Wednesday (2.16 lecture hours per week) but will only meet on SOME Fridays to keep pace with other sections. Expect approximately three Friday class periods to bring the total MWF lecture periods in synchronization with T TH sections.

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 pts. [4 @ 100 pts.]
Laboratory assignments	160 pts. [8 @ 20 pts.]
Mid-term examination	200 pts.
Final examination	300 pts.
TOTAL	1060 pts.

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 a student attain both 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. Assignments must compile to receive credit. Programs that do not compile will not be graded. 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 indicates 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 a 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:

1. Be honest at all times.
2. 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.
3. 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.

4. 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 effect 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.

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.

WRIGHT STATE UNIVERSITY
Wright State University
College of Engineering and Computer Science
Department of Computer Science and Engineering

Computer Programming I

CS 240

Fall Quarter, 2007

COURSE SCHEDULE		
DATE	**TOPIC / ACTIVITY**	**HOMEWORK ASSIGNMENT**
W 9/5	Abstraction, engineering, and the digital computer	Read: Gaddis, Ch 1; No labs this week!
F 9/7	Progamming basics	Read: Gaddis, Ch 2; No labs this week!
M 9/10	Program design and control	Read: Gaddis, Ch 1.6, 3.1, 4.2.
W 9/12	Representing information; scope	Read: Gaddis, Ch 2.
M 9/17	Introduction to methods	Read: Gaddis, Ch 5; Project 1 assigned.
W 9/19	Recitation: programming using methods	Read: Gaddis, Ch 5.
M 9/24	Control flow: Decision structures	Read: Gaddis, Ch 3.
W 9/26	Control flow: Iteration structures	Read: Gaddis, Ch 4.
M 10/1	Recitation: programming iteration/decisions	Read: Gaddis, Ch 4; Project 2 assigned.
W 10/3	Combined assignment, increment/decrement, and other operators	Read: Gaddis, Ch 2.
M 10/8	Buffered I/O and using files	Read: Gaddis, Ch 4.
W 10/10	Recitation: Using files	Read: Gaddis, Ch 4.
F 10/12	Midterm examination	Know: Gaddis, Ch 1-5
M 10/15	Recitation and discussion of Midterm	Read: Gaddis, Ch 4; Project 3 assigned.
W 10/17	Introduction to arrays	Read: Gaddis, Ch 8.
M 10/22	Objects as data, call by reference	Read: Gaddis, Ch 8.
W 10/24	Objects as data: multidimentional arrays	Read: Gaddis, Ch 8.
M 10/29	Using objects: Wrapper classes, ArrayList	Read: Ch 9, Ch 10.; Project 4 assigned.
W 10/31	Using objects: Stringbuilder	Read: Ch 10.
M 11/5	Using objects: Hashes	No labs this week.
W 11/7	Course evaluation, recitation and review	Study/Prepare questions;. No labs this week.
F 11/16	Final examination	10:45-12:45

This page was last modified on Thursday, 23-Aug-2007 09:06:30 EDT. Assignments prior to this date