

Test 2 will be Thursday April 4th in class. It will be closed-book and notes, except for one 8.5” x 11” sheet of paper containing any notes that you want. (Yes, you can use both the front and back of this piece of paper.) Plus, you can use your Python Summary handout.

The test will cover Chapters 4 and 5. The following topics (and maybe more) will be covered:

Chapter 4: Recursion

Recursive functions: base-case(s), recursive case(s), tracing recursion via run-time stack or recursion tree, “infinite recursion”

Costs and benefits of recursion

Recursive examples: countDown, OrderedDict __str__ method, fibonacci, factorial, binomial coefficient

Divide-and-Conquer technique of solving a problem. Examples: fibonacci, coin-change problem

Backtracking technique of solving a problem: Examples: coin-change problem, maze (textbook)

General concept of dynamic programming solutions for recursive problems that repeatedly solve the same smaller problems over and over. Example fibonacci, coin-change problem, binomial coefficient

Chapter 5: Searching and Sorting

Sequential/Linear search: code and big-oh analysis

Binary Search: code and big-oh analysis

Python List implementation (ListDict) of dictionaries and big-oh analysis

Hashing terminology: hash function, hash table, collision, load factor, chaining/closed-address/external chaining,

open-address with some rehashing strategy: linear probing, quadratic probing, primary and secondary clustering

hashing implementation of dictionaries (ChainingDict and OpenAddrHashDict) and their big-oh analysis

General idea of simple sorts

Simple sorts: selection, bubble, insertion sorts and their big-oh analysis

Advanced sorts and their big-oh analysis: heap sort, quick sort and merge sort