The second exam will cover all material that we covered in class since the class preceding the first midterm through March 28, together with a general overview of Shell sort, detailed below. In particular it starts with the AVL deletion algorithm and AVL tree worst case analysis. The major topic categories after those are:

- B-trees
- Hashing and hash tables
- Binary heaps
- Sorting: sorting algorithms that sort by exchanging adjacent elements, lower bound analysis, shell sort

For each topic, performance analysis is included. For example, the running times of the various insertion and deletion algorithms, or results concerning the sizes or heights of trees, are fair questions to expect on the exam.

The format of the exam includes true/false questions, short answer questions, questions that ask you to analyse algorithm performance or carry out algorithms or code on examples, and questions that ask you to write small amounts of code.

Some sample questions of various types are below.

1. Build a heap from an array containing the keys $30, 40, 25, 60, 75, 23, 86, 12, 72$ in that order.
2. Show the state of a hash table of size 31 that uses the hash function $h(x) = x\%31$ with quadratic probing, given that the keys are inserted in the order $10, 20, 30, 40, 50, 60, 70, 80, 90, 100$. Determine how many collisions take place.
3. Sort a given array using Shell sort with the increment sequence 8,4,1: (make up your own arrays).
4. What is the expected number of probes using linear probing in a successful search of a hash table?
5. True or False? The average number of comparisons to insert n elements into a heap is $O(n \cdot \log n)$.
6. What is height of the tallest AVL tree with 32 nodes?
7. Perform a deletion on some AVL tree and do the rotation necessary to rebalance the tree.
8. Write a function, that given a string of lowercase letters of length 8, encodes it as a long integer.