Spring 2013

CS 1181-01: Computer Science II

Michael Ondrasek
Wright State University - Main Campus, michael.ondrasek@wright.edu

CS 1181 Computer Science - II

Spring 2013

Last Update Wednesday January 2, 2013 at 10:00 a.m.

CRN/Sect	Lecture Time	Day	Room
10358/1	2:30 p.m. - 3:25 p.m.	MWF	RC 152C
10359/2	5:00 p.m. - 6:20 p.m.	TR	RC 152C
15391/3	11:00 a.m. - 12:20 p.m.	TR	RC 346
15393/90	Distance Learning	N/A	N/A

	Laboratory Time	Day	Room
10361/6	3:35 p.m. - 4:30 p.m.	MW	RC 355
10362/7	2:30 p.m. - 3:25 p.m.	MW	RC 355
10363/8	2:00 p.m. - 2:55 p.m.	TR	RC 355
10364/9	4:40 p.m. - 5:35 p.m.	MW	RC 355
15390/10	12:30 p.m. - 1:25 p.m.	TR	RC 346
15392/90	N/A	N/A	N/A

Course Description: This is the second course in a two-semester sequence introducing fundamental concepts and techniques for computer science and engineering. The course focuses on problem analysis, advanced programming concepts using JAVA and fundamental data structures. Students learn to analyze problems and evaluate potential solutions with respect to choice of data structures and computational efficiency. Student are exposed to the underlying implementation of basic data structures available in JAVA libraries and develop the skilled needs to extend existing data structures and design new data structures to solve increasingly complex problems. This is an integrated writing course.

Computer Science/Engineering Learning Outcomes:

By the end of this course, a student will have developed:

- competency in recursion and recursive programming
- competency in fundamental data structures and algorithms
- ability to read, reuse and extend high-level code
- competency in analyzing problem requirements and developing program design
- understanding computation cost associated with alternative designs
- ability to understand and apply defensive programming techniques

Integrated Writing Learning Outcomes:

By the end of this course, students will be able to produce writing that

- demonstrates an understanding of course content,
- is appropriate for the audience and purpose of particular writing task,
- demonstrates the degree of mastery of disciplinary writing conventions appropriate to the course (including documentation conventions),
  and
- shows competency in standard edited American English.

Instructor: Mr. Michael Ondrasek's Office: RC 450; email: michael.ondrasek@wright.edu. Office hours: M-F: 10:00 – 10:45 am, T & R: 2:30 – 3:30 pm, F: 9:00- 10:00 am, and by appointment.

Textbook: **Introduction to Java Programming,** 9rd edition, Y. Daniel Liang, Prentice Hall – Pearson; one of the 3 choices used in CS 1180: regular text with MPL ISBN – 0133050572; loose page version with MPL ISBN – 0133051463; or E-text version with MPL ISBN – 0132991705. Also, without MPL, listed under: ISBN-10: 0132936526 and ISBN-13: 9780132936521

Textbook Web Resources: See www.cs.armstrong.edu/liang/intro9e. for answers to review questions, solutions to even-numbered programming exercises, source code for the book examples, self tests, errata, etc. You will also want to download the Integrated Development Environments JDK with NetBeans with the latest version of Java from: http://java.sun.com/javase/downloads/index.jsp

Pilot: Pilot allows you to access your grades, project and lab assignments, and their submittals. Most of the course materials are posted on Pilot. To get to Pilot type **https://pilot.wright.edu/** into your web browser. You can also use the link on Wings which is found under the Academics tab.

Grading: Students must demonstrate their ability to discuss programming issues as well as solve problems. The underlying metric for the determination of a student's overall grade in this course is the mastery of programming and introductory computer science. 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	500 pts. [5 @ 100 pts.]
Laboratory assignments	200 pts. [10 @ 20 pts.]
Quizzes and homework	140 pts.
Mid-term examination	200 pts. [1 @ 200 pts.]
Final examination	500 pts.
TOTAL	2000 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 attain both an overall passing grade (70%+) in the programming labs and projects.

Policy: There are no late/early/makeup exams unless verifiable emergency and acceptable documentation in writing is provided to the instructor. Although verbal or e-mail notification can be provided, written documentation is required.

Academic Integrity :

1. Be honest at all times.
2. Act fairly towards others. For example, do not seek an unfair advantage over others by cheating. Do not cheat by looking at other individual's work during examinations or at another student's laboratory/project 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.

