

Discrete Mathematics & Mathematical Reasoning

Course Overview

Colin Stirling

Informatics

Slides based on ones by Myrto Arapinis

Teaching staff

Lecturers:

- Colin Stirling, first half of course
- Kousha Etessami, second half of course

Course TA:

- Daniel Franzen & Weili Fu

Course Secretary (ITO):

- Kendall Reid (kr@inf.ed.ac.uk)

Course web page

<http://www.inf.ed.ac.uk/teaching/courses/dmmr/>

Contains important information

- Lecture slides
- Tutorial sheet exercises
- Course organization
- ...

Tutorials

- You should receive email from the ITO informing you of preliminary allocation of tutorial groups

Tutorials

- You should receive email from the ITO informing you of preliminary allocation of tutorial groups
- See link on course web page for current assignment of tutorial groups

Tutorials

- You should receive email from the ITO informing you of preliminary allocation of tutorial groups
- See link on course web page for current assignment of tutorial groups
- If you can't make the time of your allocated group, please email Kendall suggesting some groups you can manage

Tutorials

- You should receive email from the ITO informing you of preliminary allocation of tutorial groups
- See link on course web page for current assignment of tutorial groups
- If you can't make the time of your allocated group, please email Kendall suggesting some groups you can manage
- If you change tutor groups for any reason, you must let Kendall and the ITO know **(because your marked coursework is returned at the tutorial groups)**

Tutorials

- You should receive email from the ITO informing you of preliminary allocation of tutorial groups
- See link on course web page for current assignment of tutorial groups
- If you can't make the time of your allocated group, please email Kendall suggesting some groups you can manage
- If you change tutor groups for any reason, you must let Kendall and the ITO know **(because your marked coursework is returned at the tutorial groups)**
- Tutorial attendance is mandatory. If you miss two tutorials in a row, your PT will be notified

Tutorials and (marked) exercises

- Weekly exercise sheets, available by Friday 2pm on the course web page

Tutorials and (marked) exercises

- Weekly exercise sheets, available by Friday 2pm on the course web page
- **The last question on every sheet will be graded.** The coursework grade contributes 15% to the total course grade, and every one of the 9 exercise sheets counts 1/9th of the coursework grade

Tutorials and (marked) exercises

- Weekly exercise sheets, available by Friday 2pm on the course web page
- The last question on every sheet will be graded. The coursework grade contributes 15% to the total course grade, and every one of the 9 exercise sheets counts 1/9th of the coursework grade
- Starting in week 2, deadline for submission of each tutorial sheet is Wednesday at 4:00pm at the ITO (they also have a collection box)

Tutorials and (marked) exercises

- Weekly exercise sheets, available by Friday 2pm on the course web page
- The last question on every sheet will be graded. The coursework grade contributes 15% to the total course grade, and every one of the 9 exercise sheets counts 1/9th of the coursework grade
- Starting in week 2, deadline for submission of each tutorial sheet is Wednesday at 4:00pm at the ITO (they also have a collection box)
- Solutions will be discussed in tutorials the following week. Graded sheets are returned in tutorials (or collected later from the ITO)

Tutorials and (marked) exercises

- Weekly exercise sheets, available by Friday 2pm on the course web page
- The last question on every sheet will be graded. The coursework grade contributes 15% to the total course grade, and every one of the 9 exercise sheets counts 1/9th of the coursework grade
- Starting in week 2, deadline for submission of each tutorial sheet is Wednesday at 4:00pm at the ITO (they also have a collection box)
- Solutions will be discussed in tutorials the following week. Graded sheets are returned in tutorials (or collected later from the ITO)
- Exception: no tutorial on week 1

Textbook

- Kenneth Rosen, **Discrete Mathematics and its Applications**, 7th Edition, (Global Edition) McGraw-Hill, 2012
- Available at Blackwells
- For additional material see the course webpage

Grading

- Written Examination: 85%
- Assessed Assignments: 15%. Each one of the 9 exercise sheets counts equally, *i.e.* 1/9th

Important themes

- mathematical reasoning
- combinatorial analysis
- discrete structures
- algorithmic thinking
- applications and modelling

Foundations: proof

- Rudimentary predicate (first-order) logic: existential and universal quantification, basic algebraic laws of quantified logic (duality of existential and universal quantification)
- The structure of a well-reasoned mathematical proof; Proof strategies: proofs by contradiction, proof by cases; examples of incorrect proofs (to build intuition about correct mathematical reasoning)

Foundations: sets, functions and relations

- Sets (naive): operations on sets: union, intersection, set difference, the powerset operation, examples of finite and infinite sets (the natural numbers). Ordered pairs, n-tuples, and Cartesian products of sets
- Relations: (unary, binary, and n-ary) properties of binary relations (symmetry, reflexivity, transitivity).
- Functions: injective, surjective, and bijective functions, inverse functions, composition of functions
- Rudimentary counting: size of the Cartesian product of two finite sets, number of subsets of a finite set, (number of n-bit sequences), number of functions from one finite set to another

Basic number theory and cryptography

- Integers and elementary number theory (divisibility, GCDs and the Euclidean algorithm, prime decomposition and the fundamental theorem of arithmetic)
- Modular arithmetic (congruences, Fermat's little theorem, the Chinese remainder theorem)
- Applications: public-key cryptography

Basic algorithms

- Concept and basic properties of an algorithm
- Basics of growth of function, and complexity of algorithms: comparison of growth rate of some common functions

Induction and recursion

- Principle of mathematical induction (for positive integers)
- Examples of proofs by (weak and strong) induction
- Recursive definitions and Structural induction

Counting

- Basics of counting
- Pigeon-hole principle
- Permutations and combinations
- Binomial coefficients, binomial theorem, and basic identities on binomial coefficients
- Generalizations of permutations and combinations (e.g., combinations with repetition/replacement)
- Stirling's approximation of the factorial function

Graphs

- Directed and undirected graph: definitions and examples in Informatics
- Adjacency matrix representation
- Terminology: degree (indegree, outdegree), and special graphs: bipartite, complete, acyclic, ...
- Isomorphism of graphs; subgraphs
- Paths, cycles, and (strong) connectivity
- Euler paths/circuits, Hamiltonian paths (brief)
- Weighted graphs, and shortest paths (Dijkstra's algorithm)
- Bipartite matching: Hall's marriage theorem

Trees

- Rooted and unrooted trees
- Ordered and unordered trees
- (Complete) binary (k -ary) tree
- Subtrees
- Examples in Informatics
- Spanning trees (Kruskal's algorithm, Prim's algorithm.)

Discrete probability

- Discrete (finite or countable) probability spaces
- Events
- Basic axioms of discrete probability
- Independence and conditional probability
- Bayes' theorem
- Random variables
- Expectation; linearity of expectation
- Basic examples of discrete probability distributions birthday paradox, and other subtle examples in probability

“Proof” that $1 = 2$

Step

1. $a = b$

Reason

Premise

“Proof” that $1 = 2$

Step

1. $a = b$
2. $a^2 = ab$

Reason

Premise

Multiply both sides by a

“Proof” that $1 = 2$

Step

1. $a = b$
2. $a^2 = ab$
3. $a^2 - b^2 = ab - b^2$

Reason

- Premise
Multiply both sides by a
Subtract b^2 from both sides

“Proof” that $1 = 2$

Step

1. $a = b$
2. $a^2 = ab$
3. $a^2 - b^2 = ab - b^2$
4. $(a - b)(a + b) = b(a - b)$

Reason

- Premise
Multiply both sides by a
Subtract b^2 from both sides
Algebra

“Proof” that $1 = 2$

Step

1. $a = b$
2. $a^2 = ab$
3. $a^2 - b^2 = ab - b^2$
4. $(a - b)(a + b) = b(a - b)$
5. $a + b = b$

Reason

- Premise
Multiply both sides by a
Subtract b^2 from both sides
Algebra
Divide both sides by $a - b$

“Proof” that $1 = 2$

Step

1. $a = b$
2. $a^2 = ab$
3. $a^2 - b^2 = ab - b^2$
4. $(a - b)(a + b) = b(a - b)$
5. $a + b = b$
6. $2b = b$

Reason

- Premise
- Multiply both sides by a
- Subtract b^2 from both sides
- Algebra
- Divide both sides by $a - b$
- Replace a by b because $a = b$

“Proof” that $1 = 2$

Step

1. $a = b$
2. $a^2 = ab$
3. $a^2 - b^2 = ab - b^2$
4. $(a - b)(a + b) = b(a - b)$
5. $a + b = b$
6. $2b = b$
7. $2 = 1$

Reason

- Premise
- Multiply both sides by a
- Subtract b^2 from both sides
- Algebra
- Divide both sides by $a - b$
- Replace a by b because $a = b$
- Divide both sides by b

“Proof” that $1 = 2$

Step

1. $a = b$
2. $a^2 = ab$
3. $a^2 - b^2 = ab - b^2$
4. $(a - b)(a + b) = b(a - b)$
5. $a + b = b$
6. $2b = b$
7. $2 = 1$

Reason

- Premise
- Multiply both sides by a
- Subtract b^2 from both sides
- Algebra
- Divide both sides by $a - b$
- Replace a by b because $a = b$
- Divide both sides by b

Step 5. $a - b = 0$ by the premise and division by 0 is undefined!

