MATH 363: Discrete Mathematics

Learning Objectives by topic

The levels of learning for this class are classified as follows.

1. **Basic Knowledge:** *To recall and memorize* - Assess by direct questions. The object is to test the students' ability to recall facts, to identify and repeat the information provided.
2. **Comprehension:** *To translate from one form to another* - Assess by having students' 1) restate material in their own words, 2) reorder or extrapolate ideas, predict or estimate. Assessments must provide evidence that the students have some understanding or comprehension of what they are saying.
3. **Application:** *To apply or use information in a new situation* - Assess by presenting students with a unique situation (i.e. one not identical to that used during instruction) and have them apply their knowledge to solve the problem or execute the proper procedure.

1 Logic

1. Given a collection of sentences, determine which of them are propositions.

   Remember De Morgan's laws for propositions.

   Remember the Distributive laws for propositions.

   Recall two logical equivalences of the proposition $p \to q$ $(= \neg p \vee q = \neg q \to \neg p)$.

   Remember the definition of Tautology and Contradiction.

2. Given the truth values of a collection of propositional variables, determine the truth value of a compound proposition.

   Given a compound proposition, determine whether it is a tautology, a contradiction or neither.

   Translate compound propositions to logical circuits and viceversa.

   Given a statement with (nested) quantifiers and propositional functions, represent its negation using De Morgan's law for Quantifiers.

3. Given two compound propositions, determine if these are logical equivalent.

   Simplify a given compound proposition using logical equivalences.

Given a collection of specifications, determine if these are consistent.

Given a proposition in English, translate it to propositional logic. This can include nested quantifiers.

2 Proofs and Induction

1. Remember the components of an argument: hypothesis/premises and conclusions.

   Remember two arguments which are not valid:
   '$p \to q$ and q implies p' and '$p \to q$ and $\neg p$ implies $\neg q$'

   Explain the difference between a theorem and a conjecture.

2. Given a statement $\forall n P(n)$ which is true, provide a valid argument which proves it.

   Given a statement $\forall n P(n)$ which is false, provide a counterexample which disproves it.

3. Explain which are the two components of a proof by induction.

   Explain what is the pigeonhole principle.

3 Sets and Functions

1. Distinguish between a set of size n and an n-tuple.

   Distinguish between countable and uncountable sets.

   Distinguish between relations between A and B and functions from A to B.

2. Translate the representation of a given a set, from listing its elements to representing it by the properties of its elements; and viceversa.

   Given a set S of cardinality n, fix an ordering of the elements in S and represent S with a bit string.

   Given a set S obtained from operations with three sets A, B, C, represent S using Venn Diagrams

   Given two sets S, T, determine whether these are equal using Venn Diagrams.

   Given a function $f : A \to B$, identify the domain, codomain and range of f.

   Given a function $f : A \to B$ involving floors and/or ceilings, obtain the value of $f(x)$ for some $x \in A$.

   Given a function $f : A \to B$ and $a \in A$, identify the image of a under f.

   Given a function $f : A \to B$ and $b \in B$, identify the preimage of b under f.

3. Given the elements of a set, construct its power set.

   Given a function $f : A \to B$ determine whether f injective, surjection or bijective.

Carry out intersections and unions of a given collection of sets (e.g. $\bigcup_{i=1}^{5} \bigcap_{j=i}^{2i} A_j$)

Carry out nested sums/products of a given sequence of numbers (e.g. $\sum_{i=1}^{5} \prod_{j=i}^{2i} a_j$)

4 Growth of algorithms

1. Remember the 7 properties of an algorithm.

   Remember the definition of 'f is $O(g(x))$'

   Remember the definition of 'f is $\Omega(g(x))$'

   Remember the definition of 'f is $\Theta(g(x))$'

2. Given functions f, g determine whether f is $O(g(x))$, $\Omega(g(x))$ or both.

   Given a function f, provide a function $g(x)$ for which f is $O(g(x))$,

   Given a function f, provide a function $g(x)$ for which f is $\Omega(g(x))$.

3. Given a function f which is $O(g(x))$, provide witnesses $C, k > 0$ of this fact, and justify this choice.

   Given a function f which is $\Omega(g(x))$, provide witnesses $C, k > 0$ of this fact, and justify this choice.

