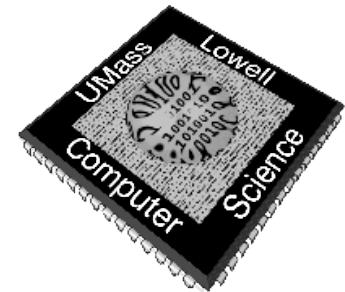


UMass Lowell Computer Science 91.404

*Analysis of Algorithms*

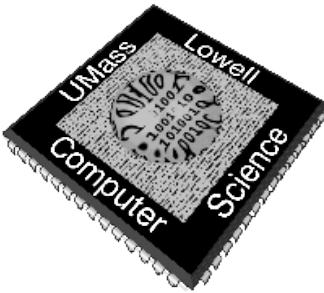
Prof. Benyuan Liu



# Lecture 1

## Introduction/Overview

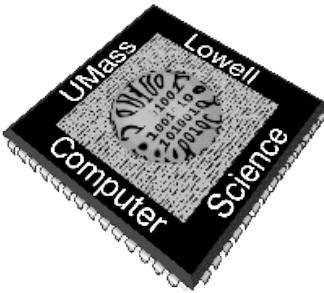
Course webpage: <http://www.cs.uml.edu/~bliu/404-F14>



# Nature of the Course

- Core course: *required for all CS majors*
- Advanced undergraduate level
  - Graduate students take separate course (91.503)
- No programming required
  - “Pencil-and-paper” exercises
  - Lectures supplemented by:
    - Programs
    - Real-world examples





# What's It All About?

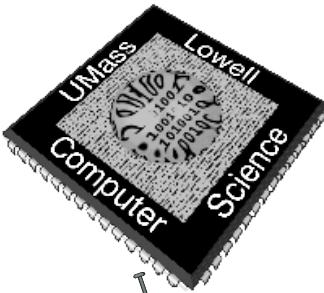
- **Algorithm:**

- steps for the computer to follow to solve a problem
- *well-defined computational procedure* that transforms input into output

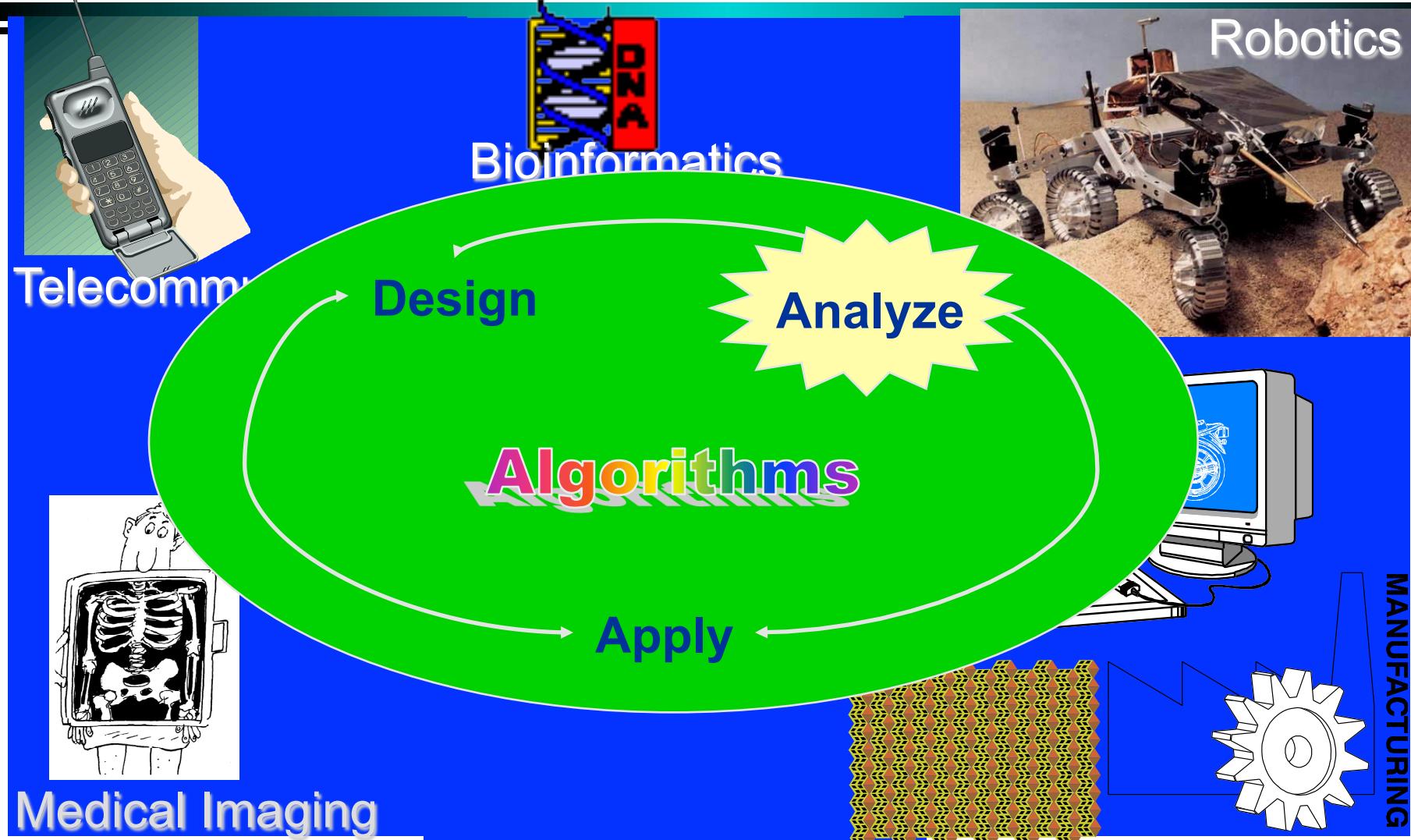
- Some of our goals:

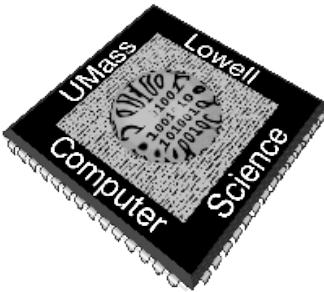
- recognize structure of some common problems
- understand important characteristics of algorithms to solve common problems
- select appropriate algorithm to solve a problem
- tailor existing algorithms
- create new algorithms





# Some Algorithm Application Areas





# Some Typical Problems

## Sorting

Input: Set of items

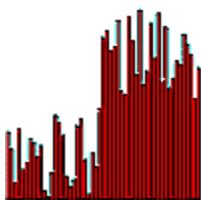
Problem: Arrange items “in order”



## Median finding

Input: Set of numbers or keys

Problem: Find item smaller than half of items and bigger than half of items



## Minimum Spanning Tree

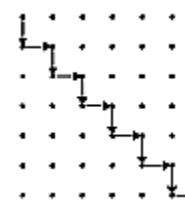
Input: Graph  $G = (V, E)$  with weighted edges

Problem: Find subset of  $E$  of  $G$  of minimum weight which forms a tree on  $V$



## Shortest Path

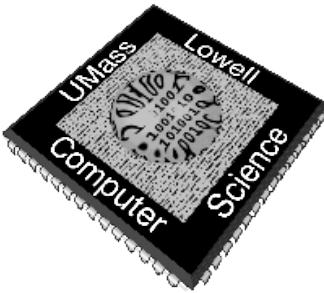
Input: Edge-weighted graph  $G$  , with start vertex  $s$  and end vertex  $t$



Problem: Find the shortest path from  $s$  to  $t$  in  $G$

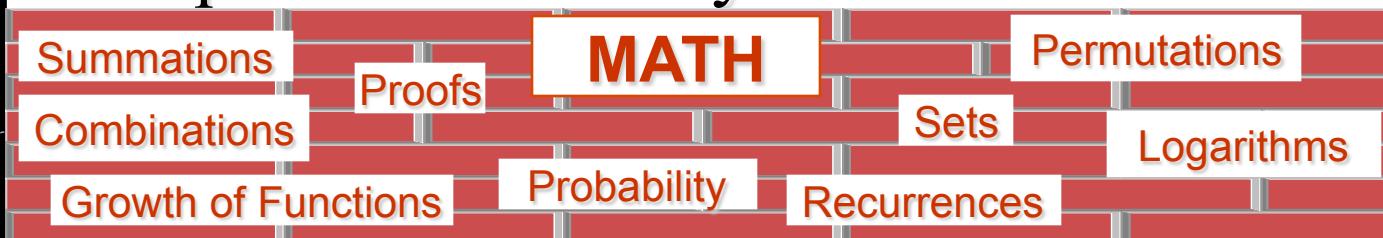
SOURCE: *Steve Skiena's Algorithm Design Manual*

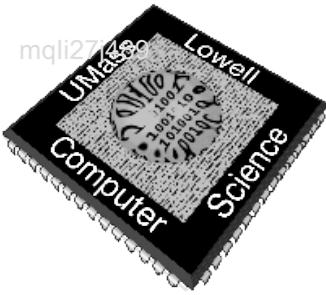
(for problem descriptions, see graphics gallery at <http://www.cs.sunysb.edu/~algorith/>)



# Tools of the Trade

- Algorithm Design Patterns such as:
  - divide-and-conquer
- Data Structures such as:
  - trees, linked lists, hash tables, graphs
- Algorithm Analysis Techniques such as:
  - asymptotic analysis
  - probabilistic analysis





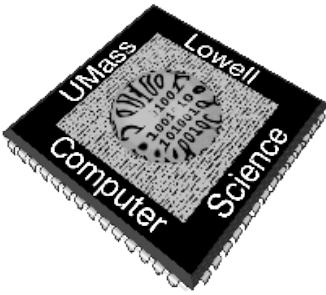
# Tools of the Trade: (continued) Algorithm Animation