Discrete Mathematics & Mathematical Reasoning

Predicates, Quantifiers and Proof Techniques

Colin Stirling

Informatics

Some slides based on ones by Myrto Arapinis

Recall propositional logic from last year (in Inf1CL)

Propositions can be constructed from other propositions using logical connectives

Recall propositional logic from last year (in Inf1CL)

Propositions can be constructed from other propositions using logical connectives

- Negation: \neg
- Conjunction: \wedge
- Disjunction: \vee
- Implication: \rightarrow
- Biconditional: \leftrightarrow

Recall propositional logic from last year (in Inf1CL)

Propositions can be constructed from other propositions using logical connectives

- Negation: \neg
- Conjunction: \wedge
- Disjunction: \vee
- Implication: \rightarrow
- Biconditional: \leftrightarrow

The truth of a proposition is defined by the truth values of its elementary propositions and the meaning of connectives

Recall propositional logic from last year (in Inf1CL)

Propositions can be constructed from other propositions using logical connectives

- Negation: \neg
- Conjunction: \wedge
- Disjunction: \vee
- Implication: \rightarrow
- Biconditional: \leftrightarrow

The truth of a proposition is defined by the truth values of its elementary propositions and the meaning of connectives

The meaning of logical connectives can be defined using truth tables

Propositional logic is not enough

Propositional logic is not enough

In propositional logic, from

- All men are mortal
- Socrates is a man

we cannot derive

- Socrates is mortal

Propositional logic is not enough

In propositional logic, from

- All men are mortal
- Socrates is a man

we cannot derive

- Socrates is mortal

We need a language to talk about objects, their properties and their relations

Predicate logic

Extends propositional logic by the new features

- Variables: x, y, z, \dots
- Predicates: $P(x), Q(x), R(x, y), M(x, y, z), \dots$
- Quantifiers: \forall, \exists

Predicate logic

Extends propositional logic by the new features

- Variables: x, y, z, \dots
- Predicates: $P(x), Q(x), R(x, y), M(x, y, z), \dots$
- Quantifiers: \forall, \exists

Predicates are a generalisation of propositions

- Can contain variables $M(x, y, z)$
- Variables stand for (and can be replaced by) elements from their domain
- The truth value of a predicate depends on the values of its variables

Examples

$P(x)$ is “ $x > 5$ ” and x ranges over \mathbb{Z} (integers)

- $P(8)$ is true
- $P(5)$ is false

Examples

$P(x)$ is “ $x > 5$ ” and x ranges over \mathbb{Z} (integers)

- $P(8)$ is true
- $P(5)$ is false

$Q(x)$ is “ x is irrational” and x ranges over \mathbb{R} (real numbers)

- $Q(\sqrt{2})$ is true
- $Q(\sqrt{4})$ is false

Examples

$P(x)$ is “ $x > 5$ ” and x ranges over \mathbb{Z} (integers)

- $P(8)$ is true
- $P(5)$ is false

$Q(x)$ is “ x is irrational” and x ranges over \mathbb{R} (real numbers)

- $Q(\sqrt{2})$ is true
- $Q(\sqrt{4})$ is false

$R(x, y)$ is “ x divides y ” and x, y range over \mathbb{Z}^+ (positive integers)

- $R(3, 9)$ is true
- $R(2, 9)$ is false

Quantifiers

- Universal quantifier, “For all”: \forall

$\forall x P(x)$ asserts that $P(x)$ is true for every x in the assumed domain

Quantifiers

- Universal quantifier, “For all”: \forall

$\forall x P(x)$ asserts that $P(x)$ is true for every x in the assumed domain

- Existential quantifier, “There exists”: \exists

$\exists x P(x)$ asserts that $P(x)$ is true for some x in the assumed domain

Quantifiers

- Universal quantifier, “For all”: \forall

$\forall x P(x)$ asserts that $P(x)$ is true for every x in the assumed domain

- Existential quantifier, “There exists”: \exists

$\exists x P(x)$ asserts that $P(x)$ is true for some x in the assumed domain

- The quantifiers are said to bind the variable x in these expressions. Variables in the scope of some quantifier are called bound variables. All other variables in the expression are called free variables

Quantifiers

- Universal quantifier, “For all”: \forall

$\forall x P(x)$ asserts that $P(x)$ is true for every x in the assumed domain

- Existential quantifier, “There exists”: \exists

$\exists x P(x)$ asserts that $P(x)$ is true for some x in the assumed domain

- The quantifiers are said to bind the variable x in these expressions. Variables in the scope of some quantifier are called bound variables. All other variables in the expression are called free variables
- A formula that does not contain any free variables is a proposition and has a truth value

Example: If n is an odd integer then n^2 is odd

- First, notice the quantifier is implicit

Example: If n is an odd integer then n^2 is odd

- First, notice the quantifier is implicit
- Let $P(n)$ mean n is odd where n is an integer (in \mathbb{Z})

Example: If n is an odd integer then n^2 is odd

- First, notice the quantifier is implicit
- Let $P(n)$ mean n is odd where n is an integer (in \mathbb{Z})
- So is: $\forall x$ (if $P(x)$ then $P(x^2)$)

Example: If n is an odd integer then n^2 is odd

- First, notice the quantifier is implicit
- Let $P(n)$ mean n is odd where n is an integer (in \mathbb{Z})
- So is: $\forall x$ (if $P(x)$ then $P(x^2)$)
- $\forall x(P(x) \rightarrow P(x^2))$

Direct proof of $\forall x (P(x) \rightarrow Q(x))$

- Assume c is an arbitrary element of the domain

Direct proof of $\forall x (P(x) \rightarrow Q(x))$

- Assume c is an arbitrary element of the domain
- Prove that $P(c) \rightarrow Q(c)$

Direct proof of $\forall x (P(x) \rightarrow Q(x))$

- Assume c is an arbitrary element of the domain
- Prove that $P(c) \rightarrow Q(c)$
- That is, assume $P(c)$ then show $Q(c)$

Direct proof of $\forall x (P(x) \rightarrow Q(x))$

- Assume c is an arbitrary element of the domain
- Prove that $P(c) \rightarrow Q(c)$
- That is, assume $P(c)$ then show $Q(c)$
- Use the definition/properties of $P(c)$

Example: If n is an odd integer then n^2 is odd

- $\forall x (P(x) \rightarrow P(x^2))$ where $P(n)$ is n is odd

Example: If n is an odd integer then n^2 is odd

- $\forall x (P(x) \rightarrow P(x^2))$ where $P(n)$ is n is odd
- Assume n is arbitrary odd integer; what does that mean?

Example: If n is an odd integer then n^2 is odd

- $\forall x (P(x) \rightarrow P(x^2))$ where $P(n)$ is n is odd
- Assume n is arbitrary odd integer; what does that mean?
- that for some k , $n = 2k + 1$

Example: If n is an odd integer then n^2 is odd

- $\forall x (P(x) \rightarrow P(x^2))$ where $P(n)$ is n is odd
- Assume n is arbitrary odd integer; what does that mean?
- that for some k , $n = 2k + 1$
- Show n^2 is odd

Example: If n is an odd integer then n^2 is odd

- $\forall x (P(x) \rightarrow P(x^2))$ where $P(n)$ is n is odd
- Assume n is arbitrary odd integer; what does that mean?
- that for some k , $n = 2k + 1$
- Show n^2 is odd
- $n^2 = (2k + 1)^2$

Example: If n is an odd integer then n^2 is odd

- $\forall x (P(x) \rightarrow P(x^2))$ where $P(n)$ is n is odd
- Assume n is arbitrary odd integer; what does that mean?
- that for some k , $n = 2k + 1$
- Show n^2 is odd
- $n^2 = (2k + 1)^2$
- So, $n^2 = 4k^2 + 4k + 1 = 2(2k^2 + 2k) + 1$

Example: If n is an odd integer then n^2 is odd

- $\forall x (P(x) \rightarrow P(x^2))$ where $P(n)$ is n is odd
- Assume n is arbitrary odd integer; what does that mean?
- that for some k , $n = 2k + 1$
- Show n^2 is odd
- $n^2 = (2k + 1)^2$
- So, $n^2 = 4k^2 + 4k + 1 = 2(2k^2 + 2k) + 1$
- n^2 has the form for some m , $n^2 = 2m + 1$; so n^2 is odd

Any odd integer is the difference of two squares

Nested quantifiers

- Every real number has an inverse w.r.t addition
The domain is \mathbb{R}

$$\forall x \exists y (x + y = 0)$$

- Every real number except zero has an inverse w.r.t multiplication
The domain is \mathbb{R}

$$\forall x (x \neq 0 \rightarrow \exists y (x \times y = 1))$$

Proving $\forall x (P(x) \rightarrow Q(x))$ by contraposition

- Uses equivalence of $(p \rightarrow q)$ and $(\neg q \rightarrow \neg p)$

Proving $\forall x (P(x) \rightarrow Q(x))$ by contraposition

- Uses equivalence of $(p \rightarrow q)$ and $(\neg q \rightarrow \neg p)$
- Assume c is an arbitrary element of the domain

