The exam will be accumulative and may cover materials from the textbook, lecture notes, quizzes, and assignments. It will focus slightly more on the materials after midterm. Below is a review of the key topics.

Java Basics and OO
- Object type variables vs. primitive type variables
- Method calls, call stack
- Local variables vs. Parameter variables vs. Instance variables
- Inheritance

Arrays
- Typical array processing and traversal
- Binary search for ordered arrays – algorithm, implementations (loop-based version and recursive version), and cost analysis

Algorithm analysis
- Common functions - constant, logarithm, linear, quadratic, polynomial, exponential, factorial in increasing order of growth
- Cost analysis using direct methods and recursive relations

Linked lists
- Linked lists vs. array
- Insert, search, delete – implementations and cost
- Variations - double-ended list, doubly-linked list, circular list, sorted linked list

Stacks and queues
- Basic operations: push and pop for stacks; enqueue and dequeue for queues
- Implementations using resizing arrays and linked list
- Generics and iterators
- Deques

Sorting
- Bubble sort, selection sort, and insertion sort – algorithms, implementations, and cost analysis
- Mergesort and quicksort – algorithms, implementations, and cost analysis

Recursion
- Recursive method, base case
- Tracing recursive method calls and evaluating the output
- Solving recurrence relations
- Divide and conquer

Binary search trees
• Definitions and terminologies
• Insert, search, finding minimum/maximum, select, rank – algorithm, implementation, and cost
• Pre-order, in-order, post-order traversal – algorithm, implementation, cost
• Delete – understand algorithm and be able to predict the result after delete

Priority queue and heaps
• Tree vs. array implementation
• Insert, removeMin/removeMax: algorithm, predict result, cost.

Hash table
• Definitions
• Common hashing functions (modulo)
• Collision resolution: separate chaining, linear probing

Graphs
• Definitions and terminologies
• Adjacency matrix and adjacency list representations
• BFS and DFS traversal algorithms – algorithm, implementation
• Shortest path for unweighted graphs (BFS)
• Connected components
• Shortest path for weighted graphs (Dijkstra, A*) – algorithm, implementation, predict results
• Minimum spanning trees (lazy Prim’s algorithm) – algorithm, implementation, predict results