All work must be your own; sharing of program labatory assignment or project code will result in a grade of "zero" for all those involved. Official university policy will be followed in cases of academic dishonesty.

What IS allowed: Students are allowed to discuss the general requirements of lab assignments 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 lab 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 lab assignment's comments. 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 may NOT discuss, look at, or debug other student's projects. Help on projects should come only from the course instructor and the CS help room. Unless directed to do so, students may NOT work together on lab assignments - students can discuss the lab and/or provide certain help with debugging (see above) but may NOT work together for any extended period of time. Students may NOT use code created by other students or during previous offerings of the course. Students may NOT look at code created by another student (even to debug) until after they have completed the entire lab/project assignment themselves. Students absolutely may NOT turn in someone else's solution with simple cosmetic changes 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.

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 three days 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 or pre-arranged special needs. Poor time management, corrupt files, or network outages will not be considered a sufficient excuse to extend this deadline. Important note: computers go down, networks fail, and data gets destroyed on the day that a project is due. Plan ahead. Back up your work.

Integrated Writing: This course is part of Wright States Integrated Writing Curriculum. You will see that all lab and project submissions require documentation which is worth up to 20% of the total lab or project grade. This is done so that Wright State students will be able to produce writing that:
(1) Demonstrates their understanding of course content,
(2) Is appropriate for the audience and purpose of a particular writing task,
(3) Demonstrates the degree of mastery of disciplinary writing conventions appropriate to good program style and documentation, and
(4) Shows competency in standard edited American English.

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 self-prepared notes on the examinations.

Expectations of Students: We will take attendance, but attendance at lecture is not required although it is strongly encouraged and expected. The Instructor considers it essential to your success in this course that you attend all lectures and lab sessions. Students are expected to study the text. **Even when you don't attend class, you are still responsible for material covered in lecture, lab, and in your text readings.** If you miss a lecture, you may also miss a quiz. If you miss an unexcused exam you will receive a zero score. Students are expected to be on time for lecture and lab sessions: lectures and labs start promptly. Early departure from lecture or lab may be unavoidable, but it is expected that this would be quite unusual. The Instructor feels that it is important that you have your own copy of the correct textbook and edition indicated above. If you have a computer at home, it is important that you practice programming using software discussed in class. If you do not have a computer, it is expected that you will use the computers in Russ Center Room 152B (or other campus locations) to practice programming skills. Questions are encouraged in lecture and lab; however, if there are no questions it is assumed that students understand the lecture, have read, and understand the text and lab materials. If you are having trouble with programs or text readings, it is expected that you will ask questions in class, come during office hours for help, or make an appointment to discuss your questions as needed. Corresponding with the Instructor or Teaching Assistants by e-mail is a good way to get help with text readings or programming assignments. Finally, it is expected that students will follow the Instructor's recommendations concerning lecture preparations. Lecture preparations may include use of the programming lab that is part of text package.

Suggestions: Get an early start on each programming assignment. Most often you will not complete the programming assignment in one lab sessions. You are urged to budget your lab time wisely and expect to spend additional time outside of the formal lab to complete your programming assignments. You should print, review, and study online materials recommended by the Instructor and Teaching Assistants. You can download the source code for the text examples from the author's web site. Whenever possible study your text in front of a computer and actively get involved in trying out the programming concepts on your own. You should try to do all text checkpoint, review questions, and exercises. This can be the most effective way to be successful in the course. If you are uncertain about how you should do this, please discuss with the Instructor or Teaching Assistant. It would be a very good idea to get your own USB 2.0 compatible flash drive (also known as a "thumb drive" or "min-drive") for use in labs and possibly at home. See the Instructor or Teaching Assistants for recommendations and usage. **Always backup your programs!** Keep copies of your work in several different places. E-mail yourself a backup copy.

Programs: Programming lab assignments will be issued in class, during the lab sessions, or on Pilot. Each assignment will state the due date. Assignments usually will be one or possibly two weeks in duration.

Syllabus Changes: The Instructor will not make changes to this syllabus without notification and understanding of all the students in the class. New paper copies will be provided. Changes would be required for the following reasons: (1)

to correct mistakes, (2) to improve student learning, (3) to clarify misunderstands, or (4) to correct serious inconsistencies in policies and/or content compared to other concurrent lecture sections sharing the same labs.

Schedule: See the table below. Topics and order of topics may vary. Exam dates are firm. The topics to be covered each week are listed, followed by the accompanying sections in the text. Not all sections listed are directly covered in detail in class. This schedule is subject to change.