Proving $\forall x (P(x) \rightarrow Q(x))$ by contraposition

- Uses equivalence of $(p \rightarrow q)$ and $(\neg q \rightarrow \neg p)$
- Assume c is an arbitrary element of the domain
- Prove that $\neg Q(c) \rightarrow \neg P(c)$

Proving $\forall x (P(x) \rightarrow Q(x))$ by contraposition

- Uses equivalence of $(p \rightarrow q)$ and $(\neg q \rightarrow \neg p)$
- Assume c is an arbitrary element of the domain
- Prove that $\neg Q(c) \rightarrow \neg P(c)$
- That is, assume $\neg Q(c)$ then show $\neg P(c)$

Proving $\forall x (P(x) \rightarrow Q(x))$ by contraposition

- Uses equivalence of $(p \rightarrow q)$ and $(\neg q \rightarrow \neg p)$
- Assume c is an arbitrary element of the domain
- Prove that $\neg Q(c) \rightarrow \neg P(c)$
- That is, assume $\neg Q(c)$ then show $\neg P(c)$
- Use the definition/properties of $\neg Q(c)$

if $x + y$ is even, then x and y have the same parity

if $x + y$ is even, then x and y have the same parity

Proof Let $n, m \in \mathbb{Z}$ be arbitrary. We will prove that if n and m do not have the same parity then $n + m$ is odd. Without loss of generality we assume that n is odd and m is even, that is $n = 2k + 1$ for some $k \in \mathbb{Z}$, and $m = 2\ell$ for some $\ell \in \mathbb{Z}$. But then

$n + m = 2k + 1 + 2\ell = 2(k + \ell) + 1$. And thus $n + m$ is odd. Now by equivalence of a statement with its contrapositive derive that if $n + m$ is even, then n and m have the same parity.

If $n = ab$ where a, b are positive integers, then $a \leq \sqrt{n}$
or $b \leq \sqrt{n}$

Proof by contradiction

- Want to prove that P is true

Proof by contradiction

- Want to prove that P is true
- Assume $\neg P$

Proof by contradiction

- Want to prove that P is true
- Assume $\neg P$
- Derive both R and $\neg R$ (a contradiction equivalent to False)

Proof by contradiction

- Want to prove that P is true
- Assume $\neg P$
- Derive both R and $\neg R$ (a contradiction equivalent to False)
- Therefore, $\neg\neg P$ which is equivalent to P

$\sqrt{2}$ is irrational

$\sqrt{2}$ is irrational

Proof Assume towards a contradiction that $\sqrt{2}$ is rational, that is there are integers a and b with no common factor other than 1, such that $\sqrt{2} = a/b$. In that case $2 = a^2/b^2$. Multiplying both sides by b^2 , we have $a^2 = 2b^2$. Since b is an integer, so is b^2 , and thus a^2 is even. As we saw previously this implies that a is even, that is there is an integer c such that $a = 2c$. Hence $2b^2 = 4c^2$, hence $b^2 = 2c^2$. Now, since c is an integer, so is c^2 , and thus b^2 is even. Again, we can conclude that b is even. Thus a and b have a common factor 2, contradicting the assertion that a and b have no common factor other than 1. This shows that the original assumption that $\sqrt{2}$ is rational is false, and that $\sqrt{2}$ must be irrational.

There are infinitely many primes

There are infinitely many primes

Lemma Every natural number greater than one is either prime or it has a prime divisor

There are infinitely many primes

Lemma Every natural number greater than one is either prime or it has a prime divisor

Proof Suppose towards a contradiction that there are only finitely many primes $p_1, p_2, p_3, \dots, p_k$. Consider the number $q = p_1 p_2 p_3 \dots p_k + 1$, the product of all the primes plus one. By hypothesis q cannot be prime because it is strictly larger than all the primes. Thus, by the lemma, it has a prime divisor, p . Because $p_1, p_2, p_3, \dots, p_k$ are all the primes, p must be equal to one of them, so p is a divisor of their product. So we have that p divides $p_1 p_2 p_3 \dots p_k$, and p divides q , but that means p divides their difference, which is 1. Therefore $p \leq 1$. Contradiction. Therefore there are infinitely many primes.

Proof by cases

- To prove a conditional statement of the form

$$(p_1 \vee \cdots \vee p_k) \rightarrow q$$

- Use the tautology

$$((p_1 \vee \cdots \vee p_k) \rightarrow q) \leftrightarrow ((p_1 \rightarrow q) \wedge \cdots \wedge (p_k \rightarrow q))$$

- Each of the implications $p_i \rightarrow q$ is a case

If n is an integer then $n^2 \geq n$

Constructive proof of $\exists x P(x)$

Constructive proof of $\exists x P(x)$

- Exhibit an actual witness w from the domain such that $P(w)$ is true

Constructive proof of $\exists x P(x)$

- Exhibit an actual witness w from the domain such that $P(w)$ is true
- Therefore, $\exists x P(x)$

There exists a positive integer that can be written as the sum of cubes of positive integers in two different ways

There exists a positive integer that can be written as the sum of cubes of positive integers in two different ways

- 1729 is such a number because

There exists a positive integer that can be written as the sum of cubes of positive integers in two different ways

- 1729 is such a number because
- $10^3 + 9^3 = 1729 = 12^3 + 1^3$

Nonconstructive proof of $\exists x P(x)$

Nonconstructive proof of $\exists x P(x)$

- Show that there must be a value v such that $P(v)$ is true

Nonconstructive proof of $\exists x P(x)$

- Show that there must be a value v such that $P(v)$ is true
- but we don't know what this value v is

There exist irrational numbers x and y such that x^y is rational

There exist irrational numbers x and y such that x^y is rational

Proof We need only prove the existence of at least one example.

Consider the case $x = \sqrt{2}$ and $y = \sqrt{2}$. We distinguish two cases:

Case $\sqrt{2}^{\sqrt{2}}$ is rational. In that case we have shown that for the irrational numbers $x = y = \sqrt{2}$, we have that x^y is rational

Case $\sqrt{2}^{\sqrt{2}}$ is irrational. In that case consider $x = \sqrt{2}^{\sqrt{2}}$ and $y = \sqrt{2}$. We then have that

$$x^y = (\sqrt{2}^{\sqrt{2}})^{\sqrt{2}} = \sqrt{2}^{\sqrt{2}\sqrt{2}} = \sqrt{2}^2 = 2$$

But since 2 is rational, we have shown that for $x = \sqrt{2}^{\sqrt{2}}$ and $y = \sqrt{2}$, we have that x^y is rational

We have thus shown that in any case there exist some irrational numbers x and y such that x^y is rational

Disproving $\forall x P(x)$ with a counter-example

- $\neg\forall x P(x)$ is equivalent to $\exists x \neg P(x)$

Disproving $\forall x P(x)$ with a counter-example

- $\neg\forall x P(x)$ is equivalent to $\exists x \neg P(x)$
- To establish that $\neg\forall x P(x)$ is true find a w such that $P(w)$ is false

Disproving $\forall x P(x)$ with a counter-example

- $\neg\forall x P(x)$ is equivalent to $\exists x \neg P(x)$
- To establish that $\neg\forall x P(x)$ is true find a w such that $P(w)$ is false
- So, w is a **counterexample** to the assertion $\forall x P(x)$

Every positive integer is the sum of the squares of 3 integers

Every positive integer is the sum of the squares of 3 integers

The integer 7 is a counterexample. So the claim is false

Discrete Mathematics & Mathematical Reasoning

Basic Structures: Sets, Functions and Relations

Colin Stirling

Informatics

Some slides based on ones by Myrto Arapinis

Some important sets

$\mathbb{B} = \{\text{true}, \text{false}\}$ Boolean values

$\mathbb{N} = \{0, 1, 2, 3, \dots\}$ Natural numbers

$\mathbb{Z} = \{\dots, -3, -2, -1, 0, 1, 2, 3, \dots\}$ Integers

$\mathbb{Z}^+ = \{1, 2, 3, \dots\}$ Positive integers

\mathbb{R} Real numbers

\mathbb{R}^+ Positive real numbers

\mathbb{Q} Rational numbers

\mathbb{C} Complex numbers

\emptyset Empty set

Sets defined using comprehension

- $S = \{x \mid P(x)\}$ where $P(x)$ is a predicate

Sets defined using comprehension

- $S = \{x \mid P(x)\}$ where $P(x)$ is a predicate
- Example Subsets of sets upon which an order is defined

$$\begin{array}{lll} [a, b] & = & \{x \mid a \leq x \leq b\} \quad \text{closed interval} \\ [a, b) & = & \{x \mid a \leq x < b\} \\ (a, b] & = & \{x \mid a < x \leq b\} \\ (a, b) & = & \{x \mid a < x < b\} \quad \text{open interval} \end{array}$$

Notation

- $x \in S$ membership

Notation

- $x \in S$ membership
- $A \cup B$ union; $A \cap B$ intersection; $A - B$ difference

Notation

- $x \in S$ membership
- $A \cup B$ union; $A \cap B$ intersection; $A - B$ difference
- $A \subseteq B$ subset; $A \supseteq B$ superset