---

---

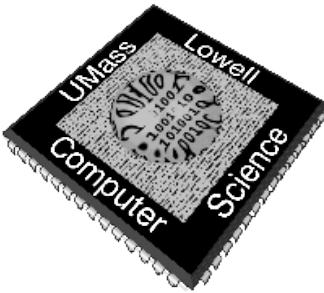
<http://www.sorting-algorithms.com/>

<http://www.site.uottawa.ca/~stan/csi2514/applets/sort/sort.html>



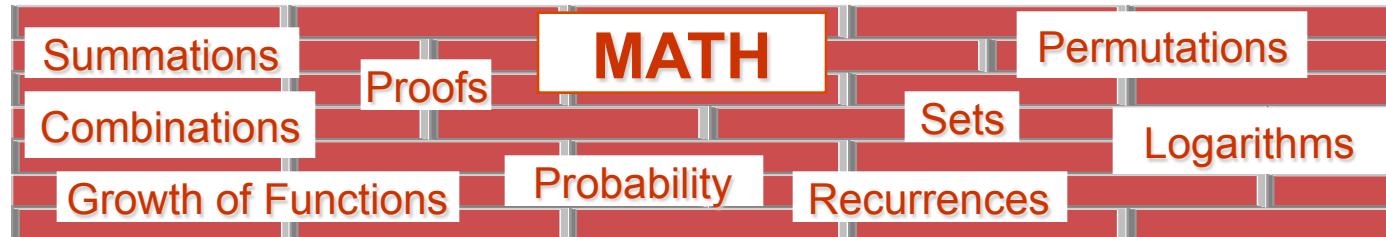
# What are we measuring?

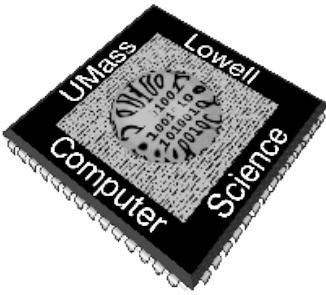
- Some Analysis Criteria:
  - Scope
    - The problem itself?
    - A particular algorithm that solves the problem?
  - “Dimension”
    - Time Complexity? Space Complexity?
  - Type of Bound
    - Upper? Lower? Both?
  - Type of Input
    - Best-Case? Average-Case? Worst-Case?
  - Type of Implementation
    - Choice of Data Structure



# Prerequisites

- Computing I (91.101)
- Computing II (91.102)
- Discrete Math I & II (92.321, 92.322)
- Statistics for Scientists and Engineers (92.386)
- Calculus I-II (92.131-132)





# Course Structure: 4 Parts

## ► Foundations

**Part 1**

- Analyzing & Designing Algorithms, Growth of Functions, Recurrences, Probability & Randomized Algorithms

## ► Sorting

**Part 2**

- Heapsort, Priority Queues, Quicksort, Sorting in Linear Time

## ► Data Structures

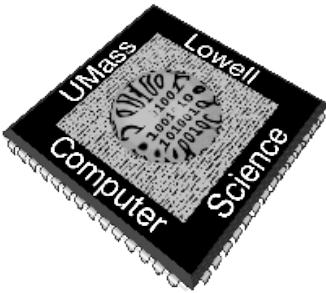
**Part 3**

- Stacks and Queues, Linked Lists, Introduction to Trees, Hash Tables, Binary Search Trees, Balancing Trees: Red-Black Trees

## ► Graph Algorithms

**Part 4**

- DFS, BFS, Topological Sort, MST, Shortest paths



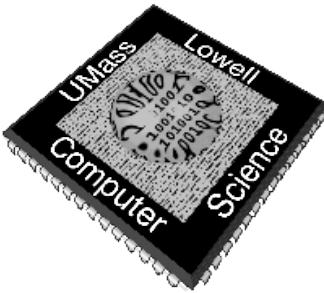
# Textbook



## Required:

- *Introduction to Algorithms* (third edition)
  - by T.H. Cormen, C.E. Leiserson, R.L. Rivest, Clifford Stein
  - McGraw-Hill
  - 2009
  - ISBN 978-0-262-03384-8
  - See course website for recommended texts

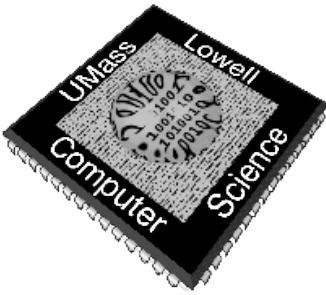




# CS Theory Math Review Sheet

## The Most Relevant Parts...

- p. 1
  - $O, \Theta, \Omega$  definitions
  - Series
  - Combinations
- p. 2 Recurrences & Master Method
- p. 3
  - Probability
  - Factorial
  - Logs
  - Stirling's approx
- p. 4 Matrices
- p. 5 Graph Theory
- p. 6 Calculus
  - Product, Quotient rules
  - Integration, Differentiation
  - Logs
- p. 8 Finite Calculus
- p. 9 Series



# Grading

- Homework (~8) 30%
- Midterm (chapters 1-6, open book & notes) 25%
- Quiz (~2) 10%
- Final Exam (cumulative, open book & notes) 30%
- Discretionary (attendance, participation, quiz) 5%