Object Oriented Programming and Design in Java

Session 24

Instructor: Bert Huang

Announcements

- Homework 4 solutions posted
- Homework 5 due next class:
  Mon. May 3rd
- Mon. May 3rd: Final review
- Mon. May 10th, Final exam. 9 AM - noon
  - closed-book/notes, focus on post-midterm material, but material is inherently cumulative

Review

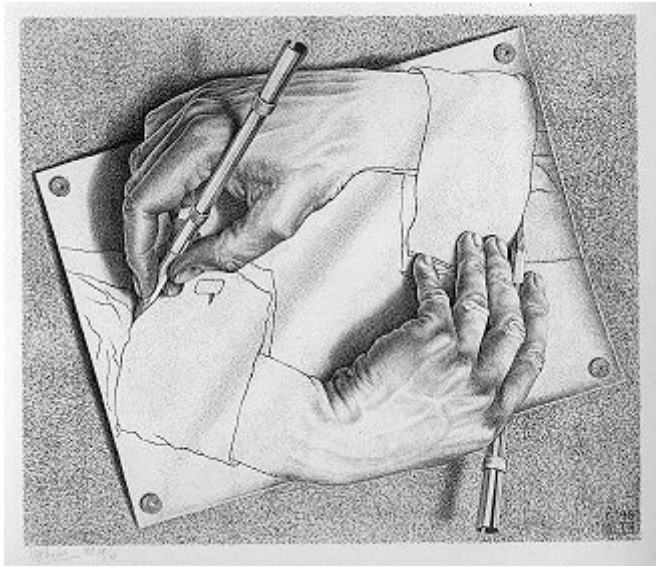
- Multithreading with Conditions review (for the homework)
- Multithreading in the chat program
- Sending non-string data over the network
- MVC over the network

Today's Plan

- Recursion: Towers of Hanoi
- Some thoughts on
  - COMS W1007, Object-oriented design in general, Computer science
- Advice for future computer science study

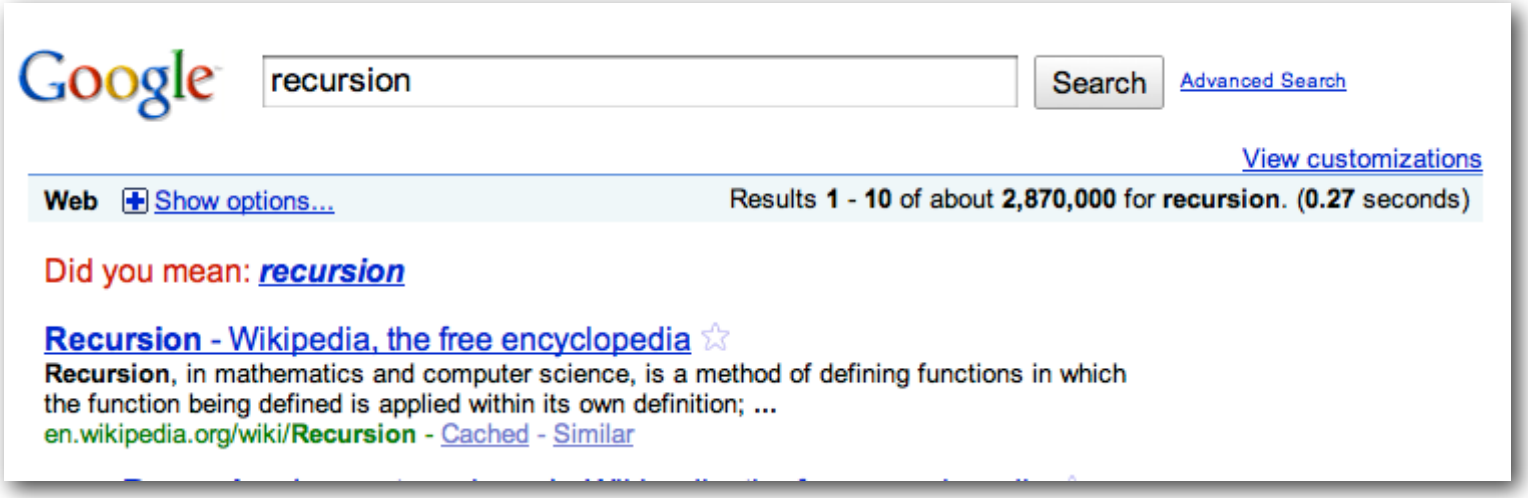
Recursion

- Recursion is self-reference
- `this` is a self-reference
- Methods can call themselves
- Allows for elegant description of some computations



