University of California, San Diego
CSE8A: Introduction to Computer Science: Java I
Spring 2020

Welcome!

I am incredibly privileged to welcome you to this class. For most of you, I also extend a warm welcome to the computing profession. The entire CSE department and I look forward to working with you in the days and years to come to help you develop the skills and abilities to transform and improve society through the understanding and application of computation – no matter your profession or academic degree program.

What does this course offer to you?
This course provides an introduction to the basics of computation. This includes understanding and being able to employ (in a programming language) basic computing concepts which allow us to control a computer and to design programs for it so it can solve problems for us. No prior programming experience is expected for this class. *In this class, our goal is to help you experience the thrill of getting a computer to solve a problem of your choosing – by expressing that solution in the Java programming language.*

Why would I want to learn a programming language rather than just a computer application (like PhotoShop)? First, applications like PhotoShop are, themselves, computer programs. Someone wrote them as solutions to a set of problems they wanted to be able to solve. But, what if you have ideas for solving a problem that PhotoShop can't solve? How can you get a computer to follow your directions? You have to write a program. Programs are one of the most direct ways one can give instructions to a computer. They give you a great deal of control over the computer – the kind of control needed to develop novel software to solve problems of your choosing. The kind of control that will allow you to explore and design solutions for society's ever changing computing needs.

Particularly in recent years, computation has come to play a critical role in the advancement of many fields – including biology, medicine, chemistry, film, archeology, etc. Our world is full of data – and being able to manipulate data or analyze it in an efficient or novel way can lead to dramatic new findings in many fields. In this class, you will learn the computational basics of flow of control and data representation which will allow you to express solutions to many problems – and to learn other programming languages (besides Java) to allow you to accomplish your goals.

How, exactly, are you going to accomplish this?
The class has two components – "lecture" and lab. They are designed to fit together. Both seek to give you introductory experience in the critical aspects of software development including understanding of new computational concepts through being able to read, understand, modify, and debug code in addition to being able to write code. Notably, most modern software is developed in teams. We have designed pair programming experiences, lecture team discussions, and programming assignments to start giving you practice to be a successful software developer. Specifically, you can expect the following cycle to repeat itself throughout the weeks of the course:

- Each week, **before lectures**, you'll complete a number of key items to prepare you for the activities in lecture and to be able to ask questions in lecture:
  1. Read a set of pages in the textbook to give you your first experience with the new computational concepts of the week. To figure out if you understood the basics of the reading, you'll take an online "**reading quiz**" due at noon the day before lecture.
  2. Watch a podcast of our lecture from last Spring. You should engage in solving the problems from the podcast and pay attention to the questions and answers asked during an in-person version of the class. Then you'll need to take an online "**podcast quiz**" to ensure you understood the material well.
- Then, **in lecture online, we'll work together** to test out your understanding of the material and discuss confusion and misconceptions that you may have about it. Note – this makes lecture a very interactive place – you are expected to come prepared to engage and help yourself and others gain a much deeper understanding of the concepts. **Lecture will depend heavily on your bringing questions to ask.**

- Lab comes in as the next step. In lab you will work in teams to get your **first "hands on experience"** with putting computational concepts into practice. Labs are short, and directed at likely mistakes you might make. They will give you the time and support to understand those mistakes and ask questions about them so you won't make them again. At the end of each lab, you'll complete a "**Lab Quiz**".
- Finally, both lectures and labs are a time to help you get prepared for the **programming assignments** (we call them Problem Solving Assignments, or PSAs) – where you can experience for yourself the thrill of getting a computer to follow your directions and create something of your choosing. This is the part where you can really test out your knowledge and understanding of computational concepts. Every line of code you write should be understandable to you – and you will be expected to be able to explain how you code works and why to someone else.

How will you (and the professor) know if you are making progress in your learning?
Each aspect of this course is designed to help you develop deeper understanding of the computational concepts that will allow you to produce computer programs to solve problems of interest. For each new concept, you'll follow the reading, podcast, lecture, lab, and homework process to give you repeated experience with that concept, at increasingly complex and less constrained levels. *You should reflect on each of these experiences each week to determine for yourself if you are making adequate progress – and plan specific changes if you are not.*

The key goal for you (and me) in this course is that, by the end, you can employ the computational concepts we've learned to solve problems by writing a computer program. In a very short period of time (maybe just a year from now) you may be paid to help people get the computer to solve their problems – through writing a computer program. Whether or not you fully understand a particular concept at the period of time we work on it in class only is important in terms of helping you stay on track with the pace of the course. The critical thing is that – by the end of our ten weeks together – you can understand and employ those computing concepts to effectively solve problems. This means that you can read, understand, explain, modify, debug, and write code using these concepts. These are the first skills to be developed in becoming a software developer and a computer scientist.

