CSE 111 Problem-Solving through Math and Quantitative Reasoning

Instructor
Bina Ramamurthy
345 Davis Hall
bina@buffalo.edu
716-645-3182

4 credits: 3 hour lecture, 1 hour recitation.

Teaching Assistants
To be determined

Course Meeting Days, Times, and Locations
MWF 12.00-12.50PM, NSC 201

Course Description

Students study algorithmic problem-solving techniques and gain an appreciation for some of the most interesting and significant results of computer science, as well as its intellectual and social significance. The course has a computational, mathematical and a laboratory component. This course involves solving real world problems using computers. Topics include: Problem statement and analysis, collecting data, analyzing data and presenting the results. More specifically, students will learn HTML5 for structuring information for communication, CSS for formatting and styling the presentation, and javascript (JS) for modeling behaviors driven by data analytics and computation. The course will also cover web standards, the importance of it, and the accessibility of information to all. Admitted computer science and computer engineering majors should not enroll in this course sequence.

Prerequisite: There are no formal prerequisites for this course, but to be successful in the course, students will have to have a solid background in high school level Mathematics, including algebra. In addition, if you have never used a computer before or do not feel comfortable using web browsers, email clients, instant messaging programs, word processors, or spreadsheets and would like a general introduction to these type of concepts, this is not the course for you.

Course Requirements

The lecture class will meet weekly for 3 hours and class time will be spent problem solving, understanding the Math involved in the problems and apply quantitative reasoning to solve the problems and implement the solutions using hands-on programs in a modern high-level language. Grading will be based on three types of work: individual labs (5) where problems are solved (as discussed above), solutions designed and implemented, quizzes (4), and exams (3).
Course Objectives

At the end of this course, each student should better understand problems solving using Math and Quantitative Reasoning (QR). Students will also have knowledge about programming in a high level language, skills, concepts, and capabilities associated with Information Technology, fundamentals of hardware, software, human-computer interfaces, and many modern internet based technologies.

<table>
<thead>
<tr>
<th>Requirement</th>
<th>Quantity</th>
<th>Topics Addressed</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>Quiz</td>
<td>4</td>
<td>Format: multiple choice; to assess learning continuously</td>
<td>Once in 3 weeks</td>
</tr>
<tr>
<td>Exam</td>
<td>3</td>
<td>Format: Written exam; to assess the thought process and application of Math and quantitative reasoning to problem solving.</td>
<td>Once in 4 weeks;</td>
</tr>
<tr>
<td>Lab</td>
<td>5</td>
<td>Format: detailed written description of a real-world problem is provided. Students apply Math, QR and programming skills to design and implement a working solution for the problem.</td>
<td>Once in 2-3 weeks</td>
</tr>
</tbody>
</table>

Course Learning Outcomes

The following table lists learning outcomes for this course. Upon completing this course, students will be able to

<table>
<thead>
<tr>
<th>Course Learning Outcome</th>
<th>Maps to the Following Program Outcomes / Competencies:</th>
<th>Delivered through the Following Instructional Method(s):</th>
<th>Student Achievement Assessed with the Following Method(s)/Assignments:</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Analyze data and apply empirical or theoretical methods to guide decision-making.</td>
<td>UBGE, SUNY Mathematics, MSCHE Quantitative Reasoning, MSCHE Critical Analysis &amp; Reasoning</td>
<td>5 or 6 hands-on lab problems involving problem solving, reasoning and programming.</td>
<td>Labs grades, quiz and exam grades.</td>
</tr>
<tr>
<td>2. Interpret mathematical models, formulas, graphs, and tables, and draw inferences from them.</td>
<td>UBGE, SUNY Mathematics, MSCHE Quantitative Reasoning, MSCHE Critical Analysis &amp; Reasoning</td>
<td>Hands-on demo and active learning that happens during lecture.</td>
<td>2 to 4 Quiz grades and 3 exam grades.</td>
</tr>
</tbody>
</table>
### 3. Choose appropriate models for a given problem, using information from observed or deduced data and knowledge of the system being studied.

**UBGE, SUNY Mathematics, MSCHE Quantitative Reasoning, MSCHE Critical Analysis & Reasoning**

Solving problems using modern a programming language. The solution involves developing rigorous algorithms and them implementing them in a modern programming language.

2 to 4 Quiz grades and 3 exam grades.

### 4. Employ quantitative methods, mathematical models and/or statistics to develop well-reasoned arguments to identify and solve real world problems beyond the level of basic algebra, while also being able to recognize the limitations of mathematics and statistics.

