Course Syllabus for CS 458
Software Engineering
Fall 2015

Basic Course Information:

Instructor: Sharon Tuttle

Lecture times and location: Tuesday, Thursday 3:00 pm - 4:20 pm BSS 408

Lab time and location: Wednesday 3:00 pm - 4:50 pm BSS 313

Instructor's office: BSS 322

Instructor’s e-mail: st10@humboldt.edu or sharon.tuttle@humboldt.edu or smtuttle@humboldt.edu

Instructor’s office phone: (707) 826-3381

Instructor’s office hours:
Monday 3:00 - 4:30 pm
Wednesday 11:00 am - 12:00 pm
Thursday 1:00 - 2:00 pm
Friday 3:30 - 5:00 pm
or by appointment

Course public web page: follow link from:
http://users.humboldt.edu/smtuttle/
or follow link from course Moodle site

Course Description:
From the HSU catalog: Introduction to software engineering principles and methodologies in the context of a semester-long software team project.

This is an undergraduate-level introductory course in software engineering, described in the ACM/IEEE-CS Joint Task Force on Computing Curricula’s Computer Science Curricula 2013 as "the discipline concerned with the
application of theory, knowledge, and practice to effectively and efficiently build reliable software systems that satisfy the requirements of customers and users." Course topics will include software engineering principles, development methodologies, requirements analysis, project planning, software design, software construction, software process metrics, project management, software testing, quality assurance, and team processes.

In this course students will enhance their existing skills not only in programming but also in oral and written communication and in working as part of a team. Course content is delivered through a combination of lectures, in-class activities, written homework, technical writing assignments, and a significant team project in which students apply some of the software engineering principles discussed. It is also expected that students will participate in a formal presentation of their team project.

Course Prerequisites:
(CS 328 AND CS 374), OR instructor's consent.

Course Objectives:

After successfully completing this course, students should be able to:

• for each of various software project scenarios, describe the project’s place in the software life cycle, identify the particular tasks that should be performed next, and identify metrics appropriate to those tasks.
• apply key elements and common methods for elicitation and analysis to produce a set of software requirements for a medium-sized software system.
• demonstrate the capability to use a range of software tools in support of the development of a software product of medium size.
• distinguish between the different types and levels of testing (unit, integration, systems, and acceptance) for medium-size software products, and create, evaluate, and implement a test plan for a medium-size code segment.
• demonstrate through involvement in a team project the central elements of team building and team management.
• participate in a team capstone project whose final report includes written justifications and rationales for design decisions made.

CS Program Learning Outcomes that this course addresses:
This course addresses departmental learning outcomes of:

• Computational Thinking
• Technical Writing
• Self-Directed Learning
• Communicating and Collaborating

This course addresses computational thinking, technical writing, self-directed learning, and communicating and collaborating at advanced levels.

HSU Learning Outcomes that this course addresses:
This course explicitly contributes to students’ acquisition of skills and knowledge relevant to HSU Learning Outcomes:

HSU graduates will have demonstrated:
• Effective communication through written and oral modes.

* Some of these are adapted from the ACM Computer Science Curriculum 2001, available from link at: http://www.acm.org/education/curricula-recommendations
• Critical and creative thinking skills in acquiring a broad base of knowledge and applying it to complex issues.
• Competence in a major area of study.
• Appreciation for and understanding of an expanded world perspective by engaging respectfully with a diverse range of individuals, communities, and viewpoints.

HSU graduates will be prepared to:
• Succeed in their chosen careers.
• Take responsibility for identifying personal goals and practicing lifelong learning.

Required Course Materials:
• Turning Point RF Response Clicker, available at the campus bookstore
  – BEWARE!! My understanding is that HSU does NOT support Responseware, EVEN THOUGH the TurningTechnologies module in Moodle may MENTION it!
    SO -- I would recommend that you DO NOT PURCHASE Responseware for this course, since students have told me they do NOT give refunds if you are unable to use it!
• Any additional required readings will be made available either via on-line links or via resources available through the HSU Library such as the ACM Digital Library and Safari TechBooks Online.

