6.042[J] Notes (Page 1)

Introduction

- Proof is a method of establishing truth

→ Mathematical Proof of a proposition is a chain of logical deductions leading to proposition from base of axioms

1) Proposition
2) Logical Deduction
3) Axiom

Propositions

Def | • A statement true or False

Compound Propositions

→ can combine & relate Propositions

→ Classify propositions as T/F (Boolean Variables)

NOT / AND / OR

P	Q	Not(P)	P OR Q	P AND Q
T	T	F	T	T
T	F	F	T	F
F	T	T	T	F
F	F	T	F	F

IMPLIES

* AN IMPLICATION is True when if-part False, or then-part True

... IF P, Then Q

P	Q	P IMPLIES Q
T	T	T
T	F	F
F	T	T
F	F	T

IFF

• "If and only If"

P	Q	P IFF Q
T	T	T
T	F	F
F	T	F
F	F	T

Notation

NOT(P)	$\neg P$
P AND Q	$P \wedge Q$
P OR Q	$P \vee Q$
P IMPLIES Q	$P \rightarrow Q$
If P Then Q	$P \rightarrow Q$
P IFF Q	$P \leftrightarrow Q$

Logically Equivalency

∆ cast in propositional logic [8]

→ If truth tables the same, they're logically equivalent

(1) Implication equivalent to contra-positive

(2) Implication NOT equivalent to Converse

Propositional Logic in Programming

ex | `if(x>0 || (x<=0 && y>100))`

↳ "||" = OR

"&&" = AND

CAN BE SIMPLIFIED

`if(x>0 || y>100)`

• simplifying logical expressions difficult but important

Predicates & Quantifiers

Propositions w/ inf many cases

→ Some Propositions involve Large or Infinite checks

ex | n^2+n+41 is prime for any [n] (fails at n=40!)

Key: can't prove an infinite proposition w/ finite set

Set Notation: $\forall n \in \mathbb{N}.\ P(n)$ prime

Predicates

→ Proposition whose truth depends on value of one or more variables

ex | "n is a perfect square"

$P(n) :=$ "n is a perfect square"

Quantifiers

- Predicate can be sometimes or always True

Always True

- For all n, P(n) is True

(set Notation) For all $x \in \mathbb{R}$, $x^2 > 0$

Sometimes True

- P(n) is true for some n

(set notation) $5x^2 - 7 = 0$ for some $x \in \mathbb{R}$

Predicate Notation

$\forall$ → For All

$\exists$ → There exists some value

* Both symbols always Followed by Variable & set variable ranges over

Order of Quantifiers

* swapping order can change meaning!

ex. "Every American has a dream"

- $\exists d \in D. \forall a \in A, H(a,d)$ (same dream)
- $\forall a \in A. \exists d \in D. H(a,d)$ (personal dream)

ex. GoldBach's in reverse order is patently False

Variables over one Domain

→ when all variables take values from non-emety set D, omit D

Negating Quantifiers

$\text{Not}(\forall x. P(x)) \iff \exists x. \text{Not}(P(x))$

Validity

→ Propositional Formula True, if T no matter what value assigned to individual Proposition vars

Satisfiability

→ satisfiable if some setting of variables makes proposition True

ex

$P \wedge \bar{Q}$ → satisfiable

$P \wedge \bar{P}$ → not Satisfiable

P vs. NP → whether the SAT truth table method can be solved in Polynomial Time

* CHAPTER 2 *

Axiomatic Method

- A Proof is a sequence of logical deductions from axioms & previous proofs that conclude w/ proposition

Terminology

1. Theorem → Important Proposition
2. Lemma → Preliminary Proposition useful later
3. Corollary → proposition that follows from Lemma or Theorem

The Grand Base Set: Zarmelo- Frankel set theory w/ Choice

Our Axioms

- ZFC → too Primitive

Δ take high school math facts as axiom

Logical Deductions

- Inference Rules for proving new propositions using previous proved old ones

Modus Ponens

- Proof of P & P→Q is a Proof of Q

↳ there are other sound inference rules as well

6.042[J] Notes (Page 2)

Proof templates

- Many Proofs follow a handful of standard templates

Proof by Cases

→ Breaking more complicated Proof into cases is a common strategy

Example

- If every Pair of people in a group have met, they are a club
- If not group of strangers

Theorem: Every collection of 6 people includes club of 3 or 3 strangers

Define Person x

Case 1: x met ≥ 3

↳ 1.1 No one have met eachother (strangers)

↳ 1.2 at least one pair met (∃club)

Case 2: x met < 3

↳ 2.1 if all have met, (∃club)

↳ 2.2 if more than one pair not met (∃strange)

Proving an Implication

Form: If-Then, $P \rightarrow Q$

Method #1: Assume P is True

- when Proving $P \rightarrow Q$, 2 cases to consider
  - P is True (More Interesting)
  - P is False

→ SO, assume P, & show Q logically follows

AVOID: If you assume P true for one proof, don't assume for all

Method 2: Prove the contra Positive

- $P \rightarrow Q$ EQUIVALENT $\overline{Q} \rightarrow \overline{P}$

→ SO, write "Prove the contra-positive & proceed"

Ex| If r irrational, $\sqrt{r}$ irrational

contr: if $\sqrt{r}$ rational, r rational

$\sqrt{r} = \frac{a}{b} \rightarrow r = \frac{a^2}{b^2}$ (integer) (integer)

Proving "If and Only If"

Method #1: Prove each statement implies the other

P iff Q $(\Leftrightarrow)$ $\{ P \rightarrow Q, \; Q \rightarrow P \}$

Method #2: Construct a chain of IFF's

Proof by Contradiction

→ Here if a proposition is False, you show False Fact to be True

1) write Proof by Contradiction
2) Write, "Suppose P is False"
3) Deduce something known to be False
4) This is contradiction, therefore P must be true

Proofs About Sets

- A set is a bunch of objects contained called elements

→ order doesn't matter

→ no notion of element appearing multiple times

Popular Sets

symbol	set	Elements
$\emptyset$	Empty set	None
$\mathbb{N}$	Non-negative Integers	$\{0, 1, 2, 3, \ldots\}$
$\mathbb{Z}$	Integers	$\{-2, -1, 0, 12\}$
$\mathbb{Q}$	Rational Numbers	$\frac{1}{21}, -\frac{5}{3}, \frac{13}{12}$
$\mathbb{R}$	Real Numbers	$\pi, e, -9, \sqrt{2}$
$\mathbb{C}$	Complex Numbers	

superscripts

D^+ → only Positive Elements

D^- → only negative Elements

Comparing & Combining Sets

$S \subseteq T$ → "S is subset of T"

$X \cup Y$ → "Union of elements in sets X & Y"

$X \cap Y$ → "Intersection of sets X & Y"

$X - Y$ → "Set Difference, only elements in X not in Y"

