Theoretical Computer Science
CS 310

Chadd Williams

Office Hours: chadd@pacificu.edu
Mon 3:00 – 4:00 PM
Tues 2:00 – 4:00 PM
Fri 11:00 – 12:00 PM
and by appointment

202 Strain

http://zeus.cs.pacificu.edu/chadd/cs310f06/
Syllabus
http://zeus.cs.pacificu.edu/chadd/cs310f06/syllabus.html

• *Introduction to the Theory of Computation* by Michael Sipser, (Second Edition)
  – I will assign problems out of this book

Grades:
• Homework: 15%
• 2 Exams: 25% each
• 1 Final 35% (Comprehensive)

Dates:
• Midterm 1, Wed Oct 11, 2006
• Midterm 2, Wed Nov 13, 2005
• Final, Tue Dec 5 (8:30 – 11:00 AM)

Policies:
• Assignments are due at the beginning of class. Late assignments will not be accepted.
• The cheating policy is defined in the Pacific Catalog
• Silence all electronic devices
Today

• Overview of class

• Mathematical Notation

• Proof by Induction

• Who Am I?
Overview

• What are the fundamental capabilities and limitations of computers?
• Computer Science is really the science of computation, not of computers.
• How does theory related to programming?
• Complexity Theory

• Computability Theory

• Automata Theory
Mathematical Notation

• Basic notations we will use in this class
  – Page 16 of your book has a partial list (no symbols!)

• Set
  \{ 7, 21, 57 \} \{ 1, 2, 3, \ldots \} \{ \text{gold, blue} \}

• Subset

• Proper Subset
Sets

- Shorthand for describing a set

\{ n \mid \text{rule about } n \}\n
\{ n \mid n = m^2 \text{ for some } m \in \mathbb{N} \}\n
\{ \{i, i^2\} \mid i \in \mathbb{N} \}\
Set Operations

• What can we do with sets?
• Union

• Intersection

• Complement
Sets

• Power Set
  \(\{ 0, 1 \}\)

• Cartesian Product (Cross Product)
  \(\{ 0, 1 \} \times \{ a, b \}\)
Sequences/Tuples

• Sequence
  (7, 21, 57)  (21, 7, 57)  (gold, blue)

• Tuple
  K-tuple
Functions

• Object that takes input, produces output
f (a) = b

• Domain and Range
  \[ f : D \rightarrow R \]

• Onto
Functions

• $f : A_1 \times A_2 \times \ldots \times A_k \rightarrow R$
  
  $(a_1, a_2, \ldots, a_k)$

  k-ary

  arity

  unary ($k=1$) binary ($k=2$)

• Notation

  Infix notation: $a + b$

  Prefix notation: $\text{add}(a,b)$
Relations

• Predicate (property)
  \( f : D \rightarrow \{\text{TRUE, FALSE}\} \)

• Relation
  \( f : A_1 \times A_2 \times \ldots \times A_n \rightarrow \{\text{TRUE, FALSE}\} \)

• Notation
  table
  Set
Equivalence Relations

binary relation
shows that two objects are equal

must satisfy 3 conditions:

1. R is **reflexive** if for every x, xRx;
2. R is **symmetric** if for every x and y, xRy if and only if yRx;
3. R is **transitive** if for every x, y, and z, xRy and yRz implies xRz
Proof by Contradiction

• Assume it is false
• Show this leads to a false consequence
• Prove $\sqrt{2}$ is irrational
  – Assume it is rational: $\sqrt{2} = \frac{m}{n}$
  – Reduce $\frac{m}{n}$ to lowest terms: $m$ and $n$ are not both even (could reduce out a 2)
  – sometimes tricky to pick exactly what false consequence to show
Proof by Induction

• Basis
  Prove $P(1)$ is true

• Induction Step
  Prove that for each $i \geq 1$, if $P(i)$ is true, then so is $P(i+1)$; assume $P(i)$ is true

• Basis + Induction Step
  $P(1)$ is true, $i = 1$
  $P( (i+1) )$ is true
  $P( (i+1) +1 )$ is true …
Proof by Induction

- Prove: $1 + 2 + \ldots + n = \frac{n(n+1)}{2}$ for $n \geq 1$

Basis:

Induction:
Chadd Williams

• New Computer Science Professor!
• Education
  – West Virginia University (BS)
  – University of Maryland, College Park (MS,PhD)
• Research
  – Systems
    • Runtime code patching
    • modify instructions in a running executable
  – Programming languages/Software Engineering
    • Studying software change history to learn about the source code
Graphs

• Graph
  Collection of nodes and edges
  \( G = (V,E) \)
  \( E = \) undirected

• Degree of a node
Graphs

- Subgraph
- Path

Connected graph

- Cycle
Trees

- Connected graph with no simple cycles
- Leaves
- Root
Directed Graph

- Arrows denote which way an edge goes.
- Outdegree/indegree
- $G=(V,E)$
- V=
- E=
- Directed path

Strongly connected