Syllabus - Computer Sciences 312 -Introduction to Programming
The University of Texas at Austin · Spring 2015
Instructor: Vallath Nandakumar

Description: This course focuses on how to write computer programs that solve problems on a general purpose computer. It is intended as a first programming course for computer science majors.

Computer programs are instructions to a computer, and often start out by being written in specialized languages that humans and computers can both understand. This course will use a particular programming language, Java, and most of your learning will be through actual programming.

Objectives: This is a first course in computer programming. The purposes of the course are to learn fundamental computer science concepts including algorithm development, problem decomposition, data types, variables, decision making, iteration, arrays, and 2D arrays. By the end of the course students are expected to be able to implement programs consisting of several programmer defined data types and several hundred lines of code employing non-trivial algorithms.

Estimates of the required effort to pass the class are:

- 3-5 hours per week of studying
- 2-10 hours per week of programming (less early in the course, more later)
- 1000 lines of Java code.

Prerequisites: Credit with a grade of at least C- or registration for Mathematics 305G, or a score of 70 on the ALEKS placement examination.

Lecture: Mon/Wed/Fri, 2 - 3 PM, All lectures will be in GAR 0.102.

Discussion Sections:

Unique	Hour	Room	Day	Discussion Section TA	Grader
51705	11 to 12	GDC 6.202	M	Kaivan Wadia	Kaivan Wadia
51710	12 to 1	GDC 4.302	M	Robert Faulk	Kaivan Wadia

Teaching Staff:

- **Instructor, Vallath Nandakumar,** email: vallathn [at] cs dot utexas dot edu Office: GDC 6.314 . Office Hours: See the Lab Hours page. Or by appointment.
- **Teaching Assistants: Kaivan Wadia** (kaivanwadia [at] gmail), **Robert Faulk** (robertfaulk [at] utexas dot edu), **Basil Hariri** (basilhariri at utexas dot edu).

	
Kaivan Wadia	Basil Hariri

Class web site: http://www.cs.utexas.edu/~vallathn/Spring2015/cs312. Course materials and announcements are available there.

Required Textbook: *Building Java Programs* by Stuart Reges and Marty Stepp, 3rd Edition. ISBN 978-0133360905, Pearson Education / Addison Wesley. Textbook homepage is http://www.buildingjavaprograms.com/. You may buy or rent the class materials.

Purchase options available at the COOP Bookstore or at www.mypearsonstore.com are below. Prices are approximate.

1. Traditional Text without MyProgrammingLab: ISBN: 9780133360905. Suggested Retail Price: $147.00
2. Traditional Text with MyProgrammingLab (**MyProgrammingLab is not required**, but can assist with practicing programming and studying): ISBN: 9780133437300. Suggested Retail Price: $149.33
3. Student Value Edition with MyProgrammingLab: ISBN: 9780133451023. Suggested Retail Price: $104.50
4. Student Value Edition without MyProgrammingLab: ISBN: 978-0-13-337527-5 Suggested Retail Price: $95.60
5. Etext and MyProgrammingLab: ISBN: 978-0-13-337983-9 Suggested Retail Price: $91.30
6. Etext only: ISBN: 978-0-13-337944-0 Suggested Retail Price: $55.99

Class Participation, iClicker

- Purchase an **iClicker** (a remote control device) from the Co-op bookstore or the iClicker website.

  Beware: There may be several incompatible brands of remote control devices on sale. Be sure you get the "iClicker" brand. We are using iClicker2, but iClicker and iClicker+ will work. iClicker Go (using iClicker through your phone) is not acceptable.

  > *The Co-op will buy back the iClicker for about half the current price. I recommend that you put a piece of transparent tape across the serial number on the back of your iClicker to prevent it from being rubbed off with use. The serial number is important when you register the iClicker (see the next step) and you want it intact when/if you sell the iClicker back to the bookstore.*
- Register your iClicker at http://www.iclicker.com/support/registeryourclicker/. You must register at the iClicker website for this course, not on Blackboard.
  Register by providing:
  - your first name
  - your last name
  - your UT EID (which they call your "student ID'")
  - your clicker's ID (the serial number under the bar code on the back of the device)

  Even if you already have a iClicker and registered it in the past you must reregister for this semester.
- Bring your clicker to every lecture. **Participation on the clicker questions is graded.**

- In order to get credit for a given day you MUST:
  - register your clicker prior to class and
  - have your functioning clicker with you and
  - answer at least one of the clicker questions during class.

