MAT 3770—Exam 2 Topics
Sections 1.1–3.3, inclusive

The following is meant to be a guide, not an exhaustive nor all–inclusive outline.

- 2.3 Graph Coloring:
  coloring, chromatic number (χ), wheel graphs, scheduling problems, interval graphs, minimizing tracks in VLSI circuit design

- 2.4 Coloring Theorems:
  triangulation of a polygon, edge chromatic number
  Thm.1. The vertices in a triangulation of a polygon can be 3–colored.
  Corollary. The Art Gallery Problem with n walls requires at most $\lfloor \frac{n}{3} \rfloor$ guards.
  Thm. 2. (Brook's) If the graph G is not an odd circuit or a complete graph, then $\chi(\mathrm{G}) \leq d$, where d is the maximum degree of a vertex of G.
  Thm. 3. For any positive integer k, there exists a triangle–free graph G with $\chi(\mathrm{G}) = k$.
  Thm. 4. (Vizing's) If the maximum degree of a vertex in a graph G is d, then the edge chromatic number of G is either d or $d + 1$.
  Thm. 5. Every planar graph can be 5–colored.

- 3.1 Properties of Trees root, tree, directed (rooted) tree, vertex, node, level number, parent, children, descendant, ancestor, siblings, leaf, internal nodes, m–ary tree, binary tree, tree height, balanced trees, Prufer sequences
  Thm. 1. A tree with n vertices has $n - 1$ edges.
  Thm. 2. Let T be an m–ary tree with n vertices, of which i vertices are internal. Then $n = mi + 1$.
  Corollary. Let T be an m–ary tree with n vertices, consisting of i internal vertices and l leaves. If we know one of n, i, or l, then the other two parameters are given by the formulas:

  1. Given i, then $l = (m - 1)/i + 1$ and $n = mi + 1$.
  2. Given l, then $i = (l - 1)/(m - 1)$ and $n = (ml - 1)/(m - 1)$.
  3. Given n, then $i = (n - 1)/m$ and $l = [(m - 1)n + 1]/m$.

- 3.2 Depth–First and Breadth–First Search

  1. Algorithms, complexity
  2. Creating DFS tree/forest, BFS tree/forest on directed and undirected graphs
  3. Applications

- Topological Sort: Algorithm, complexity, applications

- Graph Traversals: PRE–, IN–, and POST–Order

- 3.3 : Spanning Trees: Minimal spanning trees: Kruskal & Prim Algorithms

- Creating heaps ($O(n \log n)$ and $O(n)$ *heapifies*), heapsort