Clickers:
We will be using Turning Technologies student response clickers in class. There is significant literature indicating that using clickers may increase student engagement and success in learning.

Students purchase this clicker (they are available from the campus bookstore); purchased clickers can be returned at the end of the semester for a partial refund of the purchase price or they can be kept for use in future HSU courses. You register your clicker at the beginning of each semester by logging into Moodle and going to the course Moodle site; in the lower left or right columns, you should find a block labeled "Turning Technologies" -- if you click on the link in that block, you should be able to enter your clicker code (consisting of 6 characters/digits) from the back of your clicker. You then bring your clicker to every class meeting (lectures and labs).

These clickers will be used for in-class questions, which will be interspersed within class meetings. These will usually be given in a think-pair-share fashion, in which you answer a question first individually, and then discuss your answer with another student, discussing why you think your answer is correct; if they gave a different answer, you try to persuade them that yours is the correct answer, and then either of you can change your answer if you wish. The response system will record the overall class response percentages as well as keep track of individual answers.

Typically, you will receive:
• 2 points for a correct answer,
• 1 point for an incorrect answer, and
• 0 points for no answer,
• but with a maximum semester clicker-questions grade of 120.
• (There may be some no-point questions from time-to-time as well -- such questions will be noted if/when they come up. Also, reading quizzes, answered individually and graded more traditionally, may be given using clickers -- whether these will be given using clickers or using Moodle is still being determined, and we may try
both approaches during the semester.)

Thus you will be rewarded for regular attendance and participation. If you miss a class session, you miss that day's clicker questions and cannot make them up (except for extraordinary circumstances). However, there will be a sufficient number of questions asked to allow for the possibility of extra credit (up to a maximum clicker grade of 120) or to make up for a day that you are out due to illness (although note that you are still responsible for finding out what you missed on such days).

If you forget your clicker for a class meeting, then up to 5 times you may still receive some clicker credit, usually minus a 2-point penalty, by e-mailing me your clicker answers for that day, by 11:59 pm on that day, using a Subject: line of: Subject: CS 458 Clicker Answers for <date>. Later e-mails, or e-mails without the proper Subject: line, might not be accepted for credit.

The idea is that the clicker questions will help you to see if you are starting to understand concepts being discussed; sometimes they will also provide review of concepts discussed previously. Clicker questions are typically quite different from exam questions (since clicker questions are typically multiple-choice questions, while exam questions will rarely be multiple-choice). They still enable you to get some immediate feedback regarding whether you are grasping course concepts, whether you need to pay more attention to course discussions and/or readings, etc. They may even help me to know what concepts might need more explanation in-class.

I hope to run tests of the system during the first week's class meetings, and hope to begin asking questions that "count" during the second week's class meetings. Therefore, you must purchase your clicker and register it as soon as possible. If there is an issue with this (for example, if the bookstore runs out of clickers), contact me immediately.

Finally, please note that use of another CS 458 student's clicker, or having someone else use your clicker in a CS 458 class session -- that is, pretending that someone is in class who actually is not -- is considered to be cheating, with the same policies applying as would be the case if you turned in someone else's work as your own or permitted someone else to copy your work. Please ASK ME if you are not sure what I mean by this.

**Grading Breakdown:**

If you are a Computer Science (CS) major, note that you must earn at least a C- in CS 458 for this course to count towards your major.

Your semester grade will be determined by the percentage of points that you earn, subject to some minimum requirements. Here are the grade percentages, followed by the other minimum requirements:

- **Homework assignments:** 15.0%
- **Reading questions:** 10.0%
- **Clicker questions:** 10.0%
- **Attendance/Participation:** 10.0%
- **Exams:**
  - Exam 1: 10.0%
  - **Final Exam:** 15.0% Tuesday, December 15, 3:00 - 4:50, BSS 408
- **Project:** 30.0%

**Grade Requirements:**

- To earn a grade of C- or better in this course, the following four requirements must all be met:
  1. your overall semester average must equal or exceed 70% - this is to show a reasonable level of overall mastery of the course material.
  2. the average of your Exam 1 and Final Exam grades must equal or exceed 60% - this is to show that you understand at least a minimal reasonable level of the most important course concepts.
  3. the average of your Homework assignments must equal or exceed 60% - because this part includes (along with coverage of core course concepts) technical writing and presentation experience that are not reasonably
tested on exams, but are nevertheless important course components.