If you do not meet these requirements you will not receive clicker credit for that day.

Class Discussion Tool: I have set up a discussion group for the class on Piazza.

- Go to the Piazza web site and join the CS 312 - Nandakumar group, for The University of Texas at Austin.
- I will post class announcements and information to the discussion group. You must read Piazza posts, especially instructor announcements.
- Post your questions about the class to the discussion group.

Email: All students must become familiar with the University's official e-mail student notification policy. It is your responsibility to keep the University informed as to changes in your e-mail address. You are expected to check e-mail on a frequent and regular basis in order to stay current with University-related communications, recognizing that certain communications may be time-critical. It is recommended that e-mail be checked daily. Instructions for updating your e-mail address are available here.

You are responsible for checking your e-mail and the class discussion group regularly for class work and announcements.

Software: Required software for programming assignments is described on this web page: http://www.cs.utexas.edu/~vallathn/Spring2015/cs312/software.htm

Computing Facilities:

- **GDC Microcomputer Lab.** PCs are available for doing assignments in the CS Department microcomputer lab located on the 3rd and basement of GDC. You are free to work on your own computer if you wish.

CS Lab Account:

- Students must obtain an account for the CS department microlab to use the microlab machines. Visit this website to request an account: https://apps.cs.utexas.edu/udb/newaccount/ (If you had an account the previous semester it should renew automatically.)
- Accounts take at least a day for account to become active. Please request your account as soon as possible. The microlab machines are useful for printing on the department printers, and for testing out your homework solutions.
- These accounts are only for the CS microlab, and not any other labs or networks at UT. This account is not the same as your UT Direct account.

Schedule: A schedule of lecture topics, reading assignments, and assignment distribution and due dates is available online, via the class web page. The schedule page contains links to slides for the lectures, assignments, and online readings. Readings are to be completed **before** class. The schedule is subject to change.

Grading: Class components used to determine your final average.

Component Type	Number	Points	Total Points
Assignments	12	varies: 10 or 20	230

Quizzes	11	10 each	110
iClicker Participation	44	1 each	44
Background Survey	1	3	3
Midterm 1	1	150	150
Midterm 2	1	200	200
Final Exam	1	300	300

- Programming assignments, iClicker participation , and quizzes can count no more than 350 points towards your final average. Programming assignments, iClicker participation, MyProgrammingLab exercises, and quizzes add up to 387 points. There are 37 points of "slack" in these non-exam grading components. This slack (and slip days explained below) is to account for any problems that may occur during the semester that cause you to miss one of these non-exam components (such as illness, family emergencies, and hardware problems), and corresponds to 1 missed homework, 1 missed quiz, and some missed classes. Quizzes, MyProgrammingLab exercises, and iClicker participation cannot be made up for any reason. For assignments there is some leeway. See the explanation of slip days below.
- The final letter grades will be assigned based on your total points. The maximum possible points is 1000. The grade cutoffs are: <600 = F, 600 - 699 = D, 700 - 799 = C, 800 - 899 = B, >= 900 = A. Plusses and minuses (+ and -) will be assigned to scores within 25 points of the cut offs. So for example total points 875 to 899 earn a B+ and total points 900 to 924 earn an A-.
- Note, points are not added to your final total. The "slack points" are handled by basing your final average on 1000 points instead of 1037 points. There are 1037 points available in the course. Non-exam component points are capped at a maximum of 350. In other words, earning more than 350 points on the non-exam components will NOT result in points being added to your exam score.
- Depending on the results of an exam, exam scores may be adjusted. If so, exam scores may stay the same or improve. Exam scores will never get worse due to a adjustment. No other class components (assignments, clicker participation, or quizzes) are curved.
- You have 6 slip days (max of 2 per assignment) to use through out the term on your assignments if you are not able to turn an assignment in on time. See the assignments page for details on how slip days work. www.cs.utexas.edu/~vallathn/cs312/Assignments/index.htm. Slip days are to account for life circumstances (My hard drive crashed!! I got a virus downloading an mp3!!) and emergencies. Do not use your slip days frivolously. If you use up your slip days and late assignments will not be graded and be assigned a grade of 0. Failure to follow instructions on turning in assignments may result in the loss of slip days.
- The required format and procedures for turning in assignments are available at www.cs.utexas.edu/~vallathn/cs312/Assignments/index.htm. Assignments that are not turned in in the correct format will cause you to lose slip days, lose points, and/or receive a 0 on the assignment.
- Quizzes are given at the beginning of most discussion sections. If you are not present when the quiz is completed you receive a 0 for that quiz. Quizzes cannot be made up under any circumstances. Quizzes are not curved. Quiz grading will be: 10 for a perfect quiz, 7 - 9 for minor mistakes, and 6 or less for little or no effort.
- If a student misses a midterm exam with a verifiable excuse the remaining exams will count for the missed exam.
- If you are **dissatisfied with a grade** you receive on an assignment or test, you must submit your complaint via email, along with supporting evidence, to your grader within ***5 days*** of the date the teaching staff first attempted to return the assignment or test to you.
- Questions about your grade are to be sent to the grader assigned to your section. See the table above for