Complement of a Set

- For subet A of Domain D, it's useful to look at $\overline{A}$ (All elements in D, not in A)

$$\overline{A} = D - A$$

Cardinality

→ # of element in set $|A|$

The Power Set

- ** set of all subsets in a set

$P(A)$ of A

→ $B \in P(A)$ iff $B \subseteq A$

$$|P(A)| = 2^{|A|}$$

Sequences

→ (a, b, c)

Key differences

- Sets can have duplicate elements
- Terms in sequence have order
- λ = empty sequence, ϕ = emety set

Cross - Products

→ link between sets & sequences

- Product of sets is a new set of sequences where 1st element from first set, 2nd element from second set etc

ex

$$\mathbb{N} \times \{a, b\} = \{(0,a), (0,b) \ldots\}$$

→ Product of n copies of set denoted S^n

ex

Set Builder Notation

→ useful for sets not easily defined listing elements explicitly

* So we define set using Predicate

ex

$$A ::= \{n \in \mathbb{N} \mid n \text{ is prime} \wedge \text{"}\ldots\text{"}\}$$

$$B ::= \{x \in \mathbb{R} \mid x^3 - 3x + 1 > 0\}$$

$$C ::= \{a + bi \in \mathbb{C} \mid a^2 + 2b^2 \leq 1\}$$

Proving Set Equalities

* Two sets are equal if they contain same elements

Distributive Law of sets

$$A \cap (B \cup C) = (A \cap B) \cup (B \cap C)$$

Good Proofs in Practice

1. State your Game Plan
2. Keep Linear Flow
3. * Proof is an essay, not a calculation *
4. Simplify & avoid excessive symbolism
5. Introduce Notation Thoughtfully

Section #3: * Induction *

The Well Ordering Principle

- "Every Non-Emety set of Non-Negative integers has a smallest element"

Well Ordering Proofs

- One of most important proof rules in Discrete Math

→ Prove $P(n)$ true for all $n \in \mathbb{N}$ using well ordering Principle

1) Define set C of counterexamples of P being True

$$C ::= \{n \in \mathbb{N} \mid P(n) \text{ False}\}$$

2) Use a proof by contradiction & assume C is non-emety

3) By well-ordering Principle, there will be smallest element n in C

6.042[J] Notes (Page 3)

Well ordering Proofs (cont.)

4) reach a contradiction by using m to find a member of C smaller than m (or any contradiction)

5) Conclude C must be empty & no counter examples exist QED

Ordinary Induction

- Let P(n) be a predicate

If,

- P(0) is true
- P(n) IMPLIES P(n+1) for all non negatives n

then

P(m) is true for all non-negative integers m

Template for Induction Proofs

1) State the proof uses induction

2) Define Appropriate Predicate P(n)
"induction hypothesis"
→ if multiple vars, specify n

3) Prove Base Case $P(0)$

4) Prove $P(n) \rightarrow P(n+1)$

Challenging Example

- Assume a courtyard of Dimensions $2^n \times 2^n$
- One of central squares must have Statue of Donor
- L-shaped Tiling

(naïve) P(n): There exists a tiling of $2^n \times 2^n$ courtyard with Bill in a central square
{ P(0), fine
P(n)→P(n+1), Run into trouble }

*First Fallback: Look for a Stronger Induction Hypothesis

(correct P(n)): There exists a tiling of $2^n \times 2^n$ with Bill any where

Invariants

▲ A property that is preserved through a series of operations is called an Invariant

→ Induction typically used to prove invariants

- Show true at beginning
- true at step t → $t+1$

The Invariant Method

- Define P(t) to be predicate that NICE holds immediately after step t
- Show P(0) to be TRUE
- show $\forall t \in \mathbb{N}.\ P(t) \rightarrow P(t+1)$

Challenge Problem: The 8-Puzzle

a)

A	B	C
D	E	F
#	G	

b)

A	B	C
D	E	F
#		G

P(n): No sequence of legal moves change b) → a)

Lemma1) A row move does not change order of tiles

Lemma2) A column move changes relative order of exactly two pairs of Tiles

- Show that # inversions odd/even are invariant

Strong Induction

- Let P(n) be a predicate

→ If P(0) is True

→ for all $n \in \mathbb{N}$
$\{P(0), \ldots, P(n)\} \rightarrow P(n+1)$

then P(n) true for all $n \in \mathbb{N}$

ex. Making Change

Proof → can make change for anything greater than 8 w/ (3,5)

P(0): 5+3=8

P(n+1=1) 3·3=9

P(n+1=2) 2·5=10

P(n+1=3) 5+2·3=11

ex. Stacking Game

- start with stack of [n] boxes

→ Divide into two non-zero stacks

Theorem: Any way of unstacking gives

Structural Induction

Recursive Data Types

- Datatypes specified by by Recursive Definitions

1) Base Case → Don't depend on anything

2) Constructor Case → depends on previous cases

ex. String of Brackets

brkts → set of all square Bracket strings

Base case: [λ → {empty}]

Constructor: if s ∈ brkts, then {s] & s[} in brkts

ex. Arithmetic Expressions

(*) $3x^2+2x+1$

** Unit 4: Number Theory **

* we concern ourselves with variables ∈ ℤ

Divisibility

- The divides relation

> a divides b iff ak=b

→ Number theory contains simple questions w/ very difficult answers

Famous Conjecture in Number Theory

1) Fermat's Last Theorem

→ There are no x,y,z s.t $x^n+y^n=z^n$

2) Goldbach Conjecture

→ every integer >2 is the sum of 2 primes

3) Twin Prime Conjecture

→ infinitely many instances where if p is a prime, p+2 is as well

4) Primality Testing

5) Factoring

[q] = quotient

[r] = remainder of n divided by d

ex.

Quotient(2716, 10) = 271

remainder(2716, 10) = 6

Particular class of Problem: can one form [g] gallons with jugs of capacity (a, b)?

Finding Invariant Property

- suppose we've two jugs with capacities [a] & [b] where b > a

- (x, y) → state of system of jugs $0 \le x \le a$, $0 \le y \le b$

- water in each jug takes form $s \cdot a + t \cdot b$

for $\exists s,t \in \mathbb{N}$

Greatest Common Divisor

- Denoted gcd(a, b)

Thm: The greatest common divisor of a & b is equal to smallest positive linear combo of a & b

6.042 [J] Notes (Page 4)

Proof:

→ By the Well Ordering Principle There exists a smallest positive linear combo of a & b called [m]

Prove: $m = \gcd(a,b)$

Showing: ① $m \le \gcd(a,b)$

② $m \ge \gcd(a,b)$

① for any common divisor [c]

$c \mid sa+tb$

$\gcd(a,b) \mid sa+tb$

$\gcd(a,b) \mid m$

↓ (implies)

$\gcd(a,b) \le m$

② Show $m \mid a$ & $m \mid b$

recall: $a = q \cdot m + r$

↓ $m = sa+tb$

$r = (1-qs)a + (-qt)b$

BECAUSE $0 \le r < m$

[&]

m is smallest positive linear combo

implies [r must be zero!]

