

good luck!  
♡

# Discrete Basics

## FINAL STUDY GUIDE

Let's review all math up to this point...

### NUMBERS

- Natural Numbers / Whole Numbers  $\rightarrow 1 < 2 < 3 < \dots$
- Integers  $\rightarrow \dots < -3 < -2 < -1 < 0 < 1 < 2 < 3 < \dots$
- Rational Numbers  $\rightarrow$  Numbers that can be written as an integer  $\div$  an integer
- Real Numbers  $\rightarrow$  All rational and irrational numbers
- Complex Numbers  $\rightarrow$  It's made w/ imaginary  $i$   $i = \sqrt{-1}$

### LOGIC

- Proposition  $\rightarrow$  a statement that is either True or False, T or F are truth values and are binary.
- Compound Propositions  $\rightarrow$  where  $p$  is a proposition...
  - Negation  $\rightarrow \neg p$  "not  $p$ ", switches truth value
  - Conjunction  $\rightarrow p \wedge q$ , "and  $q$ ", true if both are T
  - Disjunction  $\rightarrow p \vee q$ , "or  $q$ ", true if  $p$  or  $q$  or both are T
  - Exclusive Or  $\rightarrow p \oplus q$ , if  $p$  or  $q$  are true, but not both
  - Implication  $\rightarrow p \rightarrow q$ , "if  $p$ , then  $q$ ", "p implies q", true if  $p$  and  $q$  match or if  $p$  is F and  $q$  is T
  - Converse  $\rightarrow q \rightarrow p$ , opposite of implication
  - Inverse  $\rightarrow \neg p \rightarrow \neg q$ , "not  $p$  implies not  $q$ "
  - Contradiction  $\rightarrow p \wedge \neg p$ , opposite of inverse
- Predicate  $\rightarrow$  a proposition w/ a variable, ex.  $p(x)$
- Quantifiers  $\rightarrow \forall$  all,  $\exists$  at least 1, used to define domains

$p$	$q$	$\neg p$	$\neg q$	$p \wedge q$	$p \vee q$	$p \oplus q$	$p \rightarrow q$	$q \rightarrow p$	$p \leftrightarrow q$
T	T	F	F	T	T	F	T	T	T
T	F	F	T	F	T	T	F	T	F
F	T	T	F	F	T	T	T	F	F
F	F	T	T	F	F	T	T	T	T

$\rightarrow$  this is a truth table

$\rightarrow$  2 columns are the same we have logical Equivalences and we write  $p \rightarrow q \equiv \neg p \vee q$

### SETS

- Set  $\rightarrow$  an unordered collection of objects called elements.  $S = \{x \in \mathbb{Z} : p(x)\}$
- Set operations  $\rightarrow$  much like compound propositions...
  - Union  $\rightarrow A \cup B$ , A or B
  - Intersection  $\rightarrow A \cap B$ , A and B
  - Complement  $\rightarrow A^c$ , not A
  - Disjoint  $\rightarrow A \cap B = \emptyset$ , no overlap
  - Subset  $\rightarrow A \subseteq B$ , every element of A is an element of B

### FUNCTIONS

- A function is a mapping that associates a unique element in set A with an element in set B.
- Describing Functions...
  - One-to-One  $\rightarrow$  two elements in A will not be mapped to same element in B
  - Onto  $\rightarrow$  Every element in B has an element in A it maps to
  - Bijection  $\rightarrow$  One-to-One and Onto
  - Inverse  $\rightarrow$  we can find the inverse of a function if it's Bijection
  - Increasing  $\rightarrow$  for  $x < y \Rightarrow f(x) < f(y)$
  - Decreasing  $\rightarrow$  for  $x < y \Rightarrow f(x) > f(y)$
  - Binary Relation  $\rightarrow$  A special relation where elements in A can only match to 0 or 1
  - Reflexive  $\rightarrow$  if  $f(a,a) = 1$  for all  $a \in A$
  - Symmetric  $\rightarrow$  if  $f(a,b) = 1$  for all  $a, b \in A$
  - Transitive  $\rightarrow$  if  $f(a,b) = 1$  and  $f(b,c) = 1 \Rightarrow f(a,c) = 1$



### SUMS & SEQUENCES

- Sequence  $\rightarrow$  A function that maps every integer or positive integer to a new value.
- Sum  $\rightarrow$  when we add all elements of a typically an sequence together.
  - $\sum_{j=1}^n a_j = a_1 + a_2 + \dots + a_n$
  - $b_j = a_j$  is a constant,  $\sum = nc$
  - $a_j = b_j$ ,  $\sum = b_n - b_0$
  - $a_j = j$  and  $n=1$ ,  $\sum = \frac{n(n+1)}{2}$
  - $a_j = j^2$  and  $n=1$ ,  $\sum = \frac{n(n+1)(n+2)}{6}$
- Recursion  $\rightarrow$  A sequence is a recursion if  $f_0$  is defined and  $f_n$  is defined in terms of  $f_0, f_1, \dots, f_{n-1}$ . Famous example: Fibonacci
- $F_0 = 1, F_1 = 1, F_n = F_{n-1} + F_{n-2}$
- Induction is used to prove recursion.