Important Course Details
Instructor: Leo Porter, Ph.D.
- EMAIL: leporter@eng.ucsd.edu (**but most questions should be asked on piazza**)
- OH: Please see course website for most accurate dates/times.

TAs: Rachel Chung, Adrian Salguero, Xinghan Wan, and Henry Zhang

Class website: canvas.ucsd.edu
All Course materials will be on canvas and/or piazza this quarter. Please go to canvas.ucsd.edu and login using your ucsd credentials. You are responsible for reading and staying up to date with the class website. Announcements, assignments, etc. will be posted on this site and/or on piazza (see **piazza** site for homework, concept questions). See **gradescope** (linked from piazza) for assignment submission.

Learning Goals: Students completing this course will be proficient in writing simple Java programs. This includes, but is not limited to: language features such as control structures, loops, classes, objects, variables, file I/0, and basic data structures including arrays. Other topics may include recursion and fundamental algorithms, such as elementary searching and sorting algorithms.

Meeting times and places:
- See your schedule of classes for your enrolled times.
- Lecture and Lab are considered required – you must at least watch the videos.
- Discussion is considered optional, but highly recommended

Open Lab Tutoring: If you need assistance with your programming assignments, we will have tutors available online to help you with your programs.

Course materials (required)
- Mark Guzdial and Barbara Ericson, Introduction to computing and programming with Java: A Multimedia Approach, April 2006. Prentice Hall.
- ~~I clicker Classroom Response System Transmitter~~

Class Announcements

All announcements, updates on homework assignments, etc. will be posted on canvas and/or piazza. All students are responsible for announcements and information on canvas and piazza.

Lecture Notes

I will lecture writing on ppt slides and will release my inked slides as pdf after class. I will also video podcast all lectures.

hourGrading Information:

o The grade for this class (out of 100% total) will be as follows:

- Reading and other online quizzes: 5%
- Podcast quizzes: 5%
- Professionalism: 1%
- PSAs: 35%
- Lab quizzes: 10%
- Midterm: 19%   **Thursday, April 30th during class time**
- Final: 25%

I will grade on a 90%, 80%,70%,60% scale. "Plusses" and "minuses" are given out only at my discretion. All grades are FIXED within 7 days of the grade being posted on moodle or gradescope - no regrade requests will be taken after that time.

ADDITIONALLY, YOU MUST SCORE AT LEAST A 55% ON THE FINAL EXAM TO PASS THE COURSE. ANYONE WHO SCORES LESS THAN A 55% ON THE FINAL EXAM WILL RECEIVE AN "F" – REGARDLESS OF THE PERCENTAGES DESCRIBED ABOVE.

o **Reading quizzes:** You are expected to come to class prepared to learn and that means you should do the assigned reading and complete the assigned reading quiz. In some cases, quizzes will be assigned related to course content, but not necessarily reading quizzes. The combination of coming to class prepared and participating in class (see below) should enable you to excel throughout the class. You cannot make up missed reading quizzes, however I will drop your lowest reading quiz score.

o **Podcast quizzes:** Part of coming to lecture prepared is to have watched a lecture from a previous quarter and completed a quiz about the content of that lecture. Again, this is necessary preparation for engagement in class. You cannot make up missed podcast quizzes, however I will drop your lowest podcast quiz score.

o **Participation**: You are highly encouraged to attend class (remotely) and participate in the in-class activities as well as ask questions. Attendance in a lecture is NOT REQUIRED per campus guidelines for Spring 2020, however you should plan to attend if at all possible.

o **Professionalism:** I expect everyone in the class to conduct themselves in a professional manner. *I consider professional behavior to be a program-level academic outcome of UCSD.* Though broadly defined here, it includes (at least), professional conduct with the teaching staff and your fellow classmates. Some examples of unprofessionalism in prior courses include: excessive arguing with teaching staff over assessment outcomes (grades), belittling/rude/unkind behavior toward other students or teaching staff, and excessive lack of resourcefulness (e.g., e-mailing course staff with questions already answered on piazza or in this syllabus). Though rare, I reserve the right to deduct participation points from repeated unprofessionalism. Note that particularly severe infractions (e.g., sexism, racism, lying - which are never tolerated in our community) may be subject to campus Academic Conduct Charges.

o **Programming Assignments:** Programming assignments are where you will put all your learning into practice. You must submit by the assignment deadline.