4. your overall Project grade must equal or exceed 60% - because the team course project is an essential component of this course. If you have not met this requirement, you have not truly shown competence in this course's material, since you have not successfully practiced applying software engineering practices in the context of a team programming project.

- If all four requirements above are not met, then your semester grade will be either D+ or the letter grade computed according to the mapping given below, whichever is lower.
  - (That is, if a student had an overall semester average of 74% but an overall Project grade of 55%, that student would receive a D+ for their semester grade; if a student had a Homeworks average of 61% and an Exams average of 71%, but an overall semester average of 65%. then that student would receive a D for their semester grade. You are expected to ASK ME if this aspect of the grading policy is not clear to you.)

- Including the four requirements noted above, your semester grade will be computed according to the mapping given below:

<table>
<thead>
<tr>
<th>Overall Percentage (based on the given weights)</th>
<th>Exams Average</th>
<th>Homework Average</th>
<th>Overall Project Grade</th>
<th>Letter Grade</th>
</tr>
</thead>
<tbody>
<tr>
<td>&gt;= 93</td>
<td>&gt;= 60</td>
<td>&gt;= 60</td>
<td>&gt;= 60</td>
<td>A</td>
</tr>
<tr>
<td>&gt;= 90 and &lt; 93</td>
<td>&gt;= 60</td>
<td>&gt;= 60</td>
<td>&gt;= 60</td>
<td>A-</td>
</tr>
<tr>
<td>&gt;= 87 and &lt; 90</td>
<td>&gt;= 60</td>
<td>&gt;= 60</td>
<td>&gt;= 60</td>
<td>B+</td>
</tr>
<tr>
<td>&gt;= 83 and &lt; 87</td>
<td>&gt;= 60</td>
<td>&gt;= 60</td>
<td>&gt;= 60</td>
<td>B</td>
</tr>
<tr>
<td>&gt;= 80 and &lt; 83</td>
<td>&gt;= 60</td>
<td>&gt;= 60</td>
<td>&gt;= 60</td>
<td>B-</td>
</tr>
<tr>
<td>&gt;= 77 and &lt; 80</td>
<td>&gt;= 60</td>
<td>&gt;= 60</td>
<td>&gt;= 60</td>
<td>C+</td>
</tr>
<tr>
<td>&gt;= 73 and &lt; 77</td>
<td>&gt;= 60</td>
<td>&gt;= 60</td>
<td>&gt;= 60</td>
<td>C</td>
</tr>
<tr>
<td>&gt;= 70 and &lt; 73</td>
<td>&gt;= 60</td>
<td>&gt;= 60</td>
<td>&gt;= 60</td>
<td>C-</td>
</tr>
<tr>
<td>&gt;= 70</td>
<td>&lt; 60</td>
<td>any</td>
<td>any</td>
<td>D+</td>
</tr>
<tr>
<td>&gt;= 70</td>
<td>any</td>
<td>&lt; 60</td>
<td>any</td>
<td>D+</td>
</tr>
<tr>
<td>&gt;= 70</td>
<td>any</td>
<td>any</td>
<td>&lt; 60</td>
<td>D+</td>
</tr>
<tr>
<td>&gt;= 70</td>
<td>any</td>
<td>any</td>
<td>any</td>
<td>D+</td>
</tr>
<tr>
<td>&gt;= 67 and &lt; 70</td>
<td>any</td>
<td>any</td>
<td>any</td>
<td>D+</td>
</tr>
<tr>
<td>&gt;= 60 and &lt; 67</td>
<td>any</td>
<td>any</td>
<td>any</td>
<td>D</td>
</tr>
<tr>
<td>&lt; 60</td>
<td>any</td>
<td>any</td>
<td>any</td>
<td>F</td>
</tr>
</tbody>
</table>

**Final Exam:**

Again, the Final Exam for this course is scheduled for **Tuesday, December 15, 3:00 - 4:50 pm**, in **BSS 408** (unless I announce otherwise). Note this time and date **BEFORE** making your end-of-semester travel plans.

