CS 367 - Introduction to Data Structures
Tuesday, July 7, 2015, Lecture 14

Course website: http://pages.cs.wisc.edu/~cstapleton/367/
Piazza: https://piazza.com/wisc/summer2015/cs367/

Instructor
Cea Stapleton
cstapleton@cs.wisc.edu ; Use Piazza for course related questions.

TA
Haseeb Tariq
haseeb@cs.wisc.edu ; Use Piazza for course related questions.

Last Time
- Recursion

Today
- Recursion
- Midterm on Thursday!

Next Time
- Finish Recursion
- Start Trees
Recursion Recap

Definition:

Key Rules:

1. 

2. 

Questions to keep in mind

- How can you solve the problem in terms of smaller problems of the same kind?
- What instances of the problem can be used as base cases?
- How does the problem size decrease in each recursive call?
- As the problem size decreases will a base case be reached?
Recursion Example: n choose k

Conventional Definition:

Recursive Definition:

Implementing the recursive definition:
Tracing an execution tree:

**Recursion Examples: Arrays and Linked Lists**

int countEven(int[] a):

boolean isIncreasing(ListNode<Integer> head):
Tail Recursion

http://www.xkcd.com/1270/

Analyzing Complexity of Recursive Methods
Recursion Example: Binary Search
Recursion Example: Palindromes

Write a recursive method to determine if a given input String is a palindrome.
Palindrome examples:

   eye
   kayak
   racecar
   Was it a rat I saw?
   Never odd or even!
   Amy, must I jujitsu my ma?
   Are we not drawn onward to new era?

Assume: input String is not null, all spaces and punctuation removed, all lowercase.

Useful String methods:

   char charAt(int index)
   int length()
   String substring(int begin, int end)