   Given a function f which is not $O(g(x))$, provide an argument of why there are no witnesses $C, k > 0$ for '$f = O(g(x))$'.

   Given a function f which is not $\Omega(g(x))$, provide an argument of why there are no witnesses $C, k > 0$ for '$f = \Omega(g(x))$'.

5 Number Theory

1. Given integers a, b determine if a is a factor of b

   Given an integer a, express it as a product of prime factors

   Given integers a, b, m and $m \neq 0$, determine if $a = b$ (**mod** m)

   Remember that $gcd(a, b)$ divides any linear combination of a and b.

   Remember the statement of Fermat's little Theorem.

2. Given a pair of integers, identify their greatest common divisor

   Given a pair of integers, identify their lowest common multiple

   Given a seed x_0 an a linear congruential $ax + b$ (**mod** m), generate a given number of pseudorandom numbers.

   Given a message (encrypted message) and a cipher, encode (decode) the message.

Given a modulus n and an exponent e, explain the steps to encode a message with RSA encryption.

Given a modulus n and an 'inverse' d, explain the steps to decode a message with RSA encryption.

3. Given a true statement of divisibility, use the definitions to prove it.

   Given integers a, d, m and $m \neq 0$. Use the modular exponentiation to compute a^d (**mod** m)

   Given integers a, d, m and m prime. Use Fermat's little theorem to compute a^d (**mod** m)

   Given an equation $ax = b$ (**mod** m), find the inverse of a modulo m to solve the linear congruence.

   Given a modulus n, an exponent e and an 'inverse' d. Explain why the RSA encryption method recovers exactly the message encoded.

6 Combinatorics

1. Explain what is the inclusion-exclusion principle.

   Explain what is the sum, product and division rule.

   Define what is the number of permutations of k elements of a set of n elements; with no repetitions and with repetitions allowed.

   Define what is the number of combinations of k elements of a set of n elements; with no repetitions and with repetitions allowed.

   Express the binomial coefficient $\binom{n}{k}$ as a formula involving factorial numbers.

   Explain what is a combinatorial proof.

   State the Binomial theorem.

2. Given a set S of elements with a given property, apply the rules above to count the number of elements in the set S.

   Provide examples of combinatorial proofs (e.g. Pascal's identity, Vandermonde's equality, Number of subsets of a set.)

   Use the binomial theorem to extract coefficients of monomial $x^{n-k}y^k$ in $(x+y)^n$.

3. Given an equality involving binomial coefficients, prove such equality using a combinatorial argument.

   Use the binomial theorem to prove equalities involving binomial coefficients.

7 Probability

1. Define the probability of an event using Laplace's formula. (Favorable cases/Possible cases)

   Define what is the probability function from a sample space S to $(0, 1)$.

   State the properties of a probability function.

   Define what is the conditional probability of an event E given event F.

   State the law of total probability.

   State Bayes' formula.

   Define what are independent events and independent random variables.

   Define the expected value of a random variable.

   Explain the property of linearity of expected values.

2. Given an experiment with uniform outcomes and an event E. Compute the probability of event E using Laplace's formula.

   Given an experiment and an event E. Define the sample space S, the probability function and compute the probability of event E.

   Given a random variable, compute the expected value of X; either directly, or applying the linearity of expected values.

3. Given a problem involving probability,

   - Recognize independent events.
   - Apply the inclusion-exclusion principle to compute probability of a union.
   - Apply the law of total probability to split the probability of an event according to several cases.
   - Apply Bayes' theorem to compute the probability of an event E given event F.
   - Separate a random variable $X = X_1 + \ldots X_n$ as a sum of random variables X_i.

8 Relations and digraphs

1. Define what is

   - a relation from set A to set B,
   - a relation on A,
   - the property: reflexive, transitive, symmetric, antisymmetric.
   - an equivalence relation.

   Define what is a graph and a digraph.

   Define what is a path (closed walk, cycle, simple path, trail) in a (di)graph.

Recall that the relation R^n represents the paths of length n in the digraph corresponding to R.

2. Given a relation R on A determine

   - its representation as a digraph or as a matrix,
   - which properties R satisfy,

   Given a relation R from A to B determine its representation as a digraph or as a matrix,

   Represent problems with relations; e.g. student/class, city/state, etc.