Weekly Topics and Assignments

Week	Topic	Reading
1✓	Introduction	
	Recursion	Chapter 20
	Recursion: Case Studies	
2	Recursion: Case Studies	
	Types of Recursion	
	Recursion	
3	Generic Data Types	Chapter 21
	Generic Methods, Implementation	
	Generics: Case Studies	
4	Basic Data Structures (Java Collections)	Chapter 22
	Iterators	
	Implementation of an ArrayList	
5	Stacks	
	Queues	
6	Linked List	
7	Sets	Chapter 23
	Maps	
	Midterm Examination	
8	Spring Break	
9	Big Oh	Chapter 24
	Algorithm Efficiency	
	Dynamic Programming, Brute Force Algorithms	
10	Divide-and-Conquer, Backtracking Algorithms	
	Introduction to Sorting	Chapter 25
	Merge Sort	
11	Quicksort	
	Heapsort	
	Bucket and Radix Sort	
12	ArrayList Implementation	Chapter 26
	List Implementation	
	List Implementation	
13	List Implementation: Case Studies	
	Stacks, Queues and Priority Queues	
	Binary Search Trees	Chapter 27
14	Binary Search Trees	
	Binary Search Trees: Case Studies	
	Hashing	Chapter 28
15✓	Hashing	
	Hashing: Case Studies	
	Review	
Finals	23 Apr 5:45 pm -7:45 pm Sec 02	25 Apr 10:15am-12:15pm Sec 03 / 26 Apr 2:45 pm -4:45 pm Sec 01

✓ No labs in first or last week of class or the weeks with University holidays

CS 1200: Introduction to Discrete Structures
Spring 2013 MWF 1:25 – 2:20 Russ 154
Last Update: January 6, 2013

Description: Introduction to discrete structures as relevant for computer science. Emphasis on developing a working knowledge of basic mathematical notation and manipulation with discrete structures.

Prerequisites: None

Instructor: Dr. Analee Miranda, analee.miranda@wright.edu; Office hours: By appointment.

Required Textbook: Discrete Mathematics with Applications, Fourth Edition, Susanna S. Epp, Brooks/Cole Cengage Learning, 2011 ISBN 978-0-495-39132-6

Course on PILOT: We will be using PILOT for posting of content, grades and submittal of some assignments or portions of assignments. Students should familiarize themselves with accessing PILOT: https://pilot.wright.edu/. Students are also responsible for printing copies of resource materials from PILOT as needed. Some handouts may be given in class.

Use of e-mail: All registered students have access to a Wright State e-mail account. The Instructor will use only that e-mail account to initiate communication with a student. Important: Please include in any communication with the Instructor, a Subject that starts with "CS1200." For example, a student with a question about HW 1, would use as a Subject "CS1200: Question on HW1 Problem 2."

Grading Policy: Attendance – 10%. Mid-term exams – 50%. One Comprehensive Final – 40%. In order to be allowed to attend the final exam, homework assignments must be turned in and students must take all in-class quizzes. If a student does not achieve an average score of at least 50% in all graded homework and quizzes, then the student will not be allowed to sit in on the Final Exam. Course Grade is based on the following grading scale: A: 100-90, B: 89-79, C: 78-68, D: 67-57, F: 56 or less.

Class Policies: No late or early exams unless verifiable emergency. Attendance at lecture is a component of your grade and as such students are expected to attend all lectures and to participate in class discussion. Attendance will be taken in the course for the purpose of getting to know the students and record keeping for the attendance grade. Students are allowed to miss 2 non-exam or quiz lectures without penalty. All Homework assignments are due in one week after it was assigned before the class begins. Late submittals will not be graded – and a "zero" grade will be recorded in the grade book. All work submitted must be your own unless group assignments are explicitly made by the Instructor; sharing of/or copying problem solutions from any source will result in at least a homework grade of "zero" for all involved and possibly a grade of "F" for the course. University procedures for plagiarism will be strictly followed. Sharing ideas and general mathematical and computer skills with other outside of class is encouraged. Students are expected to read, understand and follow the University Academic Integrity Policy at: http://wright.edu/students/judicial/integrit.html

Exam Dates: (All exams except the Final Exam will be held in lecture – any changes will be announced in class and in Pilot)

1. Quiz 1: Week of January 21, 2013
2. Midterm 1: February 1, 2013
3. Quiz 2: Week of February 28, 2013
4. Midterm 2: March 1, 2013
5. **Final Exam: April 22, 2013 12:30-2:30 PM**