Notation

- $x \in S$ membership
- $A \cup B$ union; $A \cap B$ intersection; $A - B$ difference
- $A \subseteq B$ subset; $A \supseteq B$ superset
- $A = B$ set equality

Notation

- $x \in S$ membership
- $A \cup B$ union; $A \cap B$ intersection; $A - B$ difference
- $A \subseteq B$ subset; $A \supseteq B$ superset
- $A = B$ set equality
- $\mathcal{P}(A)$ powerset (set of all subsets of A); also 2^A

Notation

- $x \in S$ membership
- $A \cup B$ union; $A \cap B$ intersection; $A - B$ difference
- $A \subseteq B$ subset; $A \supseteq B$ superset
- $A = B$ set equality
- $\mathcal{P}(A)$ powerset (set of all subsets of A); also 2^A
- $|A|$ cardinality

Notation

- $x \in S$ membership
- $A \cup B$ union; $A \cap B$ intersection; $A - B$ difference
- $A \subseteq B$ subset; $A \supseteq B$ superset
- $A = B$ set equality
- $\mathcal{P}(A)$ powerset (set of all subsets of A); also 2^A
- $|A|$ cardinality
- $A \times B$ cartesian product (tuple sets)

A proper mathematical definition of set is complicated
(Russell's paradox)

A proper mathematical definition of set is complicated (Russell's paradox)

- The set of cats is not a member of itself

A proper mathematical definition of set is complicated (Russell's paradox)

- The set of cats is not a member of itself
- The set of non-cats (all things that are not cats) is a member of itself

A proper mathematical definition of set is complicated (Russell's paradox)

- The set of cats is not a member of itself
- The set of non-cats (all things that are not cats) is a member of itself
- Let S be the set of all sets which are not members of themselves

A proper mathematical definition of set is complicated (Russell's paradox)

- The set of cats is not a member of itself
- The set of non-cats (all things that are not cats) is a member of itself
- Let S be the set of all sets which are not members of themselves
- $S = \{x \mid x \notin x\}$ (using naive comprehension)

A proper mathematical definition of set is complicated (Russell's paradox)

- The set of cats is not a member of itself
- The set of non-cats (all things that are not cats) is a member of itself
- Let S be the set of all sets which are not members of themselves
- $S = \{x \mid x \notin x\}$ (using naive comprehension)
- Question: is S a member of itself ($S \in S$) ?

A proper mathematical definition of set is complicated (Russell's paradox)

- The set of cats is not a member of itself
- The set of non-cats (all things that are not cats) is a member of itself
- Let S be the set of all sets which are not members of themselves
- $S = \{x \mid x \notin x\}$ (using naive comprehension)
- Question: is S a member of itself ($S \in S$) ?
- $S \in S$ provided that $S \notin S$; $S \notin S$ provided that $S \in S$

A proper mathematical definition of set is complicated (Russell's paradox)

- The set of cats is not a member of itself
- The set of non-cats (all things that are not cats) is a member of itself
- Let S be the set of all sets which are not members of themselves
- $S = \{x \mid x \notin x\}$ (using naive comprehension)
- Question: is S a member of itself ($S \in S$) ?
- $S \in S$ provided that $S \notin S$; $S \notin S$ provided that $S \in S$
- Modern formulations (such as Zermelo-Fraenkel set theory) restrict comprehension. (However, it is impossible to prove in ZF that ZF is consistent unless ZF is inconsistent.)

Functions

- Assume A and B are non-empty sets

Functions

- Assume A and B are non-empty sets
- A function f from A to B is an assignment of exactly one element of B to each element of A

Functions

- Assume A and B are non-empty sets
- A function f from A to B is an assignment of exactly one element of B to each element of A
- $f(a) = b$ if f assigns b to a

Functions

- Assume A and B are non-empty sets
- A function f from A to B is an assignment of exactly one element of B to each element of A
- $f(a) = b$ if f assigns b to a
- $f : A \rightarrow B$ if f is a function from A to B

One-to-one or injective functions

Definition

$f : A \rightarrow B$ is injective iff $\forall a, c \in A$ (if $f(a) = f(c)$ then $a = c$)

One-to-one or injective functions

Definition

$f : A \rightarrow B$ is injective iff $\forall a, c \in A$ (if $f(a) = f(c)$ then $a = c$)

- Is the identity function $\iota_A : A \rightarrow A$ injective?

One-to-one or injective functions

Definition

$f : A \rightarrow B$ is injective iff $\forall a, c \in A$ (if $f(a) = f(c)$ then $a = c$)

- Is the identity function $\iota_A : A \rightarrow A$ injective?

YES

One-to-one or injective functions

Definition

$f : A \rightarrow B$ is injective iff $\forall a, c \in A$ (if $f(a) = f(c)$ then $a = c$)

- Is the identity function $\iota_A : A \rightarrow A$ injective? YES
- Is the function $\sqrt{\cdot} : \mathbb{Z}^+ \rightarrow \mathbb{R}^+$ injective?

One-to-one or injective functions

Definition

$f : A \rightarrow B$ is injective iff $\forall a, c \in A$ (if $f(a) = f(c)$ then $a = c$)

- Is the identity function $\iota_A : A \rightarrow A$ injective? YES
- Is the function $\sqrt{\cdot} : \mathbb{Z}^+ \rightarrow \mathbb{R}^+$ injective? YES

One-to-one or injective functions

Definition

$f : A \rightarrow B$ is injective iff $\forall a, c \in A$ (if $f(a) = f(c)$ then $a = c$)

- Is the identity function $\iota_A : A \rightarrow A$ injective? YES
- Is the function $\sqrt{\cdot} : \mathbb{Z}^+ \rightarrow \mathbb{R}^+$ injective? YES
- Is the squaring function $.^2 : \mathbb{Z} \rightarrow \mathbb{Z}$ injective?

One-to-one or injective functions

Definition

$f : A \rightarrow B$ is injective iff $\forall a, c \in A$ (if $f(a) = f(c)$ then $a = c$)

- Is the identity function $\iota_A : A \rightarrow A$ injective? YES
- Is the function $\sqrt{\cdot} : \mathbb{Z}^+ \rightarrow \mathbb{R}^+$ injective? YES
- Is the squaring function $.^2 : \mathbb{Z} \rightarrow \mathbb{Z}$ injective? NO

One-to-one or injective functions

Definition

$f : A \rightarrow B$ is injective iff $\forall a, c \in A$ (if $f(a) = f(c)$ then $a = c$)

- Is the identity function $\iota_A : A \rightarrow A$ injective? YES
- Is the function $\sqrt{\cdot} : \mathbb{Z}^+ \rightarrow \mathbb{R}^+$ injective? YES
- Is the squaring function $.^2 : \mathbb{Z} \rightarrow \mathbb{Z}$ injective? NO
- Is the function $|\cdot| : \mathbb{R} \rightarrow \mathbb{R}$ injective?

One-to-one or injective functions

Definition

$f : A \rightarrow B$ is injective iff $\forall a, c \in A$ (if $f(a) = f(c)$ then $a = c$)

- Is the identity function $\iota_A : A \rightarrow A$ injective? YES
- Is the function $\sqrt{\cdot} : \mathbb{Z}^+ \rightarrow \mathbb{R}^+$ injective? YES
- Is the squaring function $.^2 : \mathbb{Z} \rightarrow \mathbb{Z}$ injective? NO
- Is the function $|\cdot| : \mathbb{R} \rightarrow \mathbb{R}$ injective? NO

One-to-one or injective functions

Definition

$f : A \rightarrow B$ is injective iff $\forall a, c \in A$ (if $f(a) = f(c)$ then $a = c$)

- Is the identity function $\iota_A : A \rightarrow A$ injective? YES
- Is the function $\sqrt{\cdot} : \mathbb{Z}^+ \rightarrow \mathbb{R}^+$ injective? YES
- Is the squaring function $.^2 : \mathbb{Z} \rightarrow \mathbb{Z}$ injective? NO
- Is the function $|\cdot| : \mathbb{R} \rightarrow \mathbb{R}$ injective? NO
- Assume $m > 1$. Is $\text{mod } m : \mathbb{Z} \rightarrow \{0, \dots, m-1\}$ injective?

One-to-one or injective functions

Definition

$f : A \rightarrow B$ is injective iff $\forall a, c \in A$ (if $f(a) = f(c)$ then $a = c$)

- Is the identity function $\iota_A : A \rightarrow A$ injective? YES
- Is the function $\sqrt{\cdot} : \mathbb{Z}^+ \rightarrow \mathbb{R}^+$ injective? YES
- Is the squaring function $.^2 : \mathbb{Z} \rightarrow \mathbb{Z}$ injective? NO
- Is the function $|\cdot| : \mathbb{R} \rightarrow \mathbb{R}$ injective? NO
- Assume $m > 1$. Is $\text{mod } m : \mathbb{Z} \rightarrow \{0, \dots, m-1\}$ injective? NO

Onto or surjective functions

Definition

$f : A \rightarrow B$ is surjective iff $\forall b \in B \exists a \in A (f(a) = b)$

Onto or surjective functions

Definition

$f : A \rightarrow B$ is surjective iff $\forall b \in B \exists a \in A (f(a) = b)$

- Is the identity function $\iota_A : A \rightarrow A$ surjective?