3. Given relations $R \subset A \times B$ and $S \subset B \times A$, determine the relation $S \circ R$.

   Given a relation R on A determine

   - how to create a relation R^* that is reflexive (transitive) and $R \subseteq R^*$,
   - the relations R^2, R^3 and more generally, R^n.

9 Graph theory

1. For graphs with no loops and no multiple edges. Define what is

   - Connected component of a graph
   - The partition of vertices into connected components
   - The neighbourhood of a vertex
   - Isolated vertices
   - K_n, C_n, H_n and $K_{n,m}$
   - Bipartite graphs
   - Planar graphs
   - A face in a planar graph.

   For graphs with no loops and no multiple edges. Define what is

   - Eulerian path /circuit
   - Hamiltonian path / cycle
   - A complete Matching
   - A proper colouring of the vertices in the graph.

2. State the following theorems

   - The handshaking Theorem
   - Necessary and sufficient conditions for having an Eulerian circuit or path.
   - Ore's thm: Sufficient conditions for having a Hamiltonian cycle.
   - Hall's thm: Necessary and sufficient conditions for having a matching in a bipartite graph

- Euler's formula: about vertices, edges and faces of planar graphs.
- 4-colour thm: about coloring faces of planar graphs.

Provide examples of graphs satisfying the conditions of the theorems above.

Give examples of non-planar graphs (e.g. $K_{3,3}, K_5$).

3. Given a problem involving graphs, determine which of the theorems above can be applied.

10 Trees and their applications

1. Define what is

   - A tree, a rooted tree and a forest.
   - An m-ary tree and a full m-ary tree.
   - A leaf and an internal vertex.
   - The parent and children of a vertex.
   - The depth of a vertex and the height of a rooted tree.
   - A spanning tree of a graph G
   - A minimum spanning tree of a graph G with weighted edges.

2. Provide a formula for the number of internal vertices, leaves and edges in a full m-ary

   Explain the goal and give the pseudocode of the following algorithms

   - Depth-first search.
   - Breadth-first search.
   - Prim's algorithm.
   - Kruskal's algorithm.

   Provide applications for the algorithms above.

3. Given a problem involving trees, determine which of the theorems above can be applied.

11 Correspondence with sections in textbook

1. **Logic**
   - Section 1.1 Propositional logic
   - Section 1.2 Applications of propositional logic
   - Section 1.3 Propositional equivalences
   - Section 1.4 Predicates and Quantifiers
   - Section 1.5 Nested quantifiers
2. **Proofs and Induction**
   - Section 1.6 Rules of inference
   - Section 1.7 Introduction to proofs
   - Section 1.8 Proof methods and Strategies
   - Section 5.1 Mathematical induction
   - Section 5.2 Strong induction
   - Section 6.2 The pigeonhole principle
3. **Sets and Functions**
   - Section 2.1 Sets
   - Section 2.2 Set Operations
   - Section 2.3 Functions
   - Section 2.4 Sequences and summations
4. **Growth of algorithms**
   - Section 3.2 The growth of functions
5. **Number Theory**
   - Section 4.1 Divisibility and modular arithmetic
   - Section 4.3 Primes and Greatest common divisors
   - Section 4.4 Solving congruences
   - Sections 4.5-4.6 Applications of congruences
6. **Counting and combinatorics**
   - Section 6.1 The basic of counting
   - Section 6.3 Permutations and combinations
   - Section 6.4 Binomial coefficients and identities
   - Section 7.1 An introduction to discrete probability
7. **Probability**
   - Sections 7.2 Probability theory

- Section 7.3 Bayes theorem
- Section 7.4 Expected value

8. **Relations and graph representation**
    - Section 9.1 Relations and their properties
    - Section 9.3 Representing relations
    - Section 9.5 Equivalence relations
9. **Graph Theory**
    - Section 10.2 Graph terminology and special types of graphs
    - Section 10.4 Connectivity
    - Section 10.5 Euler and Hamilton paths
    - Section 10.7 Planar graphs
    - Section 10.8 Graph colouring
10. **Trees**
    - Section 11.1 Introduction to trees
    - Section 11.4 Spanning trees
    - Section 11.5 Minimum spanning trees

Extra: **Recurrence functions**

- Section 5.3 Recursive definitions
- Section 8.1 Applications to recurrence relations
- Section 9.4 Closure of relations