  ▪ You must submit your assignment following the directions provided with the assignment (either through ieng6 or through gradescope).
  ▪ We will provide you with some of the same testing code we will run on your code. This means you should know, as least in part, the grade you'll get as you submit it. This allows you to keep working until you at least pass these tests. **Please note that some tests are randomized which means you might get full credit on one run and less than full credit on another. It is up to you to ensure correctness.**
  ▪ Regrades must be submitted within 3 days of the grades being released.
  ▪ You will be engaging in pair programming (or with approval, work alone) for a number of the assignments and working alone for others. You may choose your own partner but you must choose your partner before starting an assignment. You are required to do pair programming if you have a partner. Be sure you read and understand the rules regarding pair programming.
  ▪ Assignments that are submitted late, do not compile or do not run, or are misnamed, will be given 0 points.

- Programming assignments will often require you to film a video of you explaining a particular portion of the code. **You will need to include that video in your submission.**
- **Late programming assignments are not allowed, no exceptions.** We will make every effort to return assignments to you in a timely manner -- limiting your ability to turn things in late is, unfortunately, critical to that goal. **You may drop your lowest assignment grade – use that dropped score judiciously!**
- **Labs:** Labs provide you with the opportunity to get some hands on practice with a TA and tutor present. You will need to complete a lab quiz each week and that quiz will be graded. **Your lowest lab score will be dropped – again use that dropped score judiciously!**
  - **Exams:**
    - **Midterm**: See date above. Exams must be taken with your enrolled section during lecture time.
    - **Final exam**: The final will be inclusive of all course material and must be taken at the assigned time.
    - **There are no alternative/make-up exams.**
    - You have the right of appeal for grading on all tests. Please check your exam over carefully when you receive it and be sure to request a regrade in the case of a grading error. Also review your exam to understand where you made errors, if in doubt, come to see us in Office Hours (do not use regrade requests to find out what you did wrong). Note that frivolous regrade requests (requests for partial credit for a wrong answer or to argue why your answer is correct) may result in additional points deducted.
  - If you are taking the course pass/fail, you must get at least a C- to pass (for Sat/Unsat, a B-).
  - The only exceptions to the rules regarding no late assignments or exams are extended absences (one week or more) due to verifiable extraordinary circumstances, and absences due to official UCSD activity travel. In the case of absences due to a UCSD activity travel, you must give me a list of travel dates as soon as it is available.
  - If you are eligible for accommodations as per UCSD OSD policies, contact me by the end of week 2.

Pair Programming:
  - If you work with a partner on a programming assignment, you need to follow pair programming guidelines. See here for more details.
  - If you are doing pair programming, ALL work on an assignment must be done together (likely via zoom/skype and screen sharing). That means if you are working on the assignment alone and you have a partner for the assignment, both of you are violating the academic integrity policy. Tutors who see this are required to report it to the instructor and the instructor is required to report it to campus (which means you need to come as a pair to Lab Hours).
  - If a partnership dissolves and you wish to break the partnership. Both students are required to delete all code produced up to that point and start fresh. Any code in common between students in a dissolved partnership is a violation of our course academic integrity policy.
  - Only agree to pair programming if you and your partner can agree on *at least* 6 hours per week when you can both be together working on the assignment. You may need to fill out a form for each assignment stating your partner *before beginning the assignment.*
  - As you are required to work together at all times, both members of a partnership are responsible for knowing all elements of the code submitted. We reserve the right to ask you about code submitted. We may ask you about your code in class, in the lab, on exams, and at our discretion. If you receive a request from a TA, Tutor, or the instructor to meet them to go over your code, you need to meet with them within 48 hours. If you are unable to explain code submitted in your name, you are in violation of the academic integrity agreement.

Integrity:
  - **Review and sign** the course Integrity of Scholarship Agreement
  - Cheating WILL be taken seriously. It is not fair to honest students to take cheating lightly, nor is it fair to the cheater to let him/her go on thinking that cheating is a reasonable alternative in life.
  - The following is *not* considered cheating:
    - Working on programming assignments using pair programming with your pair programming partner
  - The following is considered cheating:
    - discussing assignments with another student in the class.
    - working alone on an assignment for which you have a pair programming partner.
    - using homework or programming solutions from the web, previous versions of the class, or anywhere else.
    - receiving, providing, or soliciting assistance from another person/device during a test.
  - Review the Integrity of Scholarship Agreement for full details about what is permitted.
  - To detect instances of academic integrity violations in programming assignments we will use 3rd party software.
  - Assignments and labs are not intended to be the grade-makers, but to prepare you for the tests, which are the grade-makers. Cheating on an assignment or lab is not only unethical, but shows a fundamental misunderstanding of the purpose of assignments and labs.
  - Penalties -- If I become aware of any violations of these rules by a student I will initiate the actions described in the Policy on Academic Integrity. Integrity violations may result in a zero for the assignment, a zero for that portion of your grade, an "F" in the course, among others.