**Additional Grading-Related Policies:**

- Note that the project portion of your course grade includes all milestones, required pieces, required presentations, participation, and the final project itself.
  - Note, also, that evaluations of your effort by the instructor and by your teammates will potentially influence
the participation component of your project grade; these peer evaluations will be done several times during the semester.

- Note that the team approach for the course project is **not** optional.
  
  - Software engineering is not about working in isolation on an individual project, but rather includes how to function effectively as a member of a team developing a piece of complex software.
  
  - Part of what you will be learning, then, is how to deal with a team setting.
  
  - However, if a team member, in my judgment, is refusing to sufficiently participate in/contribute to the team’s project, for whatever reason, I reserve the right to remove that member from the team. Since one cannot pass this course without participation in a team project, such a member will not pass this course.

- If any student or team has serious issues with a team member, please bring them to my attention **as soon as possible**.

- (slightly adapted from Cashman and Eschenbach's Engi 111 Team Contract Guidelines): There will be **no illegal activity** during any team meetings or working sessions.
  
  - Illegal activity includes, but is not limited to, underage drinking and illicit drug use.
  
  - This rule must be enforced regardless of whether the meeting takes place on or off campus.
  
  - The team must notify me immediately (or by the next class meeting) if a violation of this rule occurs and the offending team member will be removed from the team and assigned a 0 for the entire course project.

- Because of the team-project nature of this course and because of the nature of some of the in-class activities, there is an attendance/participation component to the course grade (**in addition** to the clicker questions component and the participation component **within** the project's grades).
  
  - Sometimes (not always!) there will be class time provided for team meetings -- even when there is not, teams benefit from the chance to informally communicate, and of course there may be announcements or activities related to the team project. When you are not in a class session, then, you are adversely affecting your team.
  
  - Some portions of class sessions will include discussions of the chapters of *The Mythical Man Month* and other reading assigned for that session. The quality/benefit of these discussions is dependent on class member participation, and such participation cannot be “made up” in any real sense.
  
  - So -- attendance will be taken during the class sessions, and 10% of the course grade will be based on attendance as well as in-class participation.
  
  - For compelling reasons, each student may have up to three "excused" absences without penalty to their attendance/participation grade, but otherwise each **unexcused** absence from a class session will generally result in a loss of **10%** of the attendance/participation grade (except for extraordinary circumstances).

  Illness and life happen -- if you find you are going to have to miss class, please send me an e-mail as soon as you are able, including "CS 458 absence" in the subject line, containing the day you will miss/missed and a general explanation as to why. (I don't need deep detail, but please include the general nature of the reason for your absence.) It is the instructor's decision as to whether an absence is considered to be "excused" or not.

  - Note that arriving at a class session late or leaving a class session early may count as an absence or as a fraction of an absence.

  - I reserve the option to further lower the participation grade of any student attending, but not participating in, class discussions, team meetings, and other in-class activities.

- When part of a class session is made available for team meetings, it is important that you realize that such time is mandatory team meeting time, and meeting notes are usually required -- there will be information about this in the upcoming project handout.

  - It is **not** a time that you can leave early and run errands, it is **not** a time to work on coursework for other courses.
This project is extensive enough that each team should be able to find something to work cooperatively upon during class session time that is made available for team meetings.

Members who leave early or work on other classes' work should expect to lose points from their attendance/participation grade and from their project participation grade.

This course is labor intensive. Students should anticipate spending a significant amount of time outside of class sessions.

In a team-oriented course, it is vital that the teams meet as often as possible -- teams will need to meet more often than just during those parts of class sessions made available for team meetings.

Historically, the most successful teams have consistently cited regular outside-of-class-sessions team meetings, starting early in the semester, as an important key to their team's success.