Onto or surjective functions

Definition

$f : A \rightarrow B$ is surjective iff $\forall b \in B \exists a \in A (f(a) = b)$

- Is the identity function $\iota_A : A \rightarrow A$ surjective?

YES

Onto or surjective functions

Definition

$f : A \rightarrow B$ is surjective iff $\forall b \in B \exists a \in A (f(a) = b)$

- Is the identity function $\iota_A : A \rightarrow A$ surjective? YES
- Is the function $\sqrt{\cdot} : \mathbb{Z}^+ \rightarrow \mathbb{R}^+$ surjective?

Onto or surjective functions

Definition

$f : A \rightarrow B$ is surjective iff $\forall b \in B \exists a \in A (f(a) = b)$

- Is the identity function $\iota_A : A \rightarrow A$ surjective?
- Is the function $\sqrt{\cdot} : \mathbb{Z}^+ \rightarrow \mathbb{R}^+$ surjective?

YES

NO

Onto or surjective functions

Definition

$f : A \rightarrow B$ is surjective iff $\forall b \in B \exists a \in A (f(a) = b)$

- Is the identity function $\iota_A : A \rightarrow A$ surjective? YES
- Is the function $\sqrt{\cdot} : \mathbb{Z}^+ \rightarrow \mathbb{R}^+$ surjective? NO
- Is the function $\cdot^2 : \mathbb{Z} \rightarrow \mathbb{Z}$ surjective?

Onto or surjective functions

Definition

$f : A \rightarrow B$ is surjective iff $\forall b \in B \exists a \in A (f(a) = b)$

- Is the identity function $\iota_A : A \rightarrow A$ surjective? YES
- Is the function $\sqrt{\cdot} : \mathbb{Z}^+ \rightarrow \mathbb{R}^+$ surjective? NO
- Is the function $\cdot^2 : \mathbb{Z} \rightarrow \mathbb{Z}$ surjective? NO

Onto or surjective functions

Definition

$f : A \rightarrow B$ is surjective iff $\forall b \in B \exists a \in A (f(a) = b)$

- Is the identity function $\iota_A : A \rightarrow A$ surjective? YES
- Is the function $\sqrt{\cdot} : \mathbb{Z}^+ \rightarrow \mathbb{R}^+$ surjective? NO
- Is the function $\cdot^2 : \mathbb{Z} \rightarrow \mathbb{Z}$ surjective? NO
- Is the function $|\cdot| : \mathbb{R} \rightarrow \mathbb{R}$ surjective?

Onto or surjective functions

Definition

$f : A \rightarrow B$ is surjective iff $\forall b \in B \exists a \in A (f(a) = b)$

- Is the identity function $\iota_A : A \rightarrow A$ surjective? YES
- Is the function $\sqrt{\cdot} : \mathbb{Z}^+ \rightarrow \mathbb{R}^+$ surjective? NO
- Is the function $\cdot^2 : \mathbb{Z} \rightarrow \mathbb{Z}$ surjective? NO
- Is the function $|\cdot| : \mathbb{R} \rightarrow \mathbb{R}$ surjective? NO

Onto or surjective functions

Definition

$f : A \rightarrow B$ is surjective iff $\forall b \in B \exists a \in A (f(a) = b)$

- Is the identity function $\iota_A : A \rightarrow A$ surjective? YES
- Is the function $\sqrt{\cdot} : \mathbb{Z}^+ \rightarrow \mathbb{R}^+$ surjective? NO
- Is the function $\cdot^2 : \mathbb{Z} \rightarrow \mathbb{Z}$ surjective? NO
- Is the function $|\cdot| : \mathbb{R} \rightarrow \mathbb{R}$ surjective? NO
- Assume $m > 1$. Is $\text{mod } m : \mathbb{Z} \rightarrow \{0, \dots, m-1\}$ surjective?

Onto or surjective functions

Definition

$f : A \rightarrow B$ is surjective iff $\forall b \in B \exists a \in A (f(a) = b)$

- Is the identity function $\iota_A : A \rightarrow A$ surjective? YES
- Is the function $\sqrt{\cdot} : \mathbb{Z}^+ \rightarrow \mathbb{R}^+$ surjective? NO
- Is the function $\cdot^2 : \mathbb{Z} \rightarrow \mathbb{Z}$ surjective? NO
- Is the function $|\cdot| : \mathbb{R} \rightarrow \mathbb{R}$ surjective? NO
- Assume $m > 1$. Is $\text{mod } m : \mathbb{Z} \rightarrow \{0, \dots, m-1\}$ surjective? YES

One-to-one correspondence or bijection

Definition

$f : A \rightarrow B$ is a bijection iff it is both injective and surjective

One-to-one correspondence or bijection

Definition

$f : A \rightarrow B$ is a bijection iff it is both injective and surjective

- Is the identity function $\iota_A : A \rightarrow A$ a bijection?

One-to-one correspondence or bijection

Definition

$f : A \rightarrow B$ is a bijection iff it is both injective and surjective

- Is the identity function $\iota_A : A \rightarrow A$ a bijection? YES

One-to-one correspondence or bijection

Definition

$f : A \rightarrow B$ is a bijection iff it is both injective and surjective

- Is the identity function $\iota_A : A \rightarrow A$ a bijection? YES
- Is the function $\sqrt{\cdot} : \mathbb{R}^+ \rightarrow \mathbb{R}^+$ a bijection?

One-to-one correspondence or bijection

Definition

$f : A \rightarrow B$ is a bijection iff it is both injective and surjective

- Is the identity function $\iota_A : A \rightarrow A$ a bijection? YES
- Is the function $\sqrt{\cdot} : \mathbb{R}^+ \rightarrow \mathbb{R}^+$ a bijection? YES

One-to-one correspondence or bijection

Definition

$f : A \rightarrow B$ is a bijection iff it is both injective and surjective

- Is the identity function $\iota_A : A \rightarrow A$ a bijection? YES
- Is the function $\sqrt{\cdot} : \mathbb{R}^+ \rightarrow \mathbb{R}^+$ a bijection? YES
- Is the function $\cdot^2 : \mathbb{R} \rightarrow \mathbb{R}$ a bijection?

One-to-one correspondence or bijection

Definition

$f : A \rightarrow B$ is a bijection iff it is both injective and surjective

- Is the identity function $\iota_A : A \rightarrow A$ a bijection? YES
- Is the function $\sqrt{\cdot} : \mathbb{R}^+ \rightarrow \mathbb{R}^+$ a bijection? YES
- Is the function $\cdot^2 : \mathbb{R} \rightarrow \mathbb{R}$ a bijection? NO

One-to-one correspondence or bijection

Definition

$f : A \rightarrow B$ is a bijection iff it is both injective and surjective

- Is the identity function $\iota_A : A \rightarrow A$ a bijection? YES
- Is the function $\sqrt{\cdot} : \mathbb{R}^+ \rightarrow \mathbb{R}^+$ a bijection? YES
- Is the function $\cdot^2 : \mathbb{R} \rightarrow \mathbb{R}$ a bijection? NO
- Is the function $|\cdot| : \mathbb{R} \rightarrow \mathbb{R}$ a bijection?

One-to-one correspondence or bijection

Definition

$f : A \rightarrow B$ is a bijection iff it is both injective and surjective

- Is the identity function $\iota_A : A \rightarrow A$ a bijection? YES
- Is the function $\sqrt{\cdot} : \mathbb{R}^+ \rightarrow \mathbb{R}^+$ a bijection? YES
- Is the function $\cdot^2 : \mathbb{R} \rightarrow \mathbb{R}$ a bijection? NO
- Is the function $|\cdot| : \mathbb{R} \rightarrow \mathbb{R}$ a bijection? NO

Function composition

Definition

Let $f : B \rightarrow C$ and $g : A \rightarrow B$. The composition function $f \circ g : A \rightarrow C$ is
 $(f \circ g)(a) = f(g(a))$

Results about function composition

Theorem

The composition of two functions is a function

Results about function composition

Theorem

The composition of two functions is a function

Theorem

The composition of two injective functions is an injective function

Results about function composition

Theorem

The composition of two functions is a function

Theorem

The composition of two injective functions is an injective function

Theorem

The composition of two surjective functions is a surjective function

Results about function composition

Theorem

The composition of two functions is a function

Theorem

The composition of two injective functions is an injective function

Theorem

The composition of two surjective functions is a surjective function

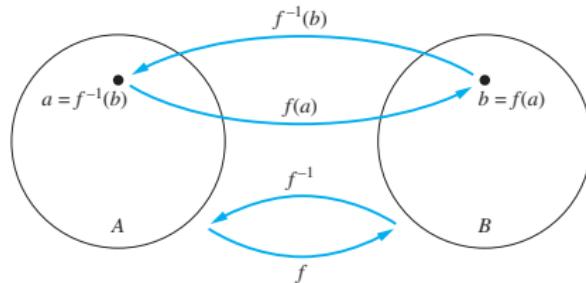
Corollary

The composition of two bijections is a bijection

Inverse function

Definition

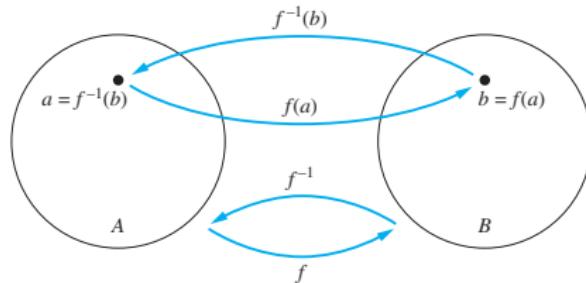
If $f : A \rightarrow B$ is a bijection, then the inverse of f , written $f^{-1} : B \rightarrow A$ is $f^{-1}(b) = a$ iff $f(a) = b$