# Graphs

Now let's explore graphs and networks...

### DEFINITIONS

- Graph  $\rightarrow$  a collection of edges and nodes  $G = (V, E)$
- Simple  $\rightarrow$  a graph is simple if it has no more than 1 edge between any pair of vertices and no loops
- Neighbors  $\rightarrow$  2 vertices are connected by an edge they're neighbors
- Neighborhood  $\rightarrow D(u)$  the set of all neighbors of a node
- Degree  $\rightarrow \deg(u)$  number of edges incident to  $u$
- Isolated node  $\rightarrow$  a node w/ degree 0
- Leaf node  $\rightarrow$  a node w/ degree 1
- Some special graphs...

- Complete graph  $\rightarrow K_n$ , every vertex is connected to every other vertex
- Cycle graphs  $\rightarrow C_n$ , vertices are arranged in a circle w/ an edge connecting every adjacent pair
- Wheel graphs  $\rightarrow W_n$ , a cycle graph w/ a vertex that every thing is connected to
- Bipartite graphs  $\rightarrow B_{m,n}$ , a graph w/ nodes in 2 sets and no nodes in A and nothing and vice versa
- Some Theorems...
  - $\sum \deg(u) = 2|E|$
  - $\rightarrow$  Every vertex w/ odd degree is even
  - $\rightarrow K_n$  will have  $n(n-1)/2$  edges
  - $\rightarrow$  simple graph w/  $n$  nodes will have  $\leq \binom{n}{2}$  edges
- Connected Graphs  $\rightarrow$  if there's a path from  $i$  to  $j$
- Connected Component  $\rightarrow$  a subgraph of  $G$  is a connected component if it's connected and not a subgraph
- A graph w/  $n$  nodes and  $n-1$  edges is always a connected w/  $(n-1)$  edges is always connected
- Trees  $\rightarrow$  a connected graph w/ no cycles
- Spanning Tree  $\rightarrow$  a subgraph of a connected graph  $G$  w/ the same vertices but no cycles and still connected

### OPERATIONS

- We have a graph  $G = (V, E)$
- Adding an edge  $\rightarrow$  we add an edge between 2 existing vertices,  $G' = (V, E \cup \{e\})$
- Deleting an edge  $\rightarrow$  remove any edge,  $G' = (V, E - \{e\})$
- Adding a vertex  $\rightarrow$  the new vertex is isolated,  $G' = (V \cup \{v\}, E)$
- Deleting a vertex  $\rightarrow$  we need to delete the vertex and any attached edges, so we get  $G' = (V - \{v\}, E - \{e\})$
- Union of graphs  $\rightarrow$  all vertices in  $G_1$  and  $G_2$  and all edges in  $G_1$  and  $G_2$  are in the new graph,  $G = V_1 \cup V_2, E = E_1 \cup E_2$
- Intersection of graphs  $\rightarrow$  a vertex and edge must show up in both  $G_1$  and  $G_2$  to be in the intersection,  $G = (V \cap V_2, E \cap E_2)$
- Complement of graph  $\rightarrow$  The complement formed by starting with the complete and removing every edge in the original graph.

### PATHS

- Path  $\rightarrow$  a sequence of edges from vertex  $i$  to  $j$  with no repeated edges or vertices
- Trail  $\rightarrow$  a path but it may have repeated vertices, but no repeated edges
- Walk  $\rightarrow$  a path but it may have repeated edges and vertices
- Cycle  $\rightarrow$  a path that starts and ends at same point, a closed path
- Circuit  $\rightarrow$  a closed trail
- Loop  $\rightarrow$  a closed walk

### DIRECTED GRAPHS

- directed edge (write  $e = (\text{start}, \text{end})$ )
- outdegree  $\rightarrow$  number of edges starting at vertex
- indegree  $\rightarrow$  number of edges ending at vertex
- sink node  $\rightarrow$  only incoming edges
- source node  $\rightarrow$  only outgoing edges
- Underlying Undirected Graph  $\rightarrow$  created by ignoring all directed edges
- Directed Acyclic Graph  $\rightarrow$  DAG, directed graph w/ no directed cycles
- At least 1 source and 1 sink node
- Have an algorithm to produce topological sort from DAG

### WEIGHTED GRAPHS

- A weighted graph is just like a normal graph w/ a weight assigned to each edge
- weights can represent things like distance, energy, or time weight
- Two common weighted graph problems...
  - shortest path  $\rightarrow$  we find the shortest path w/ the Dijkstra's algorithm. We choose a starting node and then...
  - Maximum weight spanning tree  $\rightarrow$  again, we find the shortest path but we connect all the nodes. We can do this by removing the highest weighted edges, while not making a disconnected graph, until we have no edges

# Proofs

Here's how we can use logical language to write mathematic proofs...

Proofs will come to us in the form  $p \Rightarrow q$  where  $p$  is a hypothesis and  $q$  is called the conclusion. We will look at 4 main types of proofs...