http://en.wikipedia.org/wiki/File:DrawingHands.jpg

Silly Recursion Examples

- GNU stands for GNU is Not Unix
- 

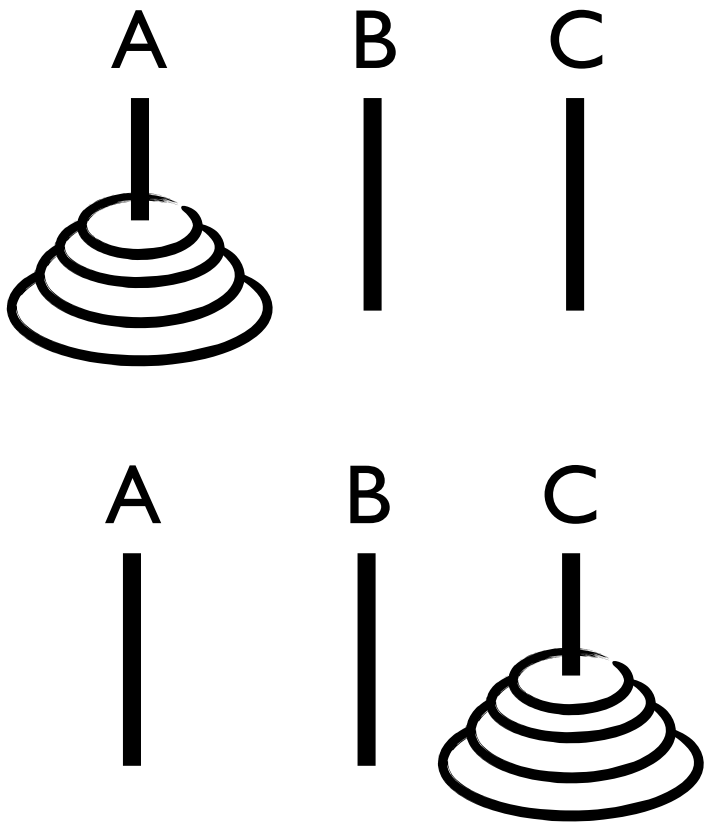
- This sentence is not true.

Concepts in Recursion

- Recursive routines (methods) solve a problem by calling themselves on subproblems
  - e.g., factorial(k) = k * factorial(k-1)
- Recursive routines have a *base case*, which is an input for which no recursion is necessary
  - e.g., factorial(0) = 1

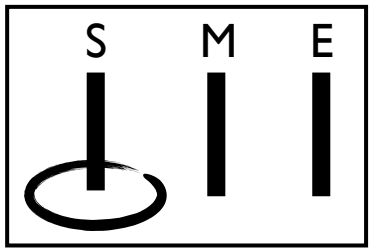
Towers of Hanoi

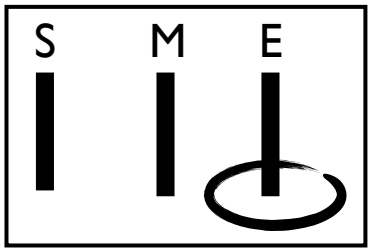
- Three pegs, N discs fit on pegs
- discs are of different sizes
- only smaller discs can be placed on larger discs
- Task: move all discs from one peg to another



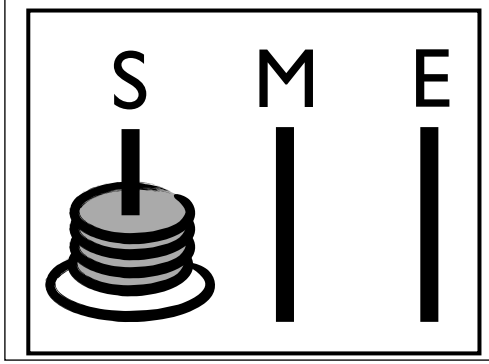
Recursive Solution

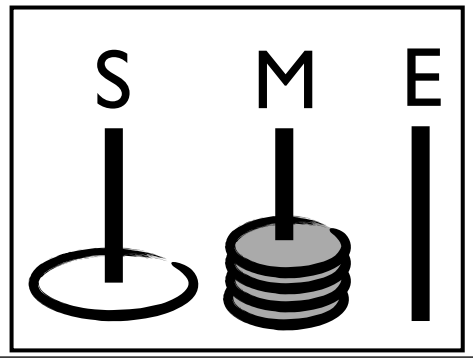
- solveHanoi(Peg start, Peg end, Peg Middle, int N)
- Base case: move directly to end

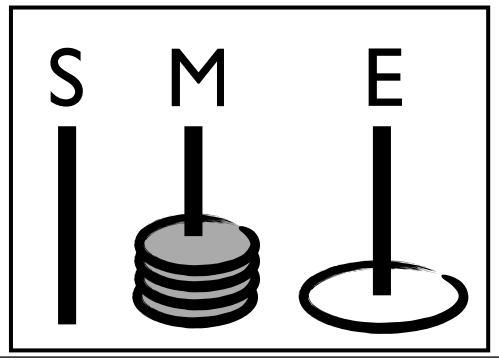


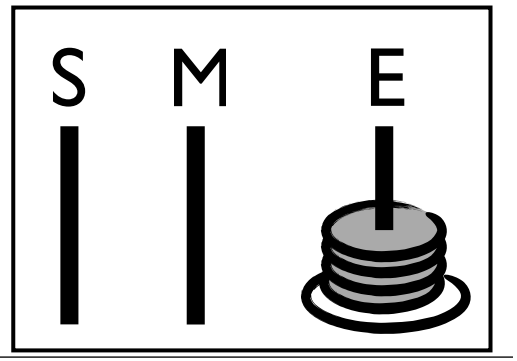


- Otherwise,
  - solveHanoi(start, middle, end, N-1) // move stack out of way
  - solveHanoi(start, end, middle, 1) // move bottom disc to end
  - solveHanoi(middle, start, end, N-1) // move stack onto bottom disc