### Course Expectations:

First: remember the general rule of thumb for college-level courses:

*To be successful in a course, you should plan to spend at least 3 hours outside of class for each 1 hour of college course credit. That implies an estimate of at least 12 hours a week spent outside of class for this 4-credit course.*

However, you should be aware that:

- This is a senior-level CS major course as well as a capstone course; it has an accordingly-rigorous workload.
- You are expected to attend every class session, and to do assigned readings in advance of class sessions. There will likely be reading questions for the assigned readings that must be answered prior to class sessions in which those readings are discussed.
- You are CS seniors -- you know at this point that programming can be a time-consuming activity.
  - You have hopefully also figured out that starting early (and not waiting until the last minute) gives you more time to ask questions with regard to homeworks, and gives you more time to collaborate with your team members with regard to the course project milestones.
- Past experience has shown that project teams that meet regularly outside of class throughout the semester learn more and create more successful, higher-quality projects.
- Homework and project milestone deadlines will not be extended because you waited too late to start or because you did not allocate enough time before the deadline to work on it; likewise, they will typically not be extended because of hardware or network failure. (Admittedly, campus failures might affect deadlines. But don't assume so until you have heard from me definitively.) You need to keep backups of your files at all times, and need to plan your schedule to be able to work on on-campus computers as necessary.
- If you have not completed a homework assignment or project milestone by the deadline, your best choice is to submit whatever you have managed to do by then, as partial credit is your friend, to carefully study the posted example solution (if the work in question is a homework assignment) as soon as it is available, to ask me about anything there that is still unclear, and to get a good early start on the next homework assignment or project milestone.

### A successful student in this class will:

- Carefully complete the assigned readings -- and, when applicable, their reading questions -- in advance of class sessions.
- Attend every class session, clicker at the ready.
- Participate in class (participating in class discussions, discussing clicker answers with other students, asking questions, paying attention, and taking notes).
- Attend and be an active, cooperative participant in regularly-scheduled outside-of-class course project team
meetings.

- Ask specific questions -- in class, in lab, in office-hours, and in e-mail.
- Work ahead on project milestones, and ask questions promptly as issues arise.
- Read through each homework assignment as soon as it is posted.
- Start working on each homework assignment early in the week.
- E-mail the instructor with specific homework-related questions starting early in the week both to clarify what a question is asking for and when hitting roadblocks (being sure to include both the code involved and any error messages or descriptions of bizarre behavior).
- Double-check the project handout before project milestones, and verify that the team is indeed submitting all of the specified pieces for each milestone.
- Always submit SOMETHING for a homework assignment or project milestone, even if it is not complete. (I believe in partial credit on homeworks, believing that if you have at least started working on a problem, the posted example solution will be more helpful/understandable than if you have not.)
- Compare their homework solutions to posted example solutions when they become available.
- Study with others for exams, and practice explaining concepts to one another.
- Attempt every exam problem, and carefully study over exams when they are returned.

**Academic Honesty:**

Students are responsible for knowing policy regarding academic honesty. For more information, visit: [http://www2.humboldt.edu/studentrights/academic-honesty](http://www2.humboldt.edu/studentrights/academic-honesty)

Observe that among the actions that are unacceptable are submitting another's program, code, or file as your own and failing to quote material (that includes algorithms, project, code, and comments, too!) taken from another person’s work. (Note that copying another student's comments is also unacceptable.)

For the team project, and possibly even for some of the course homeworks, it may be appropriate to make some use of code, tools, algorithms, etc. that you find on the web or in the computing literature. You are expected to clearly attribute such material.

Unless explicitly indicated otherwise on the homework assignment handout, typically homework assignment work is to be the work of each student, individually. If/when pair or group work is explicitly permitted for some homework assignment, the names of all students involved must be included on the work submitted. (For example, if pair programming is explicitly specified as being allowed for an assignment, then each pair-programmed file turned in will include both of the names of the students who worked on it as a pair.)

