

COP 3502 – Computer Science I



- Outline of Material Covered:
 - I. Arrays, Pointers, Strings, Files, Structs
 - Manipulation of array elements
 - Manipulation of struct components
 - Use of '.' vs '->'
 - Use of strings
 - strcmp, strlen, strcpy
 - Use of files (fopen, fscanf)
 - Dynamic memory allocation for arrays and for structs
 - malloc, calloc, and realloc



- Outline of Material Covered:
 - Linked Lists
 - Traversing a linked lists
 - Printing a list
 - Modifying list contents
 - How to allocate a node dynamically
 - Inserting elements anywhere in the list
 - Deleting elements anywhere in the list
 - Everything is fair game including insert/delete.
 - Know the code!



Outline of Material Covered:

- III. Recursion
 - Fibonacci, Factorial, Binary Search
 - Writing recursive functions
 - Tracing through recursive functions
 - Towers of Hanoi
 - Permutation
 - Reversing a string
 - Base conversion
 - Convert from some other base to base 10
 - Convert from base 10 to some other base
 - ***Convert from ANY base to ANY other base***



Outline of Material Covered:

IV. Algorithm Analysis

- Big-O definition and finding the c value as shown in class
 - Understanding the various orders and what they mean
- "Practical" Problems such as those on the slides and also during the lab
- Analyzing code fragments and determining Big-O
- Solving summations
- Putting summations in their closed form (in terms of n)
- Analyzing code fragments and using summations to determine the Big-O



- Outline of Material Covered:
 - V. Recurrence Relations
 - You will have at LEAST one full recurrence relation to work out
 - This will be a 15 point question
 - We have done SEVERAL of these in class
 - No excuse at all to miss this question
 - You may have other smaller questions on this topic:
 - Given several equals representing several steps of the recursion, give the correct recurrence relation for the kth step
 - Given code, develop the recurrence relation
 - But don't actually solve it
 - And other similar questions on the topic



Outline of Material Covered:

VI. Stacks

- What they are and how they work
- The idea of Abstract Data Types
- Stack operations
- Converting Infix to Postfix
- Evaluating Postfix expressions
- Implementation of stacks using arrays and linked lists
 - Understand how the code works!



Outline of Material Covered:

VII. Queues

- What they are and how they work
- Basic operations of a queue
- Different implementations of a queue
 - Both regular array implementations
 - Be able to explain the problems with these two methods
 - Circular array implementation
 - Be able to explain why this one is better
 - Linked list implementation
- Be able to answer short questions on coding queues
- Know when to use a queue versus a stack



- Outline of Material Covered:
 - VIII.Binary Trees
 - Basic info on trees
 - root, leaves, height, # of nodes, complete tree, full tree, etc
 - What is a BST (ordering property)
 - Implementation of a BST
 - Tree traversals
 - Depth first (preorder, inorder, postorder) and breadth first



Outline of Material Covered:

- IX. Binary Trees
 - Insertion into a BST
 - Given a list of values, know how to make a tree inserting those values into the tree
 - Know the code for insertion
 - Deletion from a BST
 - Know the three cases and understand how to delete
 - Various functions
 - You WILL have a coding question on the test
 - Most likely to code some type of function on binary trees
 - Between the slides and lab sheets, you have more than a DOZEN examples. Make sure you are okay with them.



Outline of Material Covered:

- X. Sorting
 - Know running times of all sorts!
 - Know the N-squared sorts
 - Be able to show the step-by-step sorting of a list of values using any of these sorting methods
 - Just like what was shown in lab
 - Understand the limitation of these sorting algorithms
 - Merge Sort
 - Know how it works and be okay with the analysis of it
 - Understand the merge function
 - Quick Sort
 - Partition, best type of pivot, and analysis of quick sort



Outline of Material Covered:

XI. <u>Heaps</u>

- What is a heap and what is its purpose?
 - What are the properties of a binary heap?
 - Know the variations of a heap
- Know how to add nodes to a heap
 - Where to add and then what to do afterwards
 - Percolate up
- Know how to delete from a heap
 - Where do we delete from? And what do we do afterwards?
 - Percolate down
- Know how to build a heap from scratch
- Understand the analysis of Heapify resulting in O(n)