Properties of Greatest common Divisor

1. Every common Divisor of a & b divides gcd(a,b)
2. $\gcd(ka, kb) = k \cdot \gcd(a,b)$
3. If $\gcd(a,b)=1$ [&] $\gcd(a,c)=1$ $\gcd(a,bc)=1$
4. If $a \mid bc$ and $\gcd(a,b)=1$, then $a \mid c$
5. $\gcd(a,b) = \gcd(b, \text{rem}(a,b))$

Proving Trick: gcd → linear comb

The Pulrerizer

Problem: Non-Descript way to solve for coeffs [s] & [t]

Procedure: Goes through same steps as Euclid's alg

- w/ additional bookkeeping step to write remainders as linear combos

* Last non-zero remainder is gcd

Fundamental Theorem of Arithmetic

- Every positive integer [n] can be written as a unique product of primes

Modular Arithmetic

- [a] is congruent to [b] modulo [n] iff $n \mid (a-b)$

$a \equiv b \pmod n$

$a \equiv b \pmod n$ iff $\text{rem}(a,n) = \text{rem}(b,n)$

Alternative visualization { congruence modulo n defines a partition of integers into [n] sets all congruent w/ one another

Turing's code / Arithmetic w/ prime Modulus

① sender & receiver agree on large prime [p] & secret key $K \in \{1, 2, \ldots, p-1\}$

② The message can be any integer in set $\{0, 1, 2, \ldots, p-1\}$

③ De-encryption Difficulty

m^* is a remainder

$m^* = \text{rem}(mK, p)$

Arithmetic w/ a Prime Modulus

- Multiplicative inverses exist working in modulo a prime

ex $7 \cdot 3 \equiv 1 \pmod 5$

3 = multiplicative inverse of 7

Key: can only recover Turings message if multiplying by multiplicative inverse of Key

$$m^* \cdot K^{-1} = \text{rem}(mK, p) \cdot K^{-1}$$

$$\equiv$$

Arithemtic w/ Arbitrary Modulus

- a & b relative primes if $\gcd(a, b) = 1$

Key: modulo n may be complicated, but #'s relatively prime to n behave well

RSA Algorithm

1. Receiver creates public & secret Key
   - Generate primes p & q
   - Let $n = pq$
   - select e s.t $\gcd(e, (p-1)(q-1)) =$

** Unit 5: Graph Theory **

- The graph is most important structure in CS
- A graph is a collection of lines & dots where lines connect some pairs of dots

Simple Graphs

- A simple graph G consists of a non-emety set V of vertices and set E consisting of 2-elements of V
- E is edges of G
  → written $G(V, E)$

ex)

$V = \{a, b, c, d, e\}$

$E = \{\{a, b\}, \{b, c\}\}$

* Note: for unordered sets

$\{a, b\} = \{b, a\}$

Definition

- vertices adjascent if joined by an edge
- The Degree of a vertex is # of edges incident to it

Some Common Graphs

K_n → complete graph on n vertices

L_n → Line graph of (n-1) edges

C_n → cyclic graph (n vertices, n edges)

Isomorphism

- If $G_1(V_1, E_1)$ & $G_2(V_2, E_2)$ are two graphs, then we say G_1 is **isomorphic** to G_2 iff there exists bi-jection

$$f: V_1 \to V_2$$

Key: Properties of graph preserved under isomorphism, even if vastly different looking

Sub-Graphs

- G_1 is sub-graph of G_2 if $V_1 \subseteq V_2$ & $E_1 \subseteq E_2$

Weighted Graphs

- where an edge holds a capacity / weight
- Simple graph but with weight function w

$$w: E \to \mathbb{R}$$

Adjascency Matrix

- Matrix form representation of adjascency between nodes

→ $[0, 1]$ Simple graphs

→ $[0, w(\{v_i, v_j\})$ if $\{v_i, v_j\} \in E]$

Matching Problems

- The handshaking Lemma

The sum of degrees of vertices equals twice the # of edges

Bipartite Matchings

→ A graph with a partition of it's vertices into two sets L & R where every edge incident to vertex in both

The Matching condition

- Hall's matching theorem:

↳ necessary and sufficient conditions for existence of matching in bi-partite graph

6.042[J] Notes (Page 5)

Theorem: A matching for set of men [M] with set of women [W] can only be found if matching condition holds

** Each subset of Men [M] must like as large a set of women **

Proof (see pg. 135)

An easier Matching condition

△ checking all subsets can get tedious,

solution

[G], a bi-partite graph is degree constrained, if for vertex partition L, R where $|L| \leq |R|$, $deg(l) \geq degree(R)$ for every $l \in L$ or $r \in R$

Proof by contradiction

Suppose [G] is degree constrained but there is no matching. So there must be bottleneck

- Let x be s.t $deg(l) \geq x \geq deg(r)$

$|N(S)| \, x \geq |S| x$

→ [S] not a bottleneck

Q.E.D

Stable Marriage Problem

- $|L| = |R|$
- every l, r has preference Per set
- Rankings are complete
- Stable → No pairs ~~want to~~ that ~~want~~ prefer others to spouses

Mating Ritual

→ Gaurenteed Stable Matching in Polynomial Time

1. Each male proposes to best choice on list
2. each woman asks her fave to return, tells rest to leave
3. Non-selected males cross woman off list
4. when day arrives where each woman has at most one suitor, ritual ends

Facts to prove

A) Ritual reaches termination condition

B) Every body ends up married

C) Resulting marriages are stable

[A] - every day the ritual runs (& hasn't terminated) at least one man crosses woman off list

- if [n] men & [n] women

→ max # of days procedure runs = $[n^2]$

[B] i) Show invariant by Induction on n days

that (if w is crossed off m's list, w has better suitor)

By contradiction

- Assume there are one man & one woman at end unmarried

Solution Optimality for Men

- Suprisingly, but men do get top choice & quality decreases just enough to be stable

* There are multiple stable matchings in existence

Proof of optimal case for men: Definition of optimal spouse & rogue couple compromises existence of stable matching (contradiction)

Proof of Pessimal case for women (contradiction) | non-existence of stable matching w/ woman she likes less

Coloring

Motivation:

Before edges denoted affinity

Now edges denote conflict

Example

Points = events

Edges = schedule conflict

constraint = Tile such that NO Adjacent points have same coloring

Chromatic Number

- Min value of k for which graph has valid k-coloring

Degree-Bounded Coloring

- Bi-Partite → 2 coloring

Graph of max degree k is $k+1$ colorable

- Complete Graph

best coloring is 4-coloring

2-coloring!

Why Coloring?