(Important aside: pair programming specifically means that two people sit at one computer, with one typing while the other says what to type. Both people are actively involved in the programming process. Pair-programming is not two people working at two computers, each doing different parts of the work individually. Pair-programming is also not one person doing all the work while the other does nothing or does something else. If pair-programming is ever explicitly permitted, then you are expected to actually pair-program any files you do not complete on your own.)

(If an assignment does explicitly specify that it is acceptable to pair program or work in groups, make sure that you don't get into the situation where you are merely watching someone else learn.)

For homework assignments (that are not explicitly specified as permitting pair-programming), students may discuss general approaches as long as no one involved in the discussion is writing anything down or typing anything during such discussions. Students may also help one another in determining causes of program bugs, or in determining the meaning of compiler error messages. However, in general, students may not work together to complete homework assignments, one student should not instruct another in how to write the code for a homework assignment, and any type of copying or modifying of another person's computer files, OR of providing computer files to another, related to homework assignments is definitely over the line, and never justified.
This applies to copying of documentation and comments as well as to copying of program code. Note that it is your responsibility to ensure that course assignment files are read-protected. If you are careless about this, and someone else copies your work, you will share the penalty. (In particular, be very careful about leaving work on shared network drives in campus labs, or in UNIX/Linux directories that are not read-protected.)

Learning takes hard work; when students turn in others' work as their own, it is a slap in the face to those seriously interested in learning. Not turning in an assignment results in no credit for that assignment, of course, but that is an honest grade. Work that violates the course honesty policy deserves a lower grade than that, and therefore the course policy is that work violating this policy will receive negative credit. A person providing a file for copying receives the same negative credit as the copier. Repeat offenses will be handled according to University policies.

**Asking Questions/Getting Help:**

- Sending questions by e-mail can be a very effective way to ask for help.
  - Include CS 458 along with the subject of your e-mail in the Subject: line of any class-related e-mail that you send me. This will help your e-mail be more recognizable as a class-related message, and will make it less likely that I will accidentally overlook it.
  - ALSO include a descriptive subject along with the CS 458 in that Subject: line -- this also increases the chances that I will notice and reply to your question more promptly. (In particular, do not just reply to a class e-mail message I have sent previously, and do not simply leave the Subject: line blank!)
  - That said, if I have not replied to your e-mail within 24 hours, please re-send it, just in case I did overlook it somehow.
  - You are expected to sign each e-mail you send me with your name -- sometimes the sender's identity is not obvious from one's e-mail address, especially for an off-campus e-mail address.
  - Also, DON'T INCLUDE the word "password" in your e-mails to me -- pwd is a handy abbreviation to use instead -- because, due to phishing scams, HSU's spam filtering does not seem to like e-mails with that word in it! (Odd, but this was definitely the case in Spring 2010...)

- I try to check my e-mail (st10@humboldt.edu or sharon.tuttle@humboldt.edu or smtuttle@humboldt.edu) about once a day on weekdays, and about once over each weekend. This is another reason to start assignments early, so that you have time to receive a reply to any questions that might arise.

- You are encouraged to ask me questions in class, in office hours, and by e-mail. The most successful students are those who are not afraid to ask questions early and often (I will gently let you know if you are overdoing it), who do the assigned reading, who attend lecture and lab regularly, who start homeworks promptly after they are made available from the course web page, and who practice course concepts as much as possible, who meet with their project teams regularly, and who start project milestones early.
  - It is better to ask a question sooner than later -- for example, it is better to send an e-mail with a specific question as soon as you think of it than it is to wait a day or two until the next class meeting or office hour. If you wait to ask such questions, you may not have time to complete the assignment or project milestone before its deadline.
  - It is perfectly reasonable if you send me a question and then end up finding out the answer yourself before you receive my answer; likewise, it is not a problem if you end up sending me several questions in separate e-mails (as you work on different parts of a homework or project milestone while awaiting earlier answers).

