Coverage for Test 2

Discrete Computational Structures (CS 2071) Fall 2021

Test 2 will be held on **Wednesday, November 10**.

Coverage for Test 2 will include all the material in the lectures after Test 1, i.e., from October 8th lecture to November 5th lecture, inclusive.

Lecture October 8 – RSA Public-Key Cryptosystem

- Public-key cryptosystem RSA (Rivest-Shamir-Adleman)
- Public-key cryptosystems pioneered by Diffie and Hellmann
- Encryption formula for message: $m^e \ mod\ n$
- Decryption formula for message: $c^d \ mod\ n$
- Using extended GCD to compute private key *d* from public key *e*
- Application of Euler's Totient Theorem in proof of correctness

Lecture October 13 – Intro to Graph Theory, Euler's Degree Formula

- Graph definition
- Degree of a vertex
- Euler's degree formula: $\sum_{v \in V} \deg(v) = 2m$
- Parity result that the number of vertices of odd degree is even
- *r*-regular graphs, formula: $m = \frac{rn}{2}$
- Average degree formula: $\alpha = \frac{2m}{n}$
- Complete graph, number of edges: $C(n, 2) = n(n-1)/2$
- Subgraph, induced subgraph
- Bipartite graph, complete bipartite graphs
- Handshaking Theorem

Lecture October 15 – Graph Isomorphism, Path, Coloring

- Graph isomorphism
- Paths
- Connectedness
- Connected components
- Proper vertex colorings.
- Graph search and traversal algorithms: Depth-First Search (DFS) and Depth-First Traversal (DFT)

Lecture October 18 – Planar Graphs and Euler's Polyhedron Formula

- Planar graph
- Kuratowski's characterization of nonplanar graphs.
- Dual graph, Euler's degree formula for faces
- Euler's Polyhedron Formula for connected planar graphs: $n - m + f = 2$
- Application of Euler's Polyhedron Formula and vertex and face degree formulas to show that there are only five regular polyhedra.
- Proof every planar graph has a vertex of degree at most 5
- Proof every planar graph can be properly 6-colored

Lecture October 20 – Spanning Trees and Eulerian Circuits

- Spanning trees
- Number of spanning trees in the complete graph: K_n has n^{n-2} spanning tees
- Minimum spanning trees: spanning tree having minimum weight, where the weight of the tree is the sum of the weights on its edges
- Kruskal's algorithm for computing a minimum spanning tree in a weighted connected graph is based on the Greedy Method: a forest is grown by choosing next smallest edge that does not form a cycle
- Eulerian circuits
- Konigsberg Bridge problem
- Characterization of Eulerian graphs, i.e., graphs that contains Eulerian circuits: a graph is Eulerian iff it is connected and every vertex has even degree

Lecture October 22 – Hypercubes and Hamiltonian Cycles

- Definition of k-dimensional hypercube H_k
- Number of vertices of H_k is 2^k
- Number of edges of H_k is $k2^{k-1}$
- Diameter of H_k is k
- Definition of k-bit Gray Codes
- k-bit Gray Codes correspond to Hamiltonian cycles in H_k
- H_k contains a Hamiltonian cycle for $k \geq 3$

Lecture October 25 – Implementation of Graphs and Digraphs

- Standard implementation of a graph: adjacency matrix and the adjacency lists
- Digraphs generalize graphs via the symmetric digraph
- Adjacency matrix and adjacency lists implementation extend to digraphs
- Powers of the adjacency matrix can be used to count the number of directed walks of a given length from vertex i to j for any given pair of vertices i and j

Lecture October 27 – Digraphs

- Definition of a digraph
- Digraph modeling of tournaments
- Directed acyclic graph or DAG
- Topological sorting
- Topological labeling
- Efficient algorithm based on DFT. Reverse of explored order

Lecture October 29 – The Web Digraph and PageRank

- Web digraph
- Page Rank derived from hyperlink structure of web, i.e., Web digraph
- Linear equations for Page Rank
- Matrix equation for Page Rank
- Interpretation using Principal Eigenvector
- Interpretation using random walks

Lecture November 1 – Intro to Combinatorics and Counting

- Traveling salesperson problem
- Maximum-weight perfect matching problem
- Combinatorial explosion
- Multiplication Principle and Addition Principle
- Number of permutations of an *n*-element set is *n*!
- Lexicographic order of permutation
- *r*-permutations of an *n*-element set: $P(n,r) = n(n-1)\cdots(n-r+1) = \frac{n!}{(n-r)!}$
- Combinations, i.e., number of ways $C(n,r)$ of choosing *r* elements from and *n*-element set $C(n,r) = \frac{P(n,r)}{r!} = \frac{n!}{r!(n-r)!}$
- Applications to counting number of poker and bridge hands

Lecture November 3 – Permutations and Combinations

- Computing k^{th} permutation
- Generating a random permutation
- Number of certain poker hands such as a pair, two pairs, Full House.
- Number of bridge hands with voids
- Permutations with repetitions: $P(n; r_1, r_2, \ \ldots, \ r_k) = \frac{n!}{r_1! r_2! \cdots r_k!}$
- Combinations with repetitions:
  - The number of integer solutions to the equation $x_1 + x_2 + \cdots + x_r = n$, $x_i \geq 1, \quad i = 1, \ \ldots, r$ is $C(n-1, r-1)$.

- The number of integer solutions to the equation $x_1 + x_2 + \cdots + x_r = n$, $x_i \geq 0, \quad i = 1, \ \ldots, r$ is $C(n + r - 1, r - 1)$.

Lecture November 5 – Identities, Binomial Theorem, Pascal’s Triangle

- Useful combinatorial identities
  - $C(n, k) = C(n, n - k)$
  - Newton’s Identity: $C(n, k) \times C(k, m) = C(n, m) \times C(n - m, k - m)$
  - Pascal’s Identity: $C(n, k) = C(n - 1, k) + C(n - 1, k - 1)$
- Binomial Theorem
- Pascal's Triangle