Inverse function

Definition

If $f : A \rightarrow B$ is a bijection, then the inverse of f , written $f^{-1} : B \rightarrow A$ is $f^{-1}(b) = a$ iff $f(a) = b$

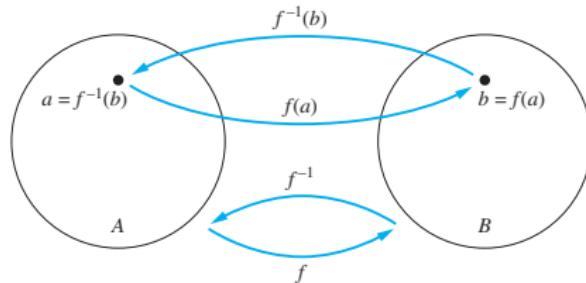


What is the inverse of $\iota_A : A \rightarrow A$?

Inverse function

Definition

If $f : A \rightarrow B$ is a bijection, then the inverse of f , written $f^{-1} : B \rightarrow A$ is $f^{-1}(b) = a$ iff $f(a) = b$



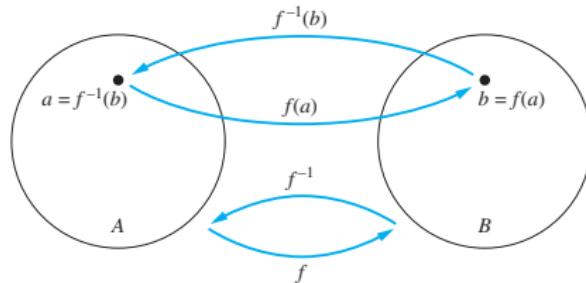
What is the inverse of $\iota_A : A \rightarrow A$?

What is the inverse of $\sqrt{\cdot} : \mathbb{R}^+ \rightarrow \mathbb{R}^+$?

Inverse function

Definition

If $f : A \rightarrow B$ is a bijection, then the inverse of f , written $f^{-1} : B \rightarrow A$ is $f^{-1}(b) = a$ iff $f(a) = b$



What is the inverse of $\iota_A : A \rightarrow A$?

What is the inverse of $\sqrt{\cdot} : \mathbb{R}^+ \rightarrow \mathbb{R}^+$?

What is $f^{-1} \circ f$? and $f \circ f^{-1}$?

The floor and ceiling functions

Definition

The floor function $\lfloor \cdot \rfloor : \mathbb{R} \rightarrow \mathbb{Z}$ is $\lfloor x \rfloor$ equals the largest integer less than or equal to x

Definition

The ceiling function $\lceil \cdot \rceil : \mathbb{R} \rightarrow \mathbb{Z}$ is $\lceil x \rceil$ equals the smallest integer greater than or equal to x

The floor and ceiling functions

Definition

The floor function $\lfloor \cdot \rfloor : \mathbb{R} \rightarrow \mathbb{Z}$ is $\lfloor x \rfloor$ equals the largest integer less than or equal to x

Definition

The ceiling function $\lceil \cdot \rceil : \mathbb{R} \rightarrow \mathbb{Z}$ is $\lceil x \rceil$ equals the smallest integer greater than or equal to x

$$\left\lfloor \frac{1}{2} \right\rfloor = \left\lceil -\frac{1}{2} \right\rceil = \lfloor 0 \rfloor = \lceil 0 \rceil = 0$$

The floor and ceiling functions

Definition

The floor function $\lfloor \cdot \rfloor : \mathbb{R} \rightarrow \mathbb{Z}$ is $\lfloor x \rfloor$ equals the largest integer less than or equal to x

Definition

The ceiling function $\lceil \cdot \rceil : \mathbb{R} \rightarrow \mathbb{Z}$ is $\lceil x \rceil$ equals the smallest integer greater than or equal to x

$$\left\lfloor \frac{1}{2} \right\rfloor = \left\lceil -\frac{1}{2} \right\rceil = \lfloor 0 \rfloor = \lceil 0 \rceil = 0$$

$$\lfloor -6.1 \rfloor = -7 \quad \lceil 6.1 \rceil = 7$$

Useful tips about floors and ceilings

- When showing properties of floors is to let $x = n + \epsilon$ if $\lfloor x \rfloor = n$ where $0 \leq \epsilon < 1$
- Similarly, for ceilings let $x = n - \epsilon$ if $\lceil x \rceil = n$ where $0 \leq \epsilon < 1$

Useful tips about floors and ceilings

- When showing properties of floors is to let $x = n + \epsilon$ if $\lfloor x \rfloor = n$ where $0 \leq \epsilon < 1$
- Similarly, for ceilings let $x = n - \epsilon$ if $\lceil x \rceil = n$ where $0 \leq \epsilon < 1$
- Prove

$$\forall x \in \mathbb{R} (\lfloor 2x \rfloor = \lfloor x \rfloor + \lfloor x + 1/2 \rfloor)$$

Useful tips about floors and ceilings

- When showing properties of floors is to let $x = n + \epsilon$ if $\lfloor x \rfloor = n$ where $0 \leq \epsilon < 1$
- Similarly, for ceilings let $x = n - \epsilon$ if $\lceil x \rceil = n$ where $0 \leq \epsilon < 1$
- Prove

$$\forall x \in \mathbb{R} (\lfloor 2x \rfloor = \lfloor x \rfloor + \lfloor x + 1/2 \rfloor)$$

- Proof in book

Prove $\lceil x \rceil + \lceil y \rceil = \lceil x + y \rceil$

Prove $\lceil x \rceil + \lceil y \rceil = \lceil x + y \rceil$

False; counterexample $x = 1/2$ and $y = 1/2$

The factorial function

Definition

The factorial function $f : \mathbb{N} \rightarrow \mathbb{N}$, denoted as $f(n) = n!$ assigns to n the product of the first n positive integers

$$f(0) = 0! = 1$$

and

$$f(n) = n! = 1 \cdot 2 \cdot \dots \cdot (n-1) \cdot n$$

Relations

Definition

A binary relation R on sets A and B is a subset $R \subseteq A \times B$

Relations

Definition

A binary relation R on sets A and B is a subset $R \subseteq A \times B$

- R is a set of tuples (a, b) with $a \in A$ and $b \in B$

Relations

Definition

A binary relation R on sets A and B is a subset $R \subseteq A \times B$

- R is a set of tuples (a, b) with $a \in A$ and $b \in B$
- Often we write $a R b$ for $(a, b) \in R$

Relations

Definition

A binary relation R on sets A and B is a subset $R \subseteq A \times B$

- R is a set of tuples (a, b) with $a \in A$ and $b \in B$
- Often we write $a R b$ for $(a, b) \in R$
- R is a relation on A if $B = A$

Relations

Definition

A binary relation R on sets A and B is a subset $R \subseteq A \times B$

- R is a set of tuples (a, b) with $a \in A$ and $b \in B$
- Often we write $a R b$ for $(a, b) \in R$
- R is a relation on A if $B = A$

Relations

Definition

A binary relation R on sets A and B is a subset $R \subseteq A \times B$

- R is a set of tuples (a, b) with $a \in A$ and $b \in B$
- Often we write $a R b$ for $(a, b) \in R$
- R is a relation on A if $B = A$

Definition

Given sets A_1, \dots, A_n , a subset $R \subseteq A_1 \times \dots \times A_n$ is an n -ary relation

Examples

- Divides $| : \mathbb{Z}^+ \rightarrow \mathbb{Z}^+$ is $\{(n, m) \mid \exists k \in \mathbb{Z}^+ (m = kn)\}$

Examples

- Divides $| : \mathbb{Z}^+ \rightarrow \mathbb{Z}^+$ is $\{(n, m) \mid \exists k \in \mathbb{Z}^+ (m = kn)\}$
- Let $m > 1$ be an integer. $R = \{(a, b) \mid a \bmod m = b \bmod m\}$

Examples

- Divides $| : \mathbb{Z}^+ \rightarrow \mathbb{Z}^+$ is $\{(n, m) \mid \exists k \in \mathbb{Z}^+ (m = kn)\}$
- Let $m > 1$ be an integer. $R = \{(a, b) \mid a \bmod m = b \bmod m\}$
- Written as $a = b \pmod{m}$

Properties of binary relations

A binary relation R on A is called

- reflexive iff $\forall x \in A (x, x) \in R$

Properties of binary relations

A binary relation R on A is called

- reflexive iff $\forall x \in A (x, x) \in R$
- \leq , $=$, and $|$ are reflexive, but $<$ is not