- That said, I am expecting that you will ask specific questions – overly vague or broad questions are problematic.
  - (For example, an example of a specific question is, "When I try to run the method: (paste in the method), I receive the following error message: (paste in the error message) Can you point me in the right direction about what is wrong?" An example of an overly vague or broad question is: "Here's my code/user stories/testing plan/etc. Is it right?")
Additional Coursework-Related Policies:

• You should expect to put in a significant amount of time outside of lecture doing the assigned reading and answering the reading questions, meeting with your project team, working on homework assignments and project milestones, and practicing concepts discussed.

• Each homework assignment and project milestone must be submitted as is specified on its handout to be accepted for credit. This may vary for different assignments and milestones. Often, parts of assignments and milestones will be submitted using a special tool on nrs-labs or nrs-projects (or possibly using git or github).

• Each homework assignment and project milestone will be clearly marked with one or more due dates (a single assignment could have multiple parts with multiple due dates).
  – **No homework assignments or project milestones will be accepted late.** If you wish to receive any credit for a homework assignment or project milestone, then you must turn in whatever you have done, even if it is incomplete, by the deadline. **Partial credit is usually preferable to no credit.** Note that "the computer/network/etc. going down" is no excuse --- if you leave an assignment or milestone for the last minute and there are technical problems, you still must turn in whatever you have by the deadline. As with any work done on computer, make frequent back-ups of your files!
  – You may submit multiple versions of assignment and milestone files before the deadline; I will grade the latest pre-deadline submission unless you inform me otherwise. This is to encourage you to turn parts in early (since you will know that you can always turn in an improved version if further inspiration strikes). You also don't have to worry about forgetting to submit something that has already been submitted!
  – If for any reason you cannot submit course work using the submission tool on nrs-projects (or as specified by the assignment), e-mail me your files as attachments by the deadline, and then submit the files using the submission tool (or other specified means) as soon as you are able. The e-mailed files will establish that these files were completed by the deadline.

• The nrs-projects tool that you will be using to submit some assignment and milestone parts results in a file that serves as your "receipt" for having submitted items. You are expected to retain these "receipt" files at least until a grade has been posted to the course Moodle site for that assignment or milestone. If there is a system glitch or other hardware/software/network problem, you may be asked to make me a copy of one or more receipt files; if you do not have them, then you will not receive credit for the files involved. These receipt files are for your protection!

• It is nearly impossible to write unambiguous specifications. If you have questions about "what she means", get them resolved very early in the development cycle by asking.

• There is more to computer code than simply whether it runs or not...
  – Part of your grade will be determined by how well your work meets the written requirements. Work that you turn in is expected to meet handout specifications precisely; when one eventually works within a team on large projects, following the specifications precisely is vital, and can mean the difference between a working product and one that just sits there.
  – Note that work may be graded on style as well as on whether it runs properly and whether it precisely meets the homework specifications and requirements. Discussions on style will be ongoing throughout the semester.

• Some course work may be graded simply based on whether it has been attempted (the instructor's decision is final as to whether this is the case) -- other course work may be graded for correctness, style, and whether it meets specifications. You will not know in advance which will be the case.

Incompletes:

Incompletes are rarely given and only in the case of a true emergency. They certainly are not appropriate for students who find they have fallen behind on assignments, missed a test, or taken on too much academic, work, or family responsibilities. For these situations, dropping the course would be appropriate (if that is still possible
according to the University policies for dropping courses).

**Additional Course Policies:**

- You are expected to read this syllabus and be prepared to verify in a required Moodle activity that you have received it, have read it, and understand its contents.

- Exam dates are given in the course schedule below. Make-up exams are only possible by special prior arrangement or because of a valid medical excuse.

- You should monitor your e-mail for course-related messages. The University provides a means for you to specify your preferred e-mail address, so if you wish to receive e-mail into an account other than the one HSU provides, change your preferred e-mail address in both Account Center and Moodle accordingly. Course-related messages from me will include CS 458 in the Subject: line.

- You are expected to check the public course web page and the course Moodle site regularly -- course handouts, homework assignments, examples from lectures and labs, and possibly more will be posted to the public course web page, and grades will be posted to the course Moodle site. You are expected to monitor your posted grades and let me know about any discrepancies.