**UBGE, SUNY Mathematics, MSCHE Quantitative Reasoning, MSCHE Critical Analysis & Reasoning**

Analyze problems and develop solutions through quantitative reasoning. Reasoning involves mathematics and decision making in computation of results. These results are obtained through programs written by the students using modern programming languages. Some of the problems they work on include analyzing a datasets to draw inferences and conclusions about the data (such as performing a statistical analysis of the data) and “games” that require math and quantitative reasoning.

Lab grades, quiz grades and exam grades.

### 5. Recognize common mistakes in empirical and deductive reasoning, and mathematical and

**UBGE, SUNY Mathematics, MSCHE Quantitative Reasoning, MSCHE Critical Analysis & Reasoning**

Debugging or recognizing and correcting errors or bugs are critical steps in computer programming. This is

Labs grades, exam and quiz grades.
<table>
<thead>
<tr>
<th>Course Learning Outcome</th>
<th>Program Outcomes*</th>
<th>Assessment Methods</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 Organize content into semantically equivalent units</td>
<td>d</td>
<td>Lab1</td>
</tr>
<tr>
<td>2 Digitize data in various forms and analyze number systems</td>
<td>b,e,i,j</td>
<td>Exam1</td>
</tr>
<tr>
<td>3 Interpret information and categorize them into coherent units (structure, style, logic) for representation</td>
<td>a,f,g</td>
<td>Lab2, Quiz 1</td>
</tr>
<tr>
<td>4 Represent data (numbers), choose appropriate models, and solve real-world problem</td>
<td>b,c</td>
<td>Lab 3, Quiz 2</td>
</tr>
<tr>
<td>5 Represent data in logical structures, evaluate it to make choices, and program with event-driven data</td>
<td>h,i,j</td>
<td>Lab3, Exam 2</td>
</tr>
<tr>
<td>6 Apply logical thinking and decision making in the design and development (of an online game)</td>
<td>g, h</td>
<td>Lab4, Quiz 3</td>
</tr>
<tr>
<td>7 Collect data for analysis and interpretation</td>
<td>c,f</td>
<td>Lab 5</td>
</tr>
<tr>
<td>8 Visualize results of data analysis to communicate results</td>
<td>a,c,d</td>
<td>Exam 3, Quiz 4</td>
</tr>
<tr>
<td>9 Analyze data and apply empirical or theoretical methods to guide decision-making.</td>
<td>b,c,f</td>
<td>Labs grades, quiz and exam grades.</td>
</tr>
<tr>
<td>10 Employ quantitative methods, mathematical models and/or statistics to develop well-reasoned arguments to identify and solve real world problems.</td>
<td>b,g</td>
<td>Labs, exams and quiz</td>
</tr>
<tr>
<td>11 Recognize common mistakes in empirical and deductive reasoning, and mathematical and quantitative problem solving.</td>
<td>k</td>
<td>Lab programs</td>
</tr>
<tr>
<td>12 Express inferences and conclusions in writing.</td>
<td>d</td>
<td>Lab reports and documentation</td>
</tr>
</tbody>
</table>
a) **Interpretation**, i.e., the ability to explain information presented in mathematical forms [e.g., equations, graphs, diagrams, tables, words],
b) **Calculation**: Ability to compute results given a set of input data.
c) **Application/Analysis**, i.e., the ability to make judgments and draw appropriate conclusions based on the quantitative analysis of data, while recognizing the limits of this analysis. This is accomplished by problem solving using high level languages.
d) **Communication**, i.e., expressing quantitative evidence in support of the argument or purpose of the work [in terms of what evidence is used and how it is formatted, presented, and contextualized].
e) **Confidence with Mathematics**. Being comfortable with quantitative ideas and at ease in applying quantitative methods.
f) **Interpreting Data**. Reasoning with data, reading graphs, drawing inferences, and recognizing sources of error.
g) **Logical Thinking**. Analyzing evidence, reasoning carefully, understanding arguments, questioning assumptions, detecting fallacies, and evaluating risks.
h) **Making Decisions**. Using mathematics to make decisions and solve problems in everyday life. For individuals who have acquired this habit, mathematics is not something done only in mathematics class but a powerful tool for living, as useful and ingrained as reading and speaking.
i) **Mathematics in Context**. Using mathematical tools in specific settings where the context provides meaning. Notation, problem-solving strategies, and performance standards all depend on the specific context.
j) **Number Sense**. Having accurate intuition about the meaning of numbers, confidence in estimation, and common sense in employing numbers as a measure of things. Practical Skills.
k) **Debugging or recognizing and correcting** errors or bugs are critical steps in computer programming.