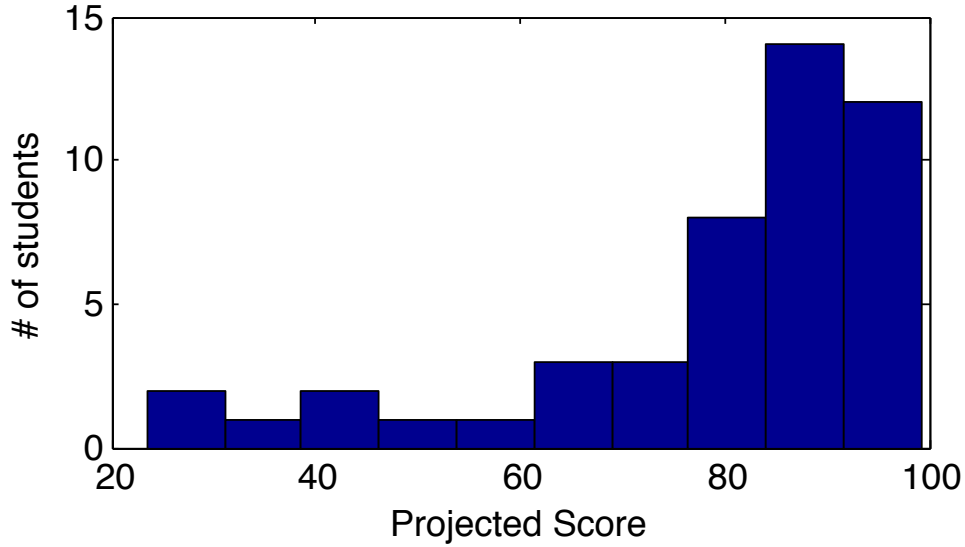
Recursion

- Provides elegant descriptions of algorithms
- May not always be the most efficient implementation
- Trade off between elegance and efficiency

Pre-Review Course Wrap-Up

- Before the review, when we'll have tons of material to cover in a quick class session, we should reflect for a moment
- Take a step back, look at big picture
- What did we study?
- What should we take away from this course?

Grades So Far



- Histogram of grades so far (hw1-hw3 and midterm exam)
- Average: 79
  - there will be a scaling of scores

COMS W1007

- The second course for majors in computer science.
- A rigorous treatment of object-oriented concepts
  - using Java as an example language.
- Development of sound programming and design skills, problem solving and modeling of real world problems from science, engineering, and economics using the object-oriented paradigm.

Core Courses of CS@CU

- 1004 Intro -
  a little programming, a little theory
- 1007 OOP and Design - lots of programming
- 3203 Discrete Math - lots of theory
- 3137 Data Structures and Algorithms -
  lots of theory, a little programming
- 3157 Advanced Programming -
  lots of programming

Object Oriented Design

- will help *tremendously* in real applications (on the job, side projects),
- will help with larger group projects,
- will help *a little* with many programming assignments,
- will probably waste your time on some small programming exercises.

Java as an Example Language

- Java is a nice programming language that does a great job implementing object-oriented ideas
  - but it is not perfect
- Java is also not the best language to learn because
  - too much is built in (e.g., data structures)
  - losing popularity in industry
    http://www.tiobe.com/index.php/content/paperinfo/tpci/index.html

What We Covered

- Programming style
- Classes and methods
- **UML diagrams**
- Programming by contract: preconditions, postconditions and invariants
- Designing interfaces
- Polymorphism
- Encapsulation
- Inheritance
- **Design patterns**
- Frameworks
- **Java graphics and user interface programming**
- **Multithreading**

Computer Science

- Much of CS is just organized common sense
- CS ideas pop up in real life
  - scheduling threads, scheduling work between collaborators, multitasking
  - “object-oriented” organization in everyday writing

Why Study CS?

- It's fun to build things
  - CS is one of the few scientific and engineering disciplines where you can actually build the things you study
- Lower-hanging fruit than many fields
- The world needs computer scientists and engineers

Random Advice for Future Classes

- Brush up on math
- Start early
- Write English (and code) clearly;
  - pay attention to details such as grammar, syntax, usage and spelling
- Go to class and use office hours

The Final Stretch

- Give Homework 5 your all, get help if you need it; plenty of office hours left
- I'll post a sample final by Monday
- Come to the review with questions
- Don't let me and the TAs get away with not teaching you something