- When reading assignments are given, you are expected to prepare (read and study) assigned readings before class and to participate in class discussions. Projected examples will be utilized frequently during discussion. You should understand that there may be material in the reading that will not be discussed in lecture/lab, and material in the lectures/labs that may not be found in the reading. You are responsible for both.

**Campus policies:**

**Students with Disabilities:**

Persons who wish to request disability-related accommodations should contact the Student Disability Resource Center in the Learning Commons, Lower Library, 826-4678 (voice) or 826-5392 (TDD). You can reach the Student Disability Resource Center's web site at:

[http://www.humboldt.edu/disability/](http://www.humboldt.edu/disability/)

Please note that some accommodations may take up to several weeks to arrange. If you are eligible for such accommodations, please contact me as soon as possible to discuss them.

**Add/Drop Policy:**

Students are responsible for knowing the University policy, procedures, and schedule for dropping or adding classes. You can find these on the web at:

[http://www.humboldt.edu/registrar/students/regulations/schedadjust.html](http://www.humboldt.edu/registrar/students/regulations/schedadjust.html)

You can find the University policies for repeating classes at:

[http://www.humboldt.edu/registrar/students/regulations/repeat.html](http://www.humboldt.edu/registrar/students/regulations/repeat.html)

Note the following highlights from the above CSU (and thus HSU) policies on withdrawing from and repeating courses:

- Students may withdraw from no more than 18 semester-units after the first four weeks of instruction; that is, students may withdraw from no more than 18 semester-units between census and the final 20% of instruction, and only then with a serious and compelling reason. (Note that: "Withdrawal from courses for reasons that are catastrophic, such as accident or serious illness do not count toward the 18-unit limit." [from the Registrar's web site])

- Students may repeat courses for grade forgiveness only if they earned grades lower than a C.

- Students may repeat up to 16 semester-units with grade forgiveness.
• Students may only repeat a course for grade forgiveness two times and each of these attempts counts toward the 16-unit maximum for repeats.

• Students may repeat up to an additional 12 semester-units with grades averaged.

Please note - it is the Registrar's Office that determines what constitutes a "serious and compelling reason".

The census date for Fall 2015 (before which you can drop without a W, and without it counting toward your 18 semester-units drop limit) is: **Monday, September 21st**.

The last date for Spring 2015 to drop with a W on your transcript, with a serious and compelling reason, and subject to the 18 semester-unit drop limit, is: **Monday, November 2nd**.

If you do drop the course, note that it is **your responsibility** to complete and submit the appropriate forms.

**Attendance and disruptive behavior:**

Students are responsible for knowing policy regarding attendance and disruptive behavior:

http://www2.humboldt.edu/studentrights/attendance-behavior

• **Late arrival to class:** Please attempt to come to class on time, with your headphones/earbuds/etc. put away and your cell phones/tablets/pads/gadgets/etc. turned off. If you must arrive late or leave early, please do so with the least possible distraction to other students. If your late/early habits become disruptive, you may be asked to leave the class permanently.

• **Class disruption:** University policy requires that instructors eliminate disruptions to the educational process. Distractions such as excess talking, ringing cell phones, working on assignments for other classes, inappropriate or distracting laptop/tablet/smartphone/gadget use, demonstrations of affection, packing of books early, loud music leaking from headphones, chronic late arrivals or early departures, excessive comings and goings or other behaviors that disrupt the class are not acceptable. Students indulging in such behaviors will first be warned before being required to leave the class permanently.

**Emergency Evacuation**

Please review the evacuation plan for the classroom (posted on the orange signs), and review the campus Emergency Preparedness web site at:

http://www.humboldt.edu/emergencymgmtprogram/campus_emergency_preparedness.php

...for information on campus Emergency Procedures. During an emergency, information regarding campus conditions can be found at **826-INFO** or:

http://www.humboldt.edu/emergency

**VERY Tentative Course Schedule: (VERY subject to change!)**

...is coming soon.

**Final Exam:**

TUESDAY, December 15, 3:00 - 4:50 pm, in BSS 408 (unless I announce otherwise)