Consent to Participate in Educational Research
University of California, San Diego

Consent to Act as a Research Subject

Investigating the Impact of Pedagogical Choices on University Student Learning and Engagement

Who is conducting the study, why you have been asked to participate, how you were selected, and what is the approximate number of participants in the study?
Gabriele Wienhausen, Director of the Teaching and Learning Commons, together with her education research colleagues is conducting a research study to find out more about how pedagogical choices affect student learning and experience in the classroom. You have been asked to participate in this study because you are a student in a class that is being studied or used as a control. There will be approximately 500,000 participants in this study.

Why is this study being done?
The purpose of this study is to create knowledge that has the potential to improve the learning and educational experience of students at UC San Diego and beyond.

What will happen to you in this study and which procedures are standard of care and which are experimental?
If you agree to be in this study, the following will happen:
  ▪ Your data from this class including grades, homework and exam submissions, and survey responses will be included in the analysis to determine the effectiveness of the pedagogical techniques used in this course compared to other similar courses.

How much time will each study procedure take, what is your total time commitment, and how long will the study last?
Your participation involves only agreeing to let us use your data in our analysis. It will require no time on your part above the time you put into this course without agreeing to the study.

What risks are associated with this study?
Participation in this study may involve some added risks or discomforts. These include the following:

1. A potential for the loss of confidentiality. We will not share your personally identifying data with people outside our research team. Data will only be kept in anonymized form for research purposes. Course data will not be used for this research study until after final grades have been posted and will be rendered confidential by removing any identifiers before analysis. Your instructor will not know whether or not you are participating in this study until after final grades have been posted. Data from students who opt out of the study will be removed prior to data analysis. Research records will be kept confidential to the extent allowed by law. Research records may be reviewed by the UCSD Institutional Review Board.

Since this is an investigational study, there may be some unknown risks that are currently unforeseeable. You will be informed of any significant new findings.

What are the alternatives to participating in this study?
The alternatives to participation in this study are not to participate. If you choose to opt-out of participating in this research study, we will exclude your data from analysis. Whether you participate will have no impact on your experience or grade in the associated class as the professor will not know who is or is not participating in the study until after final grades are assigned.

What benefits can be reasonably expected?
There is no direct benefit to you for participating in the study. The investigator, however, may learn more about how to improve student learning, and society may benefit from this knowledge.

Can you choose to not participate or withdraw from the study without penalty or loss of benefits?
Participation in research is entirely voluntary. You may refuse to participate or withdraw or refuse to answer specific questions in an interview or on a questionnaire at any time without penalty or loss of benefits to which you are entitled. If you decide that you no longer wish to continue in this study before the end of the quarter, simply respond to the online opt-out form here: https://goo.gl/forms/JSBRjEmkES6W6xYc2. If you decide to opt out after the quarter has ended, you must contact Laurel Nelson (laureln@ucsd.edu) and give the quarter and the course from which you would like your data withdrawn.
You will be told if any important new information is found during the course of this study that may affect your wanting to continue.

Can you be withdrawn from the study without your consent?
The PI may remove you from the study without your consent if the PI feels it is in your best interest or the best interest of the study. You may also be withdrawn from the study if you do not follow the instructions given you by the study personnel.

Will you be compensated for participating in this study?
You will not be compensated for participating in this study.

Are there any costs associated with participating in this study?
There will be no cost to you for participating in this study.

Who can you call if you have questions?
Gabriele Wienhausen and/or her colleague has explained this study to you and answered your questions. If you have other questions or research-related problems, you may reach Gabriele Wienhausen at gwienhausen@ucsd.edu or (858) 534-3958.
You may call the Human Research Protections Program Office at 858-246-HRPP (858-246-4777) to inquire about your rights as a research subject or to report research-related problems.

Your Consent
If you consent to participate in this study and are at least 18 years old, no action is needed. If you DO NOT consent to participate in this study, or you choose to opt-out at any time during the quarter, please submit this form online at https://goo.gl/forms/JSBRjEmkES6W6xYc2. Your instructor will not have access to the list of students who opted out until after grades are posted. Note that you must separately opt-out of the study for each course involved in this study.