### DIRECT PROOF

- Assume  $p$  is true and construct a series of propositions where each is true until you get to  $q$ ...
- $p \Rightarrow p, p \Rightarrow p_2, \dots, p_n \Rightarrow p_n, p_n \Rightarrow q$

### CONTRADICTION

- We assume  $q$  is false and hope that to show  $p$  is false, which proves  $p \Rightarrow q$  since...
- $p \Rightarrow q \equiv \neg q \Rightarrow \neg p$

### CONTRADICTION

- Assume  $p$  is true and  $q$  is false, construct a series of propositions where all are true but  $p_n$ ...
- $p \wedge q \Rightarrow p, p \Rightarrow p_2, \dots, p_n \Rightarrow p_n$
- This shows  $p \wedge q$  is false which will mean that  $p \Rightarrow q$ .

### INDUCTION

- We want to prove  $p \Rightarrow q$  for  $n = 0$  or  $n \in \mathbb{N}$ . We obviously can't prove each  $n$  indiv. so we use induction.
- Let's call our hypothesis  $P(n)$ . We will prove in two steps...
  - prove  $P(1)$  is true
  - prove for any  $n \geq 1$ , if  $P(n)$  is true then  $P(n+1)$  is true.

$$P(1) \wedge (\forall n \geq 1, P(n) \Rightarrow P(n+1)) \Rightarrow \forall n \in \mathbb{N}, P(n)$$

# Combinatorics

For probability we're gonna want some tricks for counting in a timely manner

### PRINCIPALS

- If we have 2 independent stages we multiply the # of things to find the total #. i.e. we have 2 different things we add the things. This is the Product and Sum Rule.
- Inclusion Exclusion Principle  $\rightarrow$  the # of items in the union of 2 sets is...
- $N(A \cup B) = N(A) + N(B) - N(A \cap B)$
- Pigeonhole Principle  $\rightarrow$  if  $n$  objects are placed into  $m$  boxes, where  $n > m$ , at least 1 box contains  $\lceil n/m \rceil$  or more objects.  $\lceil x \rceil$  is the smallest integer  $\geq x$

### COMBINATORICS

- Permutation (of Distinct Obj)  $\rightarrow$  An ordered arrangement of objects of length  $n$ ...
- $P(n, n) = \frac{n!}{(n-n)!}$
- Circular Permutations  $\rightarrow$  permutations in a loop...
- $\frac{(n-1)!}{n!}$
- Permutations of non distinct obj.  $\rightarrow$  a permutation of  $n$  obj,  $m_i$  of which are of type  $i$ ...
- $\frac{n!}{m_1! m_2! \dots m_k!}$
- Combinations  $\rightarrow$  An arrangement of  $r$  length, chosen from  $n$  distinct obj. where order doesn't matter.
- $C(n, r) = \frac{n!}{r!(n-r)!}$

### BINOMIALS

- We use binomials to expand  $(x+y)^n$ , but binomials will look good for probability reason...
- $\binom{n}{k} = \frac{n!}{k!(n-k)!} = C(n, k)$
- Fun fact Pascal's Triangle gives us binomial coefficients.

# Probability

Let's use these ideas to find probability of things.

### DEFINITIONS

- Sample Space  $\rightarrow$  all possible outcomes of a random experiment
- Event  $\rightarrow$  subset of the sample space
- Let's use these ideas to talk about prob. of events.

### PROB. OF EVENTS

- The probability of an event will be...
  - $P(A) = \frac{N(A)}{N(S)}$  if  $n$  eq. outcomes in A
  - $\dots$  if  $n$  eq. outcomes,  $\rightarrow S$
  - otherwise...
- When we have a conditional probability we have...
  - $P(A|B) = \frac{P(A \cap B)}{P(B)}$
  - "Prob. happens given B happened"
  - Two events are independent when...
    - $P(A|B) = P(A)$  or  $P(A \cap B) = P(A)P(B)$
    - The Law of Total Prob. says...
- $P(A) = P(A \cap B) + P(A \cap B^c)$   
 $= P(A|B)P(B) + P(A|B^c)P(B^c)$
- And Bayes' Rule says...
- $P(B|A) = \frac{P(A|B)P(B)}{P(A)}$

### RANDOM VARIABLES

- A Random Variable is a function from the sample space to the set of real numbers.
- $X: S \rightarrow \mathbb{R}$
- RV Sample Space  $\rightarrow$  Probability distribution
- $p_X(x) = P(X=x) = \sum_{s \in S: X(s)=x} p(s)$
- Binomial RV  $\rightarrow$  prob. of  $n$  successes in  $n$  indep. trials...
  - $p_X(k) = \binom{n}{k} p^k (1-p)^{n-k}$
- Expectation  $\rightarrow$  The average  $X$  gives the prob. func.
  - $EX = \sum_{k \in A} x_k p_X(x_k)$
- Some properties of expectation...
  - $\forall x, y \Rightarrow EX + y = aEX + b$
  - $\forall x, y \Rightarrow EX + y = EX + EX + EX + \dots$
  - And  $\forall$  indep.  $\rightarrow EX + Y = EX + EX$