*Reference:* Adapted from Quantitative literacy learning Rubric of AAC&U available at [http://d32ogoqmyal8w.cloudfront.net/files/NICHE/quantitative_literacy_rubric.pdf](http://d32ogoqmyal8w.cloudfront.net/files/NICHE/quantitative_literacy_rubric.pdf)

**Textbook and/or Other Required Materials**
   by Jennifer Niederst Robbins (Author)
   Publisher: O'Reilly Media; Fourth Edition edition (August 24, 2012)
   Language: English
   ISBN-10: 1449319270
2. Problem statements for data analysis will be provided by the instructor along with the lab handouts.

**Course Organization**
The course has both a lecture component and a recitation component. Each component plays a role in helping a student achieve the objectives of the course. In order do well in this course, a student needs to participate in both components.
Lectures
Conceptual and theoretical course content will be delivered primarily in the lectures, complemented by readings from the text books. A student must review readings prior to attending a lecture, and is expected to review the readings again, along with any notes he/she took, after the lecture. Some of the topics will be difficult. It is therefore absolutely essential that students ask questions whenever something is said which he/she does not understand.

Recitations
The recitations are an integral part of the course. In each recitation section, the instructor will answer questions about the currently assigned programming assignments (called labs). Students are free to ask any questions about the labs and get assistance from the instructor during this time on your personal assignment. Students are free to work on your assignments during this time. It is possible that students will be able to finish your lab work during the recitation time. However, it is also possible that students will need to work outside of lab time to finish the assignments.

Recitations do not meet on the first week of classes.

<table>
<thead>
<tr>
<th>Code</th>
<th>Section</th>
<th>Days</th>
<th>Time</th>
<th>Location</th>
</tr>
</thead>
<tbody>
<tr>
<td>10468</td>
<td>A1</td>
<td>M</td>
<td>9:00 AM - 10:50 AM</td>
<td>Capen 201A</td>
</tr>
<tr>
<td>10472</td>
<td>A2</td>
<td>M</td>
<td>3:00 PM - 4:50 PM</td>
<td>Capen 201A</td>
</tr>
<tr>
<td>10475</td>
<td>A3</td>
<td>T</td>
<td>5:00 PM - 6:50 PM</td>
<td>Capen 201A</td>
</tr>
<tr>
<td>10543</td>
<td>A4</td>
<td>F</td>
<td>8:00 AM - 9:50 AM</td>
<td>Capen 201A</td>
</tr>
<tr>
<td>10547</td>
<td>A5</td>
<td>R</td>
<td>12:00 PM - 1:50 PM</td>
<td>Capen 201A</td>
</tr>
<tr>
<td>10485</td>
<td>A6</td>
<td>T</td>
<td>12:00 PM - 1:50 PM</td>
<td>Capen 201A</td>
</tr>
</tbody>
</table>

Course Evaluation
The following indicates the grade breakdown which will be used in assigning grades in the course. The right is reserved to make small adjustments to the breakdown if it is necessary.

- **Exam component (60% of final course grade)**
  There will be two mid-term exams and a comprehensive final exam. No make-up examination will be available.
  - Best score of the exam1 and exam2 along with that of the mandatory final exam will used in computation of the exam component of the overall grade.
- **Quiz component (10% of final course grade)**
  This will involve (i) attendance and (ii) quizzes (at least 4) during the class time. More information about this will be explained in lecture. Quizzes will be good review for exams.
  - Best three quizzes will be used in the computation of the class participation grade.
- **Lab Programming Assignments (30% of final course grade)**
  There will be 5 programming labs assigned throughout the semester.
  - Best 3 lab scores of the first 4 labs along with that of the mandatory lab 5 will be used in computation of the Programming assignment/lab component of the overall grade.

- **Letter Grades (tentative)** The following table indicates the number to letter grade mapping that will be used to assign final grades at the end of the course. The final overall grade will be curved relative to the overall performance of the class.

<table>
<thead>
<tr>
<th>Score %</th>
<th>95 –100</th>
<th>90 –95</th>
<th>85 –89</th>
<th>80 –84</th>
<th>75 –79</th>
<th>70 –74</th>
<th>65 –69</th>
<th>60 –64</th>
<th>55 –59</th>
<th>50 –54</th>
<th>0-49</th>
</tr>
</thead>
<tbody>
<tr>
<td>Grade</td>
<td>A</td>
<td>A-</td>
<td>B+</td>
<td>B</td>
<td>B-</td>
<td>C+</td>
<td>C</td>
<td>C-</td>
<td>D+</td>
<td>D</td>
<td>F</td>
</tr>
</tbody>
</table>
Re-grading
If a student has a question about the grading of any piece of work, he/she should consult with his/her recitation TA. Any questions about the grading of a piece of work must be raised within one week of the date that the work was returned by the instructor. In other words, if a student does not pick up his/her work in a timely fashion; he/she may forfeit your right to question the grading of his/her work.