- updates & pairwise criticality (min waves to disruption performance)
- Allocating registers in program variables
-

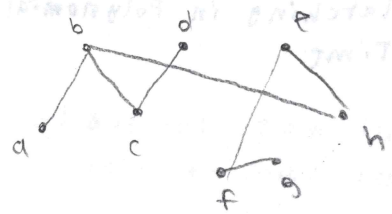
Getting from A → B

- A walk in graph G is a series of vertices $v_0, v_1, \ldots, v_k$ and edges $\{v_0, v_1\}, \ldots \{v_{k-1}, v_k\}$

Definitions:

- starts at v_0
- ends at v_k
- length = k
- For Paths, no v_i the same

ex



Finding a Path

- Where there is a walk there is a path (well-ordering Principle)

contradiction: shortest walk isn't path

Number of walks

- Can be multiple walks of length k to connect two-points

Relationship between walks of length k and k^{th} power of adjacency A_G

The Relationship

- Let $G = (V, E)$ be an n-node graph, $V = \{v_1, v_2, \ldots, v_n\}$
- $A_G = \{a_{ij}\}$
- # of length k walk between v_i & v_j = $a_{ij}^{(k)}$

Key observation: each power raise of $A_{(G)}$ is a discovery of # of paths of that increment multiplication (sub-paths of $k-1$ at your disposal, original adjacency matrix indicates whether valid or not) wow!

Proof Induction on k

$P(k): \forall i, j \in [1, n]. P_{ij}^{(k)} = a_{ij}^{(k)}$

$$P_{ij}^{(k+1)} = \sum_{t=1}^{n} a_{ij}^{(k)} a_{tj}$$

exactly as we guessed above

Connectivity

Defintion: Two vertices are connected if a path begins at one & ends at the other

- A graph is connected if each pair of vertices connected pairwise

Connected Components

→ subgraph within graph fully connected

k-connected Graphs

→ takes k edge failures to make dis-connected

MIN(#) edges in connected Graph

theorem Every graph w/ V vertices & E edges has at least $V-e$ connected components

Induction on E & cases

6.042[J] Notes (Page 6)

Nuance: Proof on Graphs

① Induction on edges & vertices

② Start w/ nth node graph

Shrink-down, grow back

↳ Done to avoid Build-up error

Build-up Error

△ Faulty Assumption that EVERY (n+1) graph can be formed from any (n) graph satisfying condition w/ same property

Cycles & closed walks

- A closed walk is a sequence of vertices & edges where

first node = last node

length = (k)

- only $\{v_0, v_k\}$ are the same

Odd-Cycles & 2-colorability

Following Properties Mutually Inclusive

① Graph Bi-Partite

② Graph is 2-colorable

③ Graph contains no cycles of odd length

④ No closed walks of odd length

Euler Tours

- A walk that traverses every edge exactly once

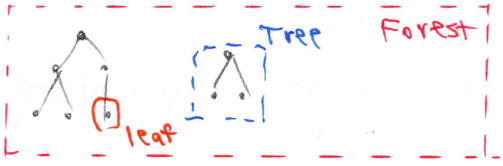
Hamiltonian Cycles

- A cycle that visits every node in (G) exactly once

Travelling Salesperson Problem

Trees

- A connected, acyclic graph is a tree
- If every connected component of (G) is a tree, G is a Forest
- A node, is a vertex in tree of degree (1)



Spanning Trees

Theorem: every connected Graph has spanning tree

Proof: By contradiction

- call (T) minimally connected sub-graph but not a tree

Ⓐ contains a cycle

→ so removing edge

Min-Weight Spanning Trees

→ Spanning tree of minimum weight

Pg. 173 → two algorithms to solve for MST

Proof on 175 shows

1) existence of edge to form tree

2) termination condition

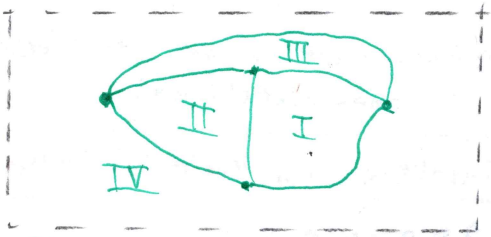
Planar Graphs

- A graph is planar if no curves cross

Recursive Definition of Planar Graphs

- Face → Continuous regions denoted by smooth edges
- Outside face extends off into infinity

Note: vertices forming boundary of each face form a cycle



Planar Graph w/ four faces

- Cycles & closed walks form Discrete faces of the drawing

Recursive Def: Planar Embeddings

- A **Planar Embedding** of a connected graph is a non-emety set of closed walks that make up Discrete Faces

Euler's Formula

- Recursive Definition provides powerful technique for proving properties of planar graphs

Thm } If a connected Graph has planar embedding, then

* $v - e + \#_{faces} = 2$

- Proof of structural induction over constructors

Chapter #6: Directed Graphs

- Endpoints now Distinguished

- Definition: A directed Graph $G(V, E)$ is a set of Nodes & Edges specified by an ordered pair of vertices

⇒ Applicable where relationship is one-directional

Parallels to Undirected Graphs

Degrees

- Indegree (←)
- Outdegree (→)

Walks & Paths

- Hamiltonian reaches every node

Connectivity

- strong connectivity
- weak connectivity

DAG → Directed Acyclic Graph (useful in scheduling & optimization)

Tournament Graphs

- [n] players compete in round-robin tournament
  → pairwise each node has played & beat one-another

Graphical form: Edge direction represents outcome

Finding Hamiltonian Path in Tournament Graph

- There exists a ranking such that each player lost to the player one position higher

Thm: Every Tournament Graph has directed Hamiltonian Path (strong Induction over vertices [n])

Base case

- P(1) trivially true, path of length 0

Inductive Step

- Assume P(0)... P(n) true
- Choose vertex [v] arbitrarily and partition rest into 2 sets
  - [T] → "pointing to v"
  - [F] → "pointing from v"

The King-Chicken Theorem

- Suppose [n] chickens are in farmyard
- Chicken u pecks v if
  1) chicken u pecks v
  or
  2) u pecks w who pecks v

Communication Networks

Cool! Pay Attention

- Directed Graphs essential to model communication networks

vertices: Computer, Processor, switch
Edge: wire, fiber, transmission line

Packet Routing

- data can be communicated in packets (Fixed amounts)

Assumption

[1] packet at each input
[1] packet destined for each output

Destination of Packets is permutation (π): $1, \ldots, N-1$

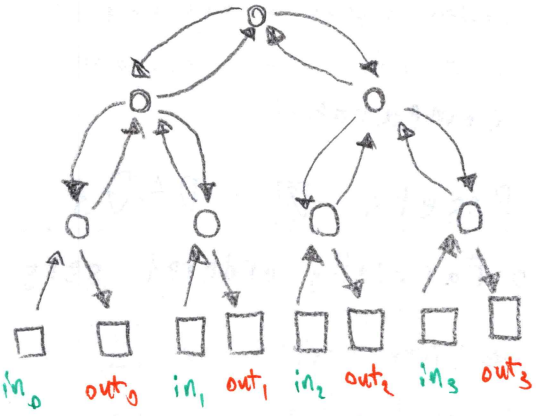
Goal: to get all packets to destination w/ as little hardware as possible