Properties of binary relations

A binary relation R on A is called

- reflexive iff $\forall x \in A (x, x) \in R$
- \leq , $=$, and $|$ are reflexive, but $<$ is not
- symmetric iff $\forall x, y \in A ((x, y) \in R \rightarrow (y, x) \in R)$
- $=$ is symmetric, but \leq , $<$, and $|$ are not

Properties of binary relations

A binary relation R on A is called

- reflexive iff $\forall x \in A (x, x) \in R$
- \leq , $=$, and $|$ are reflexive, but $<$ is not
- symmetric iff $\forall x, y \in A ((x, y) \in R \rightarrow (y, x) \in R)$
- $=$ is symmetric, but \leq , $<$, and $|$ are not
- antisymmetric iff $\forall x, y \in A (((x, y) \in R \wedge (y, x) \in R) \rightarrow x = y)$

Properties of binary relations

A binary relation R on A is called

- reflexive iff $\forall x \in A (x, x) \in R$
- \leq , $=$, and $|$ are reflexive, but $<$ is not
- symmetric iff $\forall x, y \in A ((x, y) \in R \rightarrow (y, x) \in R)$
- $=$ is symmetric, but \leq , $<$, and $|$ are not
- antisymmetric iff $\forall x, y \in A (((x, y) \in R \wedge (y, x) \in R) \rightarrow x = y)$
- \leq , $=$, $<$, and $|$ are antisymmetric

Properties of binary relations

A binary relation R on A is called

- reflexive iff $\forall x \in A (x, x) \in R$
- \leq , $=$, and $|$ are reflexive, but $<$ is not
- symmetric iff $\forall x, y \in A ((x, y) \in R \rightarrow (y, x) \in R)$
- $=$ is symmetric, but \leq , $<$, and $|$ are not
- antisymmetric iff $\forall x, y \in A (((x, y) \in R \wedge (y, x) \in R) \rightarrow x = y)$
- \leq , $=$, $<$, and $|$ are antisymmetric
- transitive iff $\forall x, y, z \in A (((x, y) \in R \wedge (y, z) \in R) \rightarrow (x, z) \in R)$
- \leq , $=$, $<$, and $|$ are transitive

Equivalence relations

Definition

A relation R on a set A is an equivalence relation iff it is reflexive, symmetric and transitive

Equivalence relations

Definition

A relation R on a set A is an equivalence relation iff it is reflexive, symmetric and transitive

- Let Σ^* be the set of strings over alphabet Σ . The relation $\{(s, t) \in \Sigma^* \times \Sigma^* \mid |s| = |t|\}$ is an equivalence relation

Equivalence relations

Definition

A relation R on a set A is an equivalence relation iff it is reflexive, symmetric and transitive

- Let Σ^* be the set of strings over alphabet Σ . The relation $\{(s, t) \in \Sigma^* \times \Sigma^* \mid |s| = |t|\}$ is an equivalence relation
- | on integers is not an equivalence relation.

Equivalence relations

Definition

A relation R on a set A is an equivalence relation iff it is reflexive, symmetric and transitive

- Let Σ^* be the set of strings over alphabet Σ . The relation $\{(s, t) \in \Sigma^* \times \Sigma^* \mid |s| = |t|\}$ is an equivalence relation
- $|$ on integers is not an equivalence relation.
- For $m > 1$ be an integer the relation $= (\text{mod } m)$ is an equivalence relation on integers

Equivalence classes

Definition

Let R be an equivalence relation on a set A and $a \in A$. Let

$$[a]_R = \{s \mid (a, s) \in R\}$$

be the equivalence class of a w.r.t. R

If $b \in [a]_R$ then b is called a representative of the equivalence class.
Every member of the class can be a representative

Theorem

Result

Let R be an equivalence on A and $a, b \in A$. The following three statements are equivalent

- ① aRb
- ② $[a]_R = [b]_R$
- ③ $[a]_R \cap [b]_R \neq \emptyset$

Theorem

Result

Let R be an equivalence on A and $a, b \in A$. The following three statements are equivalent

- ① aRb
- ② $[a]_R = [b]_R$
- ③ $[a]_R \cap [b]_R \neq \emptyset$

Proof in book

Partitions of a set

Definition

A partition of a set A is a collection of disjoint, nonempty subsets that have A as their union. In other words, the collection of subsets $A_i \subseteq A$ with $i \in I$ (where I is an index set) forms a partition of A iff

- ① $A_i \neq \emptyset$ for all $i \in I$
- ② $A_i \cap A_j = \emptyset$ for all $i \neq j \in I$
- ③ $\bigcup_{i \in I} A_i = A$

Result

Theorem

- ① If R is an equivalence on A , then the equivalence classes of R form a partition of A
- ② Conversely, given a partition $\{A_i \mid i \in I\}$ of A there exists an equivalence relation R that has exactly the sets $A_i, i \in I$, as its equivalence classes

Result

Theorem

- ① If R is an equivalence on A , then the equivalence classes of R form a partition of A
- ② Conversely, given a partition $\{A_i \mid i \in I\}$ of A there exists an equivalence relation R that has exactly the sets $A_i, i \in I$, as its equivalence classes

Proof in book

Discrete Mathematics & Mathematical Reasoning

Sequences and Sums

Colin Stirling

Informatics

Slides based on ones by Myrto Arapinis

Sequences

Sequences are ordered lists of elements

2, 3, 5, 7, 11, 13, 17, 19, ... or a, b, c, d, \dots, y, z

Sequences

Sequences are ordered lists of elements

2, 3, 5, 7, 11, 13, 17, 19, ... or a, b, c, d, \dots, y, z

Definition

A sequence over a set S is a function f from a subset of the integers (typically \mathbb{N} or \mathbb{Z}^+) to the set S . If the domain of f is finite then the sequence is finite

Examples

$f : \mathbb{Z}^+ \rightarrow \mathbb{Q}$ is $f(n) = 1/n$ defines the sequence

Examples

$f : \mathbb{Z}^+ \rightarrow \mathbb{Q}$ is $f(n) = 1/n$ defines the sequence

$$1, 1/2, 1/3, 1/4, \dots$$

Examples

$f : \mathbb{Z}^+ \rightarrow \mathbb{Q}$ is $f(n) = 1/n$ defines the sequence

$$1, 1/2, 1/3, 1/4, \dots$$

Assuming $a_n = f(n)$, the sequence is also written a_1, a_2, a_3, \dots
or as $\{a_n\}_{n \in \mathbb{Z}^+}$

Examples

$f : \mathbb{Z}^+ \rightarrow \mathbb{Q}$ is $f(n) = 1/n$ defines the sequence

$$1, 1/2, 1/3, 1/4, \dots$$

Assuming $a_n = f(n)$, the sequence is also written a_1, a_2, a_3, \dots
or as $\{a_n\}_{n \in \mathbb{Z}^+}$

$g : \mathbb{N} \rightarrow \mathbb{N}$ is $g(n) = n^2$ defines the sequence

$$0, 1, 4, 9, \dots$$

Examples

$f : \mathbb{Z}^+ \rightarrow \mathbb{Q}$ is $f(n) = 1/n$ defines the sequence

$$1, 1/2, 1/3, 1/4, \dots$$

Assuming $a_n = f(n)$, the sequence is also written a_1, a_2, a_3, \dots
or as $\{a_n\}_{n \in \mathbb{Z}^+}$

$g : \mathbb{N} \rightarrow \mathbb{N}$ is $g(n) = n^2$ defines the sequence

$$0, 1, 4, 9, \dots$$

Assuming $b_n = g(n)$, also written b_0, b_1, b_2, \dots
or as $\{b_n\}_{n \in \mathbb{N}}$

Geometric and arithmetic progressions

- A **geometric progression** is a sequence of the form

$$a, ar, ar^2, ar^3, \dots, ar^n, \dots$$

Geometric and arithmetic progressions

- A **geometric progression** is a sequence of the form

$$a, ar, ar^2, ar^3, \dots, ar^n, \dots$$

- **Example** $\{b_n\}_{n \in \mathbb{N}}$ with $b_n = (-1)^n$

Geometric and arithmetic progressions

- A **geometric progression** is a sequence of the form

$$a, ar, ar^2, ar^3, \dots, ar^n, \dots$$

- **Example** $\{b_n\}_{n \in \mathbb{N}}$ with $b_n = (-1)^n$
- An **arithmetic progression** is a sequence of the form

$$a, a+d, a+2d, a+3d, \dots, a+nd, \dots$$

Geometric and arithmetic progressions

- A **geometric progression** is a sequence of the form

$$a, ar, ar^2, ar^3, \dots, ar^n, \dots$$

- **Example** $\{b_n\}_{n \in \mathbb{N}}$ with $b_n = (-1)^n$
- An **arithmetic progression** is a sequence of the form

$$a, a+d, a+2d, a+3d, \dots, a+nd, \dots$$

- **Example** $\{c_n\}_{n \in \mathbb{N}}$ with $c_n = 7 - 3n$

Geometric and arithmetic progressions

- A **geometric progression** is a sequence of the form

$$a, ar, ar^2, ar^3, \dots, ar^n, \dots$$

- **Example** $\{b_n\}_{n \in \mathbb{N}}$ with $b_n = (-1)^n$
- An **arithmetic progression** is a sequence of the form

$$a, a+d, a+2d, a+3d, \dots, a+nd, \dots$$

- **Example** $\{c_n\}_{n \in \mathbb{N}}$ with $c_n = 7 - 3n$

where the initial elements a , the common ratio r and the common difference d are real numbers