Incomplete (I) Grades
We will follow the UB Undergraduate Catalog Statement on Incomplete Grades, found in the Undergraduate Catalog. University and department policy dictate that "I" grades can be given only if the following conditions are met:

- An Incomplete will only be given for missing a small part of the course.
- An Incomplete will only be given when the student misses work due to circumstances beyond his/her control.
- An Incomplete will only be given when the student is passing the course except for the missed material.
- An Incomplete is to be made up with the original course instructor within the time specified by the appropriate University regulation, and usually within the following semester.
- An Incomplete will not be given to allow the student to informally retake the entire course, and have that grade count as the grade of the original course.

Incompletes cannot be given as a shelter from poor grades. It is your responsibility to make a timely resignation from the course if you are doing poorly for any reason.

Disabilities
If you have a diagnosed disability (physical, learning, or psychological) that will make it difficult for you to carry out the course work as outlined, or that requires accommodations such as recruiting note-takers, readers, or extended time on exams or assignments, you must consult with the Office of Disability Services (25 Capen Hall, Tel: 645-2608, TTY: 645-2616, Fax: 645-3116, http://www.student-affairs.buffalo.edu/ods/). You must advise your instructor during the first two weeks of the course so that we may review possible arrangements for reasonable accommodations.

Counseling Center
Your attention is called to the Counseling Center (645-2720), 120 Richmond Quad. The Counseling Center staff are trained to help you deal with a wide range of issues, including how to study effectively and how to deal with exam-related stress. Services are free and confidential. Their web site is http://www.studentaffairs.buffalo.edu/shs/ccenter/

Distractions in the Classroom - Behavioral Expectations
The following is the text of a policy adopted by the Faculty Senate. Students are expected to know and adhere to this policy. The Student Conduct Regulations. Classroom "etiquette" expectations should include:

- Attending classes and paying attention.

- Not coming to class late or leaving early. If you must enter a class late, do so quietly and do not disrupt the class by walking between the class and the instructor. Do not leave class unless it is an absolute necessity.

- Not talking with other classmates while the instructor or another student is speaking.
• If you have a question or a comment, please raise your hand, rather than starting a conversation about it with your neighbor.
• Showing respect and concern for others by not monopolizing class discussion. Allow others time to give their input and ask questions. Do not stray from the topic of class discussion.
• Avoiding audible and visible signs of restlessness. These are both rude and disruptive to the rest of the class.
• Focusing on class material during class time. Sleeping, talking to others, doing work for another class, reading the newspaper, checking email, and exploring the internet for other than course work are unacceptable and can be disruptive.

Academic Integrity
Source: http://www.cse.buffalo.edu/academics-academic-integrity.shtml
The academic degrees and the research findings produced by our Department are worth no more than the integrity of the process by which they are gained. If we do not maintain reliably high standards of ethics and integrity in our work and our relationships, we have nothing of value to offer one another or to offer the larger community outside this Department, whether potential employers or fellow scholars. For this reason, the principles of Academic Integrity have priority over every other consideration in every aspect of our departmental life, and we will defend these principles vigorously. It is essential that every student be fully aware of these principles, what the procedures are by which possible violations are investigated and adjudicated, and what the punishments for these violations are. Wherever they are suspected, potential violations will be investigated and determinations of fact sought. In short, breaches of Academic Integrity will not be tolerated.

Departmental Statement on Academic Integrity in Coding Assignments and Projects
All academic work must be his/her own. Plagiarism, defined as copying or receiving materials from a source or sources and submitting this material as one's own without acknowledging the particular debts to the source (quotations, paraphrases, basic ideas), or otherwise representing the work of another as one's own, is never allowed. Collaboration, usually evidenced by unjustifiable similarity, is never permitted in individual assignments. Any submitted academic work may be subject to screening by software programs designed to detect evidence of plagiarism or collaboration.

It is the student’s responsibility to maintain the security of his/her computer accounts and his/her written work. Students should (i) not share passwords with anyone, nor write the password down where it may be seen by others, (ii) not change permissions to allow others to read his/her course directories and files and (iii) not walk away from a workstation without logging out. In groups that collaborate inappropriately, it may be impossible to determine who has offered work to others in the group, who has received work, and who may have inadvertently made their work available to the others by failure to maintain adequate personal security. In such cases, all will be held equally liable. These policies and interpretations may be augmented by individual instructors for their courses.

Departmental and Course Policy on Violations of Academic Integrity
If, after following the procedures required by the University for investigation of suspected breaches of academic integrity, a student is found guilty, the policy of the department of Computer Science & Engineering is that the student minimally receive a grade of F in the course.