CSE215, Foundations of Computer Science
Course Information

Summer 2020
Stony Brook University
Instructor: Dr. Paul Fodor

http://www.cs.stonybrook.edu/~cse215
Course Description

• “Introduction to the logical and mathematical foundations of computer science. Topics include functions, relations, and sets; recursion and functional programming; elementary logic; and mathematical induction and other proof techniques.”

• This IS NOT a course in computer programming, BUT on fundamental concepts of computing.

• We will stress mathematical problem solving skills and the use of formal concepts as tools for computer science.

• Prerequisites: AMS 151 or MAT 125 or MAT 131.
Course Outcomes

The following are the official course goals agreed upon by the faculty for this course:

- An ability to define and use discrete structures such as functions, relations, and sets.
- An ability to compute with recursion as a basic paradigm.
- An ability to use logic and basic proof techniques, such as mathematical induction.
What is Computer Science?

- Why do we study mathematics and problem solving in a major course in Computer Science?
  - Computer Science is NOT computer programming - although programming is part of it.
  - Computer Science is a **mathematical science** we study the capabilities and limitations of computers and how people can use them effectively.
  - Computer programming requires that the exact sequence of steps to perform a task must be specified completely and precisely
    - difficult and requires careful reasoning about **abstract entities**
  - **Mathematics has developed over thousands of years as a method of abstract reasoning.**
Why Isn't CS “Just Programming”?

• Programs of only a few hundred lines are easy for one person to build with little training.

• BUT:
  • Real-world software systems are **large**
    • Developing and understanding such complicated objects requires mental and mathematical discipline.
  • Real-world software systems must be **reliable**
    • They control economies, airplanes, nuclear weapons and your car
    • **Systematic** discipline is necessary to avoid errors

• Mathematics provides the disciplined and systematic language to reason about such systems.
General Information

- **Blackboard** will be used for assignments, grades and course material.

- **Staff:**
  - **Instructor:** Dr. Paul Fodor
  - **214 New Computer Science Department, Stony Brook University**
  - **Email:** paul.fodor@stonybrook.edu

- **Class Time and Place**
  - **Lecture:** Tuesdays and Thursdays 9:00AM - 12:25PM, Online.
  - **Recitation:** Tuesdays and Thursdays 12:30PM - 1:25PM, Online.
Textbook

- Discrete Mathematics: Introduction to Mathematical Reasoning
  
  Author: Susanna S. Epp
  
  Publisher: Brooks Cole; 1st edition (2011)
  
  ISBN-10: 0495826170
  
Grading

- Grades will be based on homework and exams according to the following formula:
  - Homeworks -- 25%
  - Midterm exams (2) -- 50% (25% each)
  - Final exam -- 25%
    - Do not miss the exams. Make-up exams will be given only in extenuating circumstances (e.g., doctor's note stating that you were ill and unfit to take the exam). Students who miss an exam for a valid reason may need to take a make-up exam; specific arrangements will be made on a case-by-case basis.

- Grade Cutoffs:
  - A [95-100], A- [90-95], B+ [87-90], B [83-87], B- [80-83], C+ [77-80], C [73-77], C- [70-73], D+ [65-70], D [60-65], F [0-60]

- SPECIAL RULE: If all your grades, including homework assignments, quizzes, recitation and your three exam grades are above the respective class averages, you're guaranteed to receive a grade of C or higher for this class.
Grading

- The Pass/No Credit (P/NC) option is not available for this course.
  - This policy applies to all CSE/ISE undergraduate courses used to satisfy the graduation requirements for the major.

- Exam dates:
  - Midterm exam 1: Tu. 7/21, on Respondus Lockdown Browser with Monitoring.
  - Midterm exam 2: Th. 7/30, on Respondus Lockdown Browser with Monitoring.
  - Final exam: Th. 8/13, on Respondus Lockdown Browser with Monitoring.

- The grades will be posted on Blackboard: http://blackboard.stonybrook.edu for privacy reasons.
Homework

- There will be homework assignments given regularly.
- The homework assignments are to be completed individually in the allotted time.
- No Late Submissions Are Allowed.
- No makeup homework will be given.
- The homework assignments will be posted on Blackboard: http://blackboard.stonybrook.edu.
Regrading of Homework/Exams

- Please meet with a TA or the instructor and arrange for regrading.
- You have one week from the day grades are posted or mailed or announced.
- Late requests will not be entertained.
## Tentative Class Schedule