graders and section unique ids.
- You may request a regrade of an exam if you feel the grading criteria was not applied correctly. To ensure accuracy the entire exam will be regraded and your score may go down.
- There are no opportunities for extra credit in this course.

Guiding Principle - Feedback and concerns about the course are always welcome; legitimate grading errors that are identified in a timely fashion will certainly be corrected, but whining is counter-productive and will only irritate those who evaluate your work to determine grades.

Important Dates for Changing Academic Status and Dropping the Course: Refer to the Registrar's academic calendar for the deadlines for changes in academic status. Highlights are:

- Jan 23: Last day of the official add/drop period; after this date, changes in registration require the approval of the department chair and usually the student's dean.
- Feb 4: (12th class day) Last day to drop for a possible refund. Last day to add a course.
- Apr 6: Last day an undergraduate student may, with the dean's approval, withdraw from the University or drop a class (Q drop) except for urgent and substantiated, nonacademic reasons. Last day a student may change registration in a class to or from the pass/fail or credit/no credit basis. **In practice I always grant approval for "Q drops" up to this date as long as there are no issues with academic integrity.**
- After Apr 6, students must go to the academic advisors in their college Dean's office. To be eligible for an **incomplete (UT uses the symbol X to indicate incompletes)** you must have a letter grade of **C-** or better and a written, verifiable excuse for missing the last test. This is a **necessary** but **insufficient** condition for receiving an incomplete.

Students experiencing significant nonacademic problems (extended health problems or family emergencies) should contact the CNS Dean's Office (WCH 1.106, (512) 471-4536) or the Dean of Student's Office (http://deanofstudents.utexas.edu/emergencyresources.php) for assistance.

See the College of Natural Science Guidelines and Procedures page for more information.

Academic Dishonesty: Taken from the CS department Code of Conduct.

"The University and the Department are committed to preserving the reputation of your degree. It means a lot to you. In order to guarantee that every degree means what it says it means, we must enforce a strict policy that guarantees that the work that you turn in is your own and that the grades you receive measure your personal achievements in your classes:

Every piece of work that you turn in with your name on it must be yours and yours alone unless explicitly allowed by an instructor in a particular class. Specifically, unless otherwise authorized by an instructor:

- Students may not discuss their work with anyone except the instructor and other members of the instructional staff (instructor, TA, lab proctor or partner on a pair assignment).
- Students may not acquire from any source (e.g., another student or an internet site) a partial or complete solution to a problem or project that has been assigned.

You are responsible for complying with this policy in two ways:

1. You must not turn in work that is not yours, except as expressly permitted by the instructor of each course.
2. You must not enable someone else to turn in work that is not theirs. Do not share your work with anyone else. Make sure that you adequately protect all your files. Even after you have finished a class, do not share your work or published answers with the students who come after you. They need to do their work

on their own. This means do not post your solution code to any public web site such as pastebin. Also, do not post your work to the web even after you have completed CS312.