6.042 [J] (Page 7)

Complete Binary Tree

- n inputs, n outputs

ex



Key

□ = terminals

○ = switches

Net work Diameter

Latency → time between a packet arriving between input & output

Diameter → Length of shortest path between probable input/output

Switch Size

- One can reduce network diameter using larger switches

Goal: keep the switch # count as low as possible

Con gestion

- Fatal Flaw of Binary Tree

→ Passing all packets through single switch prone to hold-up

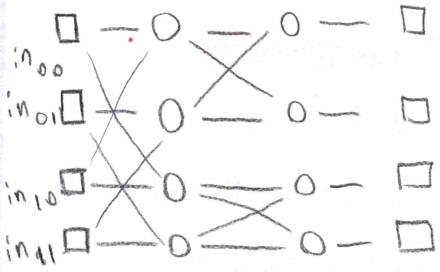
- Congestion will be a function of permutation problem i

Maxi-Min Problem dictates

The Butterfly

- Ideal solution would combine few switches & low congestion

ex lvl-1 lvl-2



Characteristics

- All terminals & switches in N rows
- There exis $\log_2(N)+1$ number of levels of switches

→ binary $b_0 b_1 \ldots b_{\log(N)}$ denotes binary indicator on crossing diagonals

Chapter #7: Relations & Partial Orders

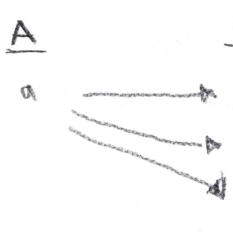
Binary Relations

- Given sets A & B, a Binary Relation $R: A \to B$ is a subset of $A \times B$

Notation: aRb, $a \underset{R}{\rightsquigarrow} b$

Function vs Relation

- ALL Functions are Relations
- Not ALL Relations are Functions
  - ↳ can be multiple elements in B to single element a



Relational Images

- The Image of set Y under relation $R: A \to B$, written $R(Y)$

$$R(Y) := \{b \in B \mid yRb \text{ for some } b \in B\}$$

→ sean of all possible outcomes of apply ing relation

Inverse Relations & Images

- Inverse R^{-1} of $R: A \to B$ defined by rule

$$bR^{-1}a \quad \text{IFF} \quad aRb$$

Combining Relations

(1) composition of Relations

$(R \circ S): A \to C$

$$a(R \circ S)c \text{ IFF } \exists b \in B \; \begin{matrix} bRc \\ \& \\ aSb \end{matrix}$$

2) Product of Two Relations

$R_1 : A_1 \to B_1$

$R_2 : A_2 \to B_2$

$S = R_1 \times R_2$

Relations & Cardinality

- Surjective & Injective Relations

1) Surjective IF every element of B assigned to at least one element of A

2) total IF every element of A assigned to some element of B

3) Injective if every element in B mapped at most one

4) Bijective if R is total, surjective, injective, & a function

Cardinality

Key Point: Relational Properties useful in figuring out size of domains & co-domains

$|A|$ = Cardinality of A

Relations on One Set

- All relations on single set A take form of subset $R \subseteq A \times A$

(can be represented as di-graph)

Key Concept: Properties of Relation $R : A \to A$

1) Reflexivity

Math| $\forall x \in A.\ xRx$

Graph| every node has loop

2) Symmetry

$\forall x, y \in A.\ xRy \to yRx$

3) Anti-symmetry

$\forall x, y \in A\ (xRy \text{ AND } yRx) \to x = y$

Graph| No distinct pair of people like eachother

4) Asymmetry

Math| $\neg \exists x, y \in A.\ xRy \text{ AND } yRx$

Graph| No one likes themselves, and no one likes eachother

5) Transitivity

Math| $\forall x, y, z \in A$

$xRy \text{ AND } yRz \to xRz$

Partitions

- A partition of finite set A is a collection of disjoint subsets whose product of union is A

$A_1 \cup A_2 \cup A_3 = A$

(A_1, A_2, A_3 = partitions)

Partial Orders

- Strong & weak Partial orders

→ A relation R is a weak Partial order if

- Transitive
- Anti-symmetric
- Reflexive

symbol: $\preceq$ (weak partial order) $\prec$ strong

Total Orders

- A partial order is **partial** b/c there can be two elements in A w/ no relation

Comparable if $(a \preceq b \text{ or } b \preceq a)$

- A Total Order is a partial order where every pair of distinct elements are comparable

Posets & DAGs

o Partially ordered sets

→ Given partial order $\preceq$ on set A, the pair $(\preceq, A)$ is a Poset

o Posets are Acyclic

- A Poset has no directed cycles other than self-loops

Transitive Closure

Every Poset → DAG. Is the reverse True?

Yes, but we must modify by adding edges to satisfy transitivity

Given| Di-Graph $G(V, E)$

the transitive closure is di-graph $G^+(V, E^+)$

where, $E^+ = \{u \to v |$ Add edges s.t Transitivity preserved

6.042 [J] (Page 8)

The Hasse Diagram

Problem: viewing a Poset as a Di-Graph, tends to be alot of edges

→ But we need not write all edges implied by transitivity

Hasse Diagram: Minus

① Self Loops

② edges implied by transitivity

Topological Sort

A Topological sort is a total order s.t

$$x \preceq y \longrightarrow x \preceq_T y$$

Parallel Task Scheduling

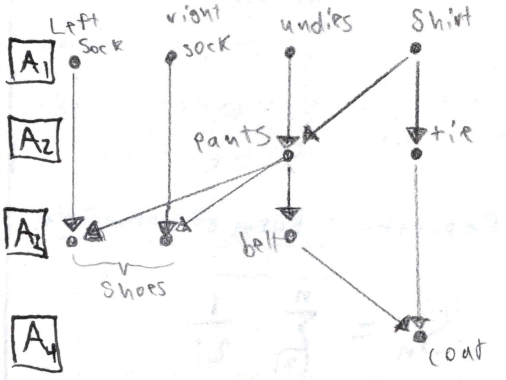
Poset → Tasks to be done

Partial order $\preceq$ is Precedence Constraint

Chain: sequence $a_1 \preceq a_2 \preceq \dots \preceq a_n$ s.t each item is comparable to the next item in the chain