<table>
<thead>
<tr>
<th>Week</th>
<th>Date</th>
<th>Lecture Topics / Notes</th>
<th>Readings</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Tu. 7/7</td>
<td>Administrative (course information and introduction to speaking mathematically), The Logic of Compound Statements</td>
<td>Read Epp chs. 1 and 2, and <em>Introduction to LaTeX</em></td>
</tr>
<tr>
<td></td>
<td>Th. 7/9</td>
<td>The Logic of Compound Statements: Logical arguments, The Logic of Quantified Statements, Supplemental: Application of Logic - Digital Circuits</td>
<td>Read Epp ch. 3</td>
</tr>
<tr>
<td>2</td>
<td>Tu. 7/14</td>
<td>Elementary Number Theory and Methods of Proof</td>
<td>Read Epp ch. 4</td>
</tr>
<tr>
<td></td>
<td>Th. 7/16</td>
<td>Sequences and Mathematical Induction</td>
<td>Read Epp ch. 5</td>
</tr>
<tr>
<td>3</td>
<td>Tu. 7/21</td>
<td>MIDTERM EXAM 1</td>
<td>see Blackboard</td>
</tr>
<tr>
<td></td>
<td>Th. 7/23</td>
<td>Set Theory</td>
<td>Read Epp ch. 6</td>
</tr>
<tr>
<td>4</td>
<td>Tu. 7/28</td>
<td>Functions</td>
<td>Read Epp ch. 7</td>
</tr>
<tr>
<td></td>
<td>Th. 7/30</td>
<td>MIDTERM EXAM 2</td>
<td>see Blackboard</td>
</tr>
<tr>
<td>5</td>
<td>Tu. 8/4</td>
<td>Recursion</td>
<td>n/a</td>
</tr>
<tr>
<td></td>
<td>Th. 8/6</td>
<td>Functional Programming (ML)</td>
<td>Standard ML</td>
</tr>
<tr>
<td>6</td>
<td>Tu. 8/11</td>
<td>Relations</td>
<td>Read Epp ch. 8</td>
</tr>
<tr>
<td></td>
<td>Th. 8/13</td>
<td>FINAL EXAM 2</td>
<td>see Blackboard</td>
</tr>
</tbody>
</table>

(c) Paul Fodor (CS Stony Brook)
Academic Integrity

• You can discuss general assignment concepts with other students: explaining how to use systems or tools and helping others with high-level design issues

• You **MAY NOT share** assignments, source code or other answers by copying, retyping, looking at, or supplying a file

• Assignments are subject to manual and automated similarity checking (**We do check!** and our tools for doing this are much better than cheaters think)

• If you cheat, you will be brought up on academic dishonesty charges - we follow the university policy:
  • [http://www.stonybrook.edu/uaa/academicjudiciary](http://www.stonybrook.edu/uaa/academicjudiciary)
Disability

- If you have a physical, psychological, medical or learning disability, contact the SACS office at phone 631-632-6748

- All documentation of disability is confidential.
What do you need to get started?

- Blackboard account
  - [http://blackboard.stonybrook.edu](http://blackboard.stonybrook.edu)
Variables

- Is there a number with the following property: doubling it and adding 3 gives the same result as squaring it?
  - In this sentence you can introduce a variable to replace the potentially ambiguous word “it”: Is there a number \( x \) with the property that \( 2x + 3 = x^2 \)?
  - A variable is a temporary name until we can find the possible value(s)

- No matter what number might be chosen, if it is greater than 2, then its square is greater than 4.
  - A variable is a temporary name to the (arbitrary) number you might choose enables you to maintain the generality of the statement: No matter what number \( n \) might be chosen, if \( n \) is greater than 2, then \( n^2 \) is greater than 4.
Some Important Kinds of Mathematical Statements

- **Universal conditional statement**: For all animals \( a \), if \( a \) is a dog, then \( a \) is a mammal.
- **Universal existential statement**: Every real number has an additive inverse.
- **Existential universal statement**: There is a positive integer that is less than or equal to every positive integer.
Sets

- Introduced in 1879 by Georg Cantor (1845–1918).
- A set is, intuitively, a collection of elements.

**Set-Roster Notation:**
- Let \( A = \{1, 2, 3\} \), \( B = \{3, 1, 2\} \), and \( C = \{1, 1, 2, 3, 3, 3\} \).
  - What are the elements of \( A \), \( B \), and \( C \)?
  - How are \( A \), \( B \), and \( C \) related?

**Set-Builder Notation:**
\[
\{ x \in \mathbb{R} \mid -2 < x < 5 \}
\]

- Subset: is a basic relation between sets: \( \{2\} \subseteq \{1, 2, 3\} \)
Cartesian product

- Example: $\mathbb{R} \times \mathbb{R}$ is the set of all ordered pairs $(x,y)$ where both $x$ and $y$ are real numbers.

- Cartesian plane:
Relations

• The notation $x \, R \, y$ as a shorthand for the sentence “$x$ is related to $y$”, for example: $1 < 2$

• From relations to sets: $x \, R \, y$ means that $(x, y) \in R$

• Arrow Diagrams of Relations:
Functions

- **Definition**

A **function** $F$ **from a set** $A$ **to a set** $B$ is a relation with domain $A$ and co-domain $B$ that satisfies the following two properties:

1. For every element $x$ in $A$, there is an element $y$ in $B$ such that $(x, y) \in F$.
2. For all elements $x$ in $A$ and $y$ and $z$ in $B$,
   
   \[ \text{if } (x, y) \in F \text{ and } (x, z) \in F, \text{ then } y = z. \]

Example: The **successor function** $g$ from $\mathbb{Z}$ to $\mathbb{Z}$ is defined by the formula $g(n) = n + 1$
Please

• Please be on time
• Please show respect for your classmates
• Please turn off (or use vibrate for) your cellphones

…

• On-topic questions are welcome
Welcome
and Enjoy!