Outline of Material Covered:

XII. Hash Tables

- Know why we use hash tables
- Understand the use of hash functions
- Know what collisions are
- Know the different collision resolution methods:
 - Linear probing
 - Quadratic probing
 - Double hashing
 - Bucket hashing
 - Separate chaining
- Be able to compare and contrast resolution methods



Outline of Material Covered:

XIII. AVL Trees

- Know what is an AVL tree and what is its purpose
- Know how to insert node into an AVL tree
- Know how to delete nodes from an AVL tree
- Know how to fix an imbalances that occur
 - You can use EITHER the rotate method
 - Or the A,B,C method
- Know how to create a tree from a list of nodes
 - Just a series of insertions



- How to study:
 - KNOW and UNDERSTAND the notes
 - Make sure you are 100% on the notes
 - Make sure you are 100% on all the lab questions and their respective solutions
 - Don't waste time memorizing algorithms
 - Understand how they work and WHY they work
 - And be prepared to come up with your own
 - Look at the archive of Foundation Exams
 - Practice some of the problems (ones that are applicable)
 - http://www.cs.ucf.edu/registration/exm/index.html



- Types of Questions:
 - Some short answer questions:
 - Tracing through code
 - Questions on an algorithm discussed in class
 - Small questions on code
 - etc.
 - Writing Functions:
 - You will have to write functions
 - Perhaps on binary trees or on Heaps
 - And usually will be recursive



Exam Aids:

- You may use TWO 8-1/2"x11" sheets of paper
 - FRONT AND BACK
 - Typed or written doesn't matter
 - I don't care what you put on it
- What you CANNOT use:
 - Any electronic device:
 - Calculator, phone, ipad, you get the idea
 - If you are seen holding ANY electronic device, you will get 10 points off immediately! If you were cheating with that device, then the consequences are, of course, far worse.



- So what is covered?
 - EVERYTHING since birth:
 - Penmanship, basic math (2+2=3), spelling, etc.
 - Also, EVERYTHING taught this semester
 - Even if I didn't "cover" it during this review
 - Anything and everything that was taught or shown in class or in the labs is fair game.
 - Including material from the first two exams

Questions:



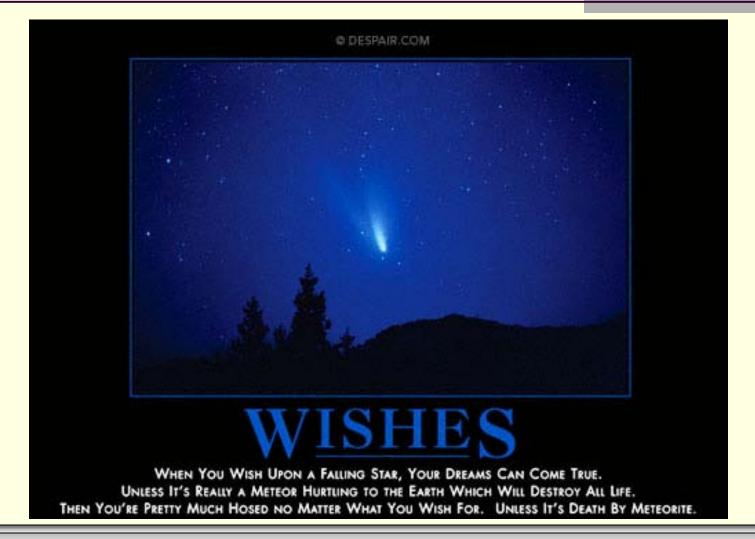
Final Exam TIME

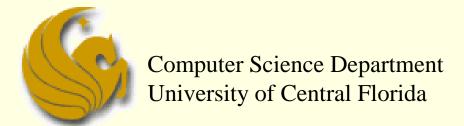
- Exam is on Friday, April 27th
 - Exam time: 8:00 AM sharp (not 7:00 AM)
 - You can come at 7 AM. But I'll be here at 8!
 - You will have until 9:50 AM to complete your work

Again, IN BOLD, start time is 8:00 AM



Daily Demotivator





COP 3502 - Computer Science I