Length of Time ~ Len(Longest chain)

ex Getting dressed



sample chain { undies ≤ pants ≤ belt ≤ coat

* Unit # 3 : Counting *

Chapter #9: Sums & Asymptotics

- **Sums** and **Products** arise regularly in analysis of algorithms, finance, physical, Probabilistic Problems

ex # of Nodes in complete Binary Tree

$$\sum_{i=0}^{\log(N)} 2^i \longrightarrow \boxed{2N-1}$$

closed form

Closed Form → Not making use of summation notation

The Perturbation Method

- useful to perturb the sum to get something simpler

ex

$$S = 1 + x + x^2 + \dots x^{n-1}$$

(perturb) $\times S = x + \dots x^n$

$$(S - Sx) = 1 - x^n$$

$$S = \frac{1-x^n}{1-x}$$

Variations in Geometric Sums

- Can obtain new summation by differentiating or integrate

$$\sum_{i=0}^{n-1} \frac{d}{dx}(x^i) = \sum_{i=0}^{n-1} i\,x^{i-1}$$

- If differentiate messes up powers of x, Multiply by x

Power Sums

Examine: $\sum_{i=1}^{n} i^2$

Guess (Grows as 3rd Degree Polynomial)

$$\sum_{i=1}^{n} i^2 = an^3 + bn^2 + cn + d$$

→ Try values & get unique linear solution

NOTE: ALWAYS Check Result by Induction

Approximating Sums

- Sometimes summations have no closed forms
  - → Can form upper/lower bound replacing Sum w/ Integral

$$S = \sum_{i=1}^{n} f(i)$$

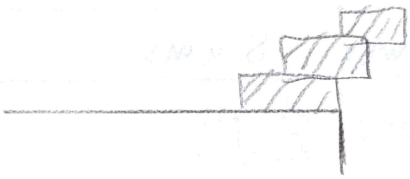
$$I = \int_{1}^{n} f(i)\,di$$

$$I + f(1) \leq S \leq I + f(n)$$

↳ sign inequality change for Decreasing function

Guided Example: Hanging over the edge

- Is there some number n blocks that'll extend past table's edge

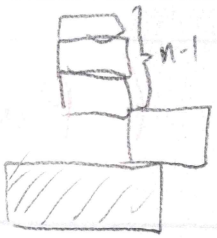


- Stability
  - → wont fall over
- Find Spread N possible w/ n blocks

Know: $S_1 = 1/2$

Cases of Configuration

Case 1 Right-most block in S is bottom Block



$$C_m = \frac{(n-1)1 + (1)\left(\frac{1}{2}\right)}{n} = 1 - \frac{1}{2n}$$

Case 2 Right most block in S on top

Recurrence form of Spread Soln

$$S_n = \sum_{i=1}^{n} \frac{1}{2i}$$

Harmonic Numbers

$$H_n ::= \sum_{i=1}^{n} \frac{1}{i}$$

- → no closed form
- → Also not Divergent as $n \to \infty$, $S \to \infty$

Double Summation

Try exchanging order of summation

ex

$$\sum_{K=1}^{n} H_K = \sum_{K=1}^{n} \sum_{j=1}^{K} \frac{1}{j}$$

Products

- Can transform

$$\prod_{i=1}^{n} \xrightarrow{\ln()} \sum_{i=1}^{n}$$

taking the log

Key Concept: Asymptotic Notation

→ used to characterize behavior of function as n grows large

Little Oh

- Functions f asymptotically smaller than g if

$$\lim_{x \to \infty} \frac{f(x)}{g(x)} = 0$$

BIG Oh

→ Gives upper bound on run-time of algorithm

$$\limsup_{x \to \infty} f(x)/g(x) < \infty$$

pinch

ex

$$2x = O(x)$$

pegging run-time to functional form complexity

Omega

Big → Inverse of Big Oh

Chapter #10: Recurrences

- A recurrence describes a sequence of numbers whose later terms are products of predecessors

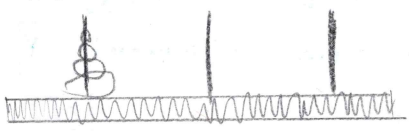
ex

$$T_1 = 1$$
$$T_n = T_{n-1} + 1$$

→ Reduces Big Problem to smaller problems w/ easy base cases

6.042[J] (Page 9)

Towers of Hanoi



- You labor to move all disks from one to other according to 2 Rules

① Only permitted action is moving top disk to another

② A large disk can never lie above smaller disk

A recursive solution

T_n → min(# of steps to solve n-disk problem)

3 steps

1. Move top n-1 disks to another post in T_{n-1} steps
2. Move largest disk (1 step)
3. same as 1)

$$T_n = 2T_{n-1} + 1$$

Solving Recurrence

→ Goal is to achieve closed form

Method 1 | Guess solution form

- Observe series values
- Observe Mechanics of Process

Plug 'n' Chug Method

→ spot pattern in sequence of expressions

① Identify Pattern

② Verify Pattern

③ write in terms of known recurrence base values

MERGE SORT

Input: List of n numbers

Procedure: first half | second half of list are sorted recursively

Q: Max # of possible comparisons used in sorting n items?

$$T_1 = 0$$

$$T_n = 2T_{n/2} + (n-1)$$

Solving

* Form not easy to guess

Linear Recurrences

$$f(n) = a_1 f(n-1) + a_2 f(n-2), + \ldots$$

add $f(n-1)$

Key Point: Solving Generalized Linear Recurrences

① Equation - Form

$$f(n) = \underbrace{\sum_{i=1}^{d} a_i f(n-i)}_{\text{homogenous Part}} + \underbrace{g(n)}_{\text{inhomogenous Part}}$$

Just know method exists

Chapter 11: Cardinality Rules

- Can count one thing by counting another
  - Determine size of set S

Bijection Rule

- If there exists bi-jection $f: A \to B$, then $|A| = |B|$

ex)

A = Ways to select dozen doughnuts of five varieties

B = 16-bit sequences w/ exactly four 1's

bi-jection!

→ Knowing size of one set, you know the other

Counting Sequences

Product Rule

$$|P_1 \times P_2 \times P_3| = |P_1| \cdot |P_2| \cdot |P_3|$$

Subsets of n-element list

- Bi-jection: Subsets of x to n-bit sequences

$\rightarrow 2^n$

subset: X_1 X_2 X_3

sequence: 00 1 0 1 00 1 00

Generalized Product Rule

- 3 prizes awarded into group of n people

$$|S| = n \cdot (n-1) \cdot (n-2)$$

The Division Rule

- If $f: A \rightarrow B$ is K-to-1

$$|A| = K \cdot |B|$$

Counting Subsets

- How many K-element subsets in n-element set

Notation $\binom{n}{k} ::=$ # of K element subsets in n-element list

$$\binom{n}{k} = \frac{n!}{k!(n-k)!}$$

Examples w/ Poker

① Hands w/ four of a kind
- Rank of four cards
- Rank of extra card
- Suit of extra card

Bijection: {sequence of two ranks & suit} → hands

② Full house

③ 2-pair → Don't forget 2-factor!