The penalty for academic dishonesty will be a course grade of F and a referral of the case to the Dean of Students. Further penalties, including suspension or expulsion from the university may be imposed by that office.

One final word: This policy is not intended to discourage students from learning from each other, nor is it unmindful of the fact that most significant work in computer science and in the computing industry is done by teams of people working together. But, because of our need to assign individual grades, we are forced to impose an otherwise artificial requirement for individual work. In some classes, it is possible to allow and even encourage collaboration in ways that do not interfere with the instructor's ability to assign grades. In these cases, your instructor will make clear to you exactly what kinds of collaboration are allowed for that class."

For CS312 the policy on collaboration is modified as follows:

If you are repeating the course you may reuse code you completed on your own. You may NOT use code from a program you worked on as part of a pair or code that was from a program involved in an academic dishonesty case. You must start from scratch on any and all programs that:

- were part of an academic dishonesty case
- you worked with a partner during a pervious semester
- you are working with a partner this semester

You are encouraged to study for tests together, to ***discuss*** methods for solving the assignments, to help each other in using the software, and to discuss methods for debugging code. **Essentially if you *talk* about an assignment with any one else you are okay, but the moment you start *looking at someone else's source code* or *showing someone else your source code* you have crossed the line into cheating.** You should not ask anyone to give you a copy of their code or, conversely, give your code to another student who asks you for it. Similarly, you should not discuss your algorithmic strategies to such an extent that you and your collaborators end up turning in exactly the same code. **Discuss** high level approaches together, but do the coding on your own.

Examples of cheating are many and include accessing another student's account, looking at someone else's solution code, copying or downloading someone else's solution code, referring to solutions from previous semesters, having another student walk you through the solution and how to code it, having another student perform significant debugging of your code, having another student write your code for you and / or allowing others to copy of access your solution code. This means you **shall not** look on the internet for code to solve your problems.

Examples of allowable collaboration include discussions and debate of general concepts and solution strategies and help with syntax errors.

The code you can reuse in this course are:

1. You may use any code you develop with the instructor, TAs, or proctors.
2. You may use code (with attribution) from the class slides and the class coding examples.
3. You may share additional test cases and expected results of test cases. You may not share solution code or experiment code.

You shall not make use of code you find from other sources including the world wide web. Materials from the web should only be used for educational purposes. Thus, you can read about linked lists and look at examples of linked list code, but you must not copy any code from the web or be looking at any of this code from the web when writing anything you turn in. If you discuss an assignment with another student or look at examples from

the web you should employ the World of Warcraft Rule:

World of Warcraft Rule: After a discussion with another student or looking at example code you should do something that has nothing to do with computer science or programming for al least half an hour -- playing World of Warcraft or other similar activity. (Watching a sitcom, reading a book, working on another class.)

You are also allowed to post short segments of code (2 lines or less) of code that are giving you syntax errors to the class listserv in order to get help on fixing the syntax error.

If you have any doubts about what is allowed, ask the instructor.

Plagiarism detection software will be used on assignments to find students who have copied code from one another.

For more information on Scholastic Dishonesty see the University Policy on Scholastic Dishonesty

---

Religious Holidays: By UT Austin policy, you must notify me of your pending absence at least fourteen days prior to the date of observance of a religious holy day. If you must miss a class, an examination, a work assignment, or a project in order to observe a religious holy day, you will be given an opportunity to complete the missed work within a reasonable time after the absence.

Students with Disabilities: students with disabilities may request appropriate academic accommodations from the Division of Diversity and Community Engagement, Services for Students with Disabilities, 471-6259, www.utexas.edu/diversity/ddce/ssd/. **Please present written proof of your special need from the above-mentioned office not later than the twelfth day of class, i.e. not later than Feb. 4, 2015**. If your accommodation includes a longer duration or quiet facilities for the midterms or final, it is your responsibility to secure a spot in the SSD facilities on the day of the exam, overlapping with the regular exam time (if possible). You will have to request these facilities a month in advance of the test. The spots fill up fast!

To the CS 312 home page