④ Hands with every suit
- composition of suits $\binom{4}{3}$

Inclusion – Exclusion

Union of Two Sets

$$|S_1 \cup S_2| = |S_1| + |S_2| - |S_1 \cap S_2|$$

ex|
How many permutations of $\{1, 2, \ldots 9\}$ contain 42, 04, or 60 consecutive

→ call this Pairs

P_{42}, P_{04}, P_{60} → where pair wise permutations occur

$P_{42,60}$, $P_{1,\ldots}$, $P_{1,2,3}$ } use inclusion/exclusion to solve

Combinatorial Proofs

① Define a set S

② Show that $|S| = n$ counting one way

③ Show $|S| = m$, another

⇒ prove $n = m$

The Pigeon hole Principle

- If $|X| > |Y|$, then for every total function $f: X \rightarrow Y$, there exist two elements in X mapped to same element Y

A B

Key to identify 3-things

① Set A ② set B ③ Mapping Function (f)

Guided Example: Magic Trick

- Audience selects 5 cards
- Assistant holds up 4 cards
- Magician guesses 5th one

tacit signal of communication { Order of Card Reveal

X: All sets of 5 cards

Y: Sequences of four distinct cards

{A, C, D, E, F} ⟋ {A, C, D, E}

Bi-Partite Matching Scheme

- Every distinct purpose of action adds permutation complexity

Chapter 12: Generating Functions

△ Transforms projects around sequences into problems about functions

Definition

- An ordinary generating function for sequence $\langle g_0, g_1, g_2, g_3 \rangle$ is power series

$$G(x) = g_0 + g_1 x + \ldots$$

i^{th} term in sequence = coefficient of x^i in generating function

Notation

$$\langle g_0, g_1, \ldots \rangle \longleftrightarrow g_0 + g_1 x + \ldots$$

ex| Closed Sum of infinite geometric series

$\langle 1,1,1 \rangle \longleftrightarrow 1+x+x^2 = \frac{1}{1-x}$

$\langle 1,-1,1 \rangle \longleftrightarrow 1-x+x^2 = \frac{1}{1+x}$

$\langle 1, a, a^2, a^3 \rangle \longleftrightarrow 1+ax \quad \frac{1}{1-ax}$

$\ldots \longleftrightarrow \ldots$

Operations on Generating Functions

Carry out manipulations on sequences by doing mathematical operations on generating functions

Scaling

$$\langle cf_0, cf_1, cf_2 \ldots \rangle \longleftrightarrow c \cdot F(x)$$

Addition

$$\langle f_0, f_1, f_2, \ldots \rangle \longleftrightarrow F(x)$$
$$+ \langle g_0, g_1, g_2, \ldots \rangle \longleftrightarrow G(x)$$
$$\langle f_0+g_0, f_1+g_1, f_2+g_2 \rangle \longleftrightarrow F(x)+G(x)$$

Right-Shift

$$\langle f_0, f_1, f_2 \rangle = F(x)$$

$$\langle \underbrace{0, \ldots, 0}_{K}, f_0, \ldots f_n \rangle = x^K \cdot F(x)$$

Differentiation

$$\langle f_1, 2f_2, 3f_3 \rangle \longleftrightarrow F'(x)$$

ex|

$$\frac{d}{d}\langle 1,1,1,1,1 \rangle \to \langle 1,2,3,4 \rangle$$
$$\langle 1,2,3,4 \rangle \to \langle 0,1,2,3,4 \rangle$$

* can continue to get square series $\langle 1,4,9, \ldots \rangle$

Products

$$\langle a_0, a_1, a_2 \rangle \to A(x) \quad \boxtimes \quad \langle b_0, b_1, b_2 \rangle \to B(x)$$

$$\Downarrow$$

$$\langle c_0, c_1, c_2 \rangle = A(x) \cdot B(x)$$

where, $c_n := a_0 b_n + a_1 b_{n-1} \cdots + a_n b_0$

signal processing { convolution

Evaluating Sums

Suppose: $B(x) = \frac{1}{1-x}$

then, $b_i = 1$, n^{th} co-ef of $A(x) \cdot B(x) = \sum_{a_i=0}^{n} a_i$

$$s_n = \sum_{i=0}^{n} a_i$$

Summation Rule

$$\langle a_0, a_1, a_2 \ldots \rangle \longleftrightarrow A(x)$$

$$\langle s_0, s_1, s_2, \ldots \rangle \longleftrightarrow \frac{A(x)}{1-x}$$

where,

$$S_n = \sum_{i=0}^{n} a_i \quad \text{for } n \geq 0$$

Extracting Co-efficients

- given $\langle f_0, f_1, f_2 \rangle \to F(x)$ simple

Key| To go from Generating Function to coefficient sequence need:

Taylor Series

Let $F(x)$ be generating function of sequence $\langle f_0, f_1, f_2 \ldots \rangle$

$$f_0 = F(0)$$

$$f_n = \frac{F^{(n)}(0)}{n!}$$

→ allows for expansion of well-known functions

ex)

$$F(x) = \frac{x+x^2}{(1-x)^4} = \frac{x}{(1-x)^4} + \frac{x^2}{(1-x)^4}$$

Key: x^n in $F(x)$ is sum of x^{n-1} in $\frac{1}{(1-x)^4}$

& x^{n-2} in $\frac{1}{(1-x)^4}$

Partial Fractions

- underlies trick we employed above

(terms of form) $\frac{cx^a}{(1-dx)^b}$

n^{th} derivative of $\frac{1}{(1-dx)^b}$

$$\frac{(n+b-1)\,d^n}{(b-1)!\,(1-dx)^{b+n}}$$

→ Then Following Partial Fraction Decomposition, add appropriate terms

Solving Linear Recurrences

- Just know it can be done for general Linear Recurrences

Counting w/ Generating Functions

- Useful for counting Problems (Choosing items from a set)

→ coefficient of x_n is way to select n things

Choosing distinct items

$\langle \binom{k}{0}, \binom{k}{1}, \ldots \binom{k}{k}, 0, 0 \rangle$

co-eff of x^n is number of ways to select n from K

Chapter #13: Infinite Sets

- Just know power series is LoL always larger

Chapter #14: Events & Probability Space

Four-Step Method

Step #1: Find Sample space

- Identify All possible outcomes
- Find Key qualities of Decision space prior to event

Step #2: Define event of interest

Step #3: Determine Outcome Probabilities

Step 4 → compute event probabilities

Set Theory & Probability

- Set = Sample Space
- ~~Subset~~ = Event

Terminology

- A countable sample space $\int$ is a non-empty countable set, whose elements we $\int$ are outcomes
- subset of $\int$ is an event

$$Pr(E) ::= \sum_{w \in E} Pr(w)$$

Sum Rule (if events Disjoint)

$$Pr[E \cup F] = Pr[E] + Pr[F]$$

Uniform Probability Space

$$Pr(E) = \frac{|E|}{|S|}$$

(counting # of outcomes akin to probability)

Chapter #15: Conditional Probability

Notation: $Pr[A|B]$

A Posteriori Probabilities

- conditional Probability $P[B|A]$ is A posteriori if

B before A in time

⇓ Gives rise to Bayes Rule

Key Point: Equivalency between A posteriori & regular event spaces

A = Won the series

B = won first match

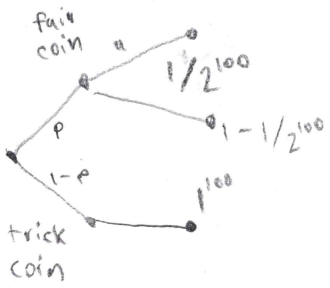
$$P[B|A] = \frac{Pr(A)\,Pr(A|B)}{Pr(B)}$$

6.042[J] (Page 1)

ex) Coin Problem

- fair or trick coin w/ heads on both sides

⟹ can we derive intuition as to whether coin was fair if we flip heads 100 times



A → handed fair coin

B → flipped all heads

Conditional Identities

$$Pr(A) = Pr[A|E] \cdot P[E] + Pr[A|\bar{E}] \cdot Pr[\bar{E}]$$

Chapter #16: Independence

- Events A & B independent if
  - $Pr(B) = 0$ or
  - $Pr(A|B) = Pr(A)$

Note: Disjoint ≠ Independent

Independence is an Assumption

Mutual Independence

$$Pr\left[\bigcap_{j \in S} E_j\right] = \prod_{j \in S} Pr(E_j)$$

Pair-wise Independence

- a set of $A_1, A_2 \ldots$ of events is k-way independent iff every set of k of these events is mutually independent

The Birthday Paradox

- In 100-person class what is probability a birthday is shared?

Assumptions

- All birthdays equally likely
- Birthdays are Mutually independent

Application to: Hashing

- set of m numbers you'd like to assign from [1, N]

Chapter #17: Random Variables and Distributions

Random Variables

- Random variable R on a probability space is a total function whose Domain is sample space

Mapping

{event} ⟶ Real #

- Indicator Random Variables
  - → Maps every outcome to [0,1]

- RV's partition sample spaces into subsets (like events)

→ same rules for events apply to RV

Distribution Functions

- Random Variables on Different Probabily spaces can share Probability Density Functions

$$PDF_R(x) = \begin{cases} Pr[R=x] & \text{if } x \in \text{range}(R) \\ 0, & \text{else} \end{cases}$$

$$CDF_R(x) \quad Pr[R \leq x]$$

Bernoulli Distributions

→ Distribution of an indicator RV

$$f_p(0) = p$$

$$f_p(1) = 1 - p$$

Bi-nomial Distribution

$$f_n(k) = \binom{n}{k} 2^{-n}$$

In General

$$f_n(k) = \binom{n}{k} p^i \cdot (1-p)^{n-i}$$

Chapter #19: Deviations

Variance

$$\text{Var}[R] ::= Ex[R - Ex[R]]^2$$

- we square to avoid sign cancellation

Markov's Theorem

$$Pr[R \geq x] \leq \frac{Ex[R]}{x}$$

Proof

$$Ex[R] = \sum_{y \in \text{range}(R)} y \cdot Pr[R = y]$$

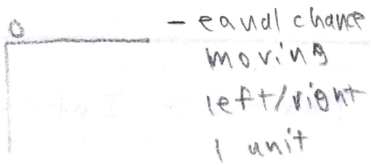
$$\geq \sum_{\substack{y \geq x \\ y \in \text{range}(R)}} x \cdot Pr[R = y]$$

$$= x \, Pr[R \geq x]$$

** See Last Page for in depth Statistics Notes on Markov Theorem & Chebyshev

Chapter #20: Unbiased Random walks

Inspiration

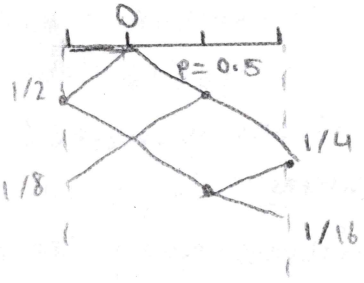


Random walk → some value moves up or down over time

Terminology

boundary condition → if value hits walk ends

Example



- Unbiased Random walk with absorbing barriers at [0] & [3]

→ Takes form of Recurrence (18.03 will provide for info on how to solve such systems)

Markov's Theorem

Derivation

$$\Pr[R \geq x] \leq \frac{Ex[R]}{x}$$

1) $Ex[R] = \sum_{y \in Range(R)} y \cdot \Pr[R=y]$

2) " " $\geq \sum_{\substack{y \geq x \\ y \in range(R)}} x \cdot \Pr[R=y]$

$\geq x \cdot \sum_{\substack{y \geq x \\ y \in Range(R)}} \Pr[R=y]$

$$Ex[R] \geq x \cdot \Pr[R \geq x]$$

Caveat: only for non-negative Ramdoom variable

→ gives guidance on bounded Probability of ability of RV to attain value

CASE: Bounded Variables

- Suppose $Ex[R] = 150$
  $\Rightarrow \Pr(R \geq 200)$

[8] R is no less than 100

Trans formation: $T ::= R - 100$

$\Pr[R > 200] = \Pr[T > 100] \leq \frac{50}{100}$

* Better Markow bounds found by transforming variable

Case: Deviations Below Mean

Transformation: $S ::= \mu - R$

Chebyshev's Theorem

Key Idea: Applying Markov's Thm to powers of [R]

$$\Pr[|R - Ex[R]| \geq x] \leq \frac{Var[R]}{x^2}$$

special substitution:

$$x = c\sigma_R$$

c = constant

σ_R = std Dev

Bounds for Sums of Random Variables

Motivating Example

- 24 K posts received every 10 minutes
- Processing Avg = [1/4]
- → assigned across [m] machines
- * Random solutions can come in handy where deterministic methods fail

Murphy's Law

- Let $A_1, \ldots A_n$ be mutually independent events

$T = T_1 + T_2 + T$

where [T] is indicator variable of [A]

$$\Pr[T=0] \leq e^{-Ex[T]}$$

The Chernoff Bound

* High Parallels to Stat Mech

Def: The sum of lots of little independent RV's unlikely to significantly exceed mean of sum

Theorem

→ Let $T_1, \ldots, T_n$ be mutually independent RV s.t $0 \leq T_i \leq 1$

Let $T = \sum_{i=1}^{n} T_i$

- Sum takes on value on $[0,1]$
- No assumed Distribution form
- No dependency on [n]

Derivation

$$\Pr[T \geq c\,Ex[T]] = \Pr[c^T \geq c^{c\,Ex[T]}]$$

$$\left(\begin{matrix}\text{By}\\ \text{Markov}\end{matrix}\right) \leq \frac{Ex[c^T]}{c^{c\,Ex[T]}}$$

$$\leq \frac{e^{(c-1)\,Ex[T]}}{c^{c\,Ex[T]}}$$