Recurrence relations

Definition

A recurrence relation for $\{a_n\}_{n \in \mathbb{N}}$ is an equation that expresses a_n in terms of one or more of the elements a_0, a_1, \dots, a_{n-1}

Recurrence relations

Definition

A recurrence relation for $\{a_n\}_{n \in \mathbb{N}}$ is an equation that expresses a_n in terms of one or more of the elements a_0, a_1, \dots, a_{n-1}

- Typically the recurrence relation expresses a_n in terms of just a fixed number of previous elements (such as $a_n = g(a_{n-1}, a_{n-2})$)

Recurrence relations

Definition

A recurrence relation for $\{a_n\}_{n \in \mathbb{N}}$ is an equation that expresses a_n in terms of one or more of the elements a_0, a_1, \dots, a_{n-1}

- Typically the recurrence relation expresses a_n in terms of just a fixed number of previous elements (such as $a_n = g(a_{n-1}, a_{n-2})$)
- The initial conditions specify the first elements of the sequence, before the recurrence relation applies

Recurrence relations

Definition

A recurrence relation for $\{a_n\}_{n \in \mathbb{N}}$ is an equation that expresses a_n in terms of one or more of the elements a_0, a_1, \dots, a_{n-1}

- Typically the recurrence relation expresses a_n in terms of just a fixed number of previous elements (such as $a_n = g(a_{n-1}, a_{n-2})$)
- The initial conditions specify the first elements of the sequence, before the recurrence relation applies
- A sequence is called a solution of a recurrence relation iff its terms satisfy the recurrence relation

Rabbits and Fibonacci sequence

A young pair of rabbits (one of each sex) is placed on an island

Rabbits and Fibonacci sequence

A young pair of rabbits (one of each sex) is placed on an island

A pair of rabbits does not breed until they are 2 months old. After they are 2 months old each pair produces another pair each month

Rabbits and Fibonacci sequence

A young pair of rabbits (one of each sex) is placed on an island

A pair of rabbits does not breed until they are 2 months old. After they are 2 months old each pair produces another pair each month

Find a recurrence relation for number of rabbits after n months assuming no rabbits die

Rabbits and Fibonacci sequence

A young pair of rabbits (one of each sex) is placed on an island

A pair of rabbits does not breed until they are 2 months old. After they are 2 months old each pair produces another pair each month

Find a recurrence relation for number of rabbits after n months assuming no rabbits die

Answer is the Fibonacci sequence

$$\begin{cases} f(0) = 0 \\ f(1) = 1 \\ f(n) = f(n-1) + f(n-2) \text{ for } n \geq 2 \end{cases}$$

Yields the sequence 0, 1, 1, 2, 3, 5, 8, 13, ...

Solving recurrence relations

- Finding a formula for the n^{th} term of the sequence generated by a recurrence relation is called solving the recurrence relation

Solving recurrence relations

- Finding a formula for the n^{th} term of the sequence generated by a recurrence relation is called solving the recurrence relation
- Such a formula is called a closed formula

Solving recurrence relations

- Finding a formula for the n^{th} term of the sequence generated by a recurrence relation is called solving the recurrence relation
- Such a formula is called a closed formula
- Various more advanced methods for solving recurrence relations are covered in Chapter 8 of the book (not part of this course)

Solving recurrence relations

- Finding a formula for the n^{th} term of the sequence generated by a recurrence relation is called solving the recurrence relation
- Such a formula is called a closed formula
- Various more advanced methods for solving recurrence relations are covered in Chapter 8 of the book (not part of this course)
- Here we illustrate by example the method of iteration in which we need to guess the formula

Solving recurrence relations

- Finding a formula for the n^{th} term of the sequence generated by a recurrence relation is called solving the recurrence relation
- Such a formula is called a closed formula
- Various more advanced methods for solving recurrence relations are covered in Chapter 8 of the book (not part of this course)
- Here we illustrate by example the method of iteration in which we need to guess the formula
- The guess can be proved correct by the method of induction (to be covered)

Iterative solution - working upwards

Forward substitution

Iterative solution - working upwards

Forward substitution

$$a_n = a_{n-1} + 3 \text{ for } n \geq 2 \text{ with } a_1 = 2$$

Iterative solution - working upwards

Forward substitution

$a_n = a_{n-1} + 3$ for $n \geq 2$ with $a_1 = 2$

$$a_2 = 2 + 3$$

Iterative solution - working upwards

Forward substitution

$a_n = a_{n-1} + 3$ for $n \geq 2$ with $a_1 = 2$

$$a_2 = 2 + 3$$

$$a_3 = (2 + 3) + 3 = 2 + 3 \cdot 2$$

Iterative solution - working upwards

Forward substitution

$a_n = a_{n-1} + 3$ for $n \geq 2$ with $a_1 = 2$

$$a_2 = 2 + 3$$

$$a_3 = (2 + 3) + 3 = 2 + 3 \cdot 2$$

$$a_4 = (2 + 2 \cdot 3) + 3 = 2 + 3 \cdot 3$$

Iterative solution - working upwards

Forward substitution

$a_n = a_{n-1} + 3$ for $n \geq 2$ with $a_1 = 2$

$$a_2 = 2 + 3$$

$$a_3 = (2 + 3) + 3 = 2 + 3 \cdot 2$$

$$a_4 = (2 + 2 \cdot 3) + 3 = 2 + 3 \cdot 3$$

⋮

$$a_n = a_{n-1} + 3 = (2 + 3 \cdot (n - 2)) + 3 = 2 + 3 \cdot (n - 1)$$

Iterative solution - working downward

Backward substitution

Iterative solution - working downward

Backward substitution

$a_n = a_{n-1} + 3$ for $n \geq 2$ with $a_1 = 2$

$$a_n = a_{n-1} + 3$$

Iterative solution - working downward

Backward substitution

$a_n = a_{n-1} + 3$ for $n \geq 2$ with $a_1 = 2$

$$\begin{aligned}a_n &= a_{n-1} + 3 \\&= (a_{n-2} + 3) + 3 = a_{n-2} + 3 \cdot 2\end{aligned}$$

Iterative solution - working downward

Backward substitution

$a_n = a_{n-1} + 3$ for $n \geq 2$ with $a_1 = 2$

$$\begin{aligned}a_n &= a_{n-1} + 3 \\&= (a_{n-2} + 3) + 3 = a_{n-2} + 3 \cdot 2 \\&= (a_{n-3} + 3) + 3 \cdot 2 = a_{n-3} + 3 \cdot 3\end{aligned}$$

Iterative solution - working downward

Backward substitution

$a_n = a_{n-1} + 3$ for $n \geq 2$ with $a_1 = 2$

$$\begin{aligned}a_n &= a_{n-1} + 3 \\&= (a_{n-2} + 3) + 3 = a_{n-2} + 3 \cdot 2 \\&= (a_{n-3} + 3) + 3 \cdot 2 = a_{n-3} + 3 \cdot 3 \\&\vdots \\&= a_2 + 3(n-2) = (a_1 + 3) + 3 \cdot (n-2) = 2 + 3 \cdot (n-1)\end{aligned}$$

Compound interest

- Suppose a person deposits £1000 in a savings account yielding 3% per year with interest compounded annually. How much is in the account after 20 years?

Compound interest

- Suppose a person deposits £1000 in a savings account yielding 3% per year with interest compounded annually. How much is in the account after 20 years?
- Let P_n denote amount after n years

Compound interest

- Suppose a person deposits £1000 in a savings account yielding 3% per year with interest compounded annually. How much is in the account after 20 years?
- Let P_n denote amount after n years
- $P_n = P_{n-1} + 0.03 P_{n-1} = (1.03)P_{n-1}$

Compound interest

- Suppose a person deposits £1000 in a savings account yielding 3% per year with interest compounded annually. How much is in the account after 20 years?
- Let P_n denote amount after n years
- $P_n = P_{n-1} + 0.03 P_{n-1} = (1.03)P_{n-1}$
- The initial condition $P_0 = 1000$.

Compound interest

- Suppose a person deposits £1000 in a savings account yielding 3% per year with interest compounded annually. How much is in the account after 20 years?
- Let P_n denote amount after n years
- $P_n = P_{n-1} + 0.03 P_{n-1} = (1.03)P_{n-1}$
- The initial condition $P_0 = 1000$.
- $P_1 = (1.03)P_0, \dots, P_n = (1.03)P_{n-1} = (1.03)^n P_0$

Compound interest

- Suppose a person deposits £1000 in a savings account yielding 3% per year with interest compounded annually. How much is in the account after 20 years?
- Let P_n denote amount after n years
- $P_n = P_{n-1} + 0.03 P_{n-1} = (1.03)P_{n-1}$
- The initial condition $P_0 = 1000$.
- $P_1 = (1.03)P_0, \dots, P_n = (1.03)P_{n-1} = (1.03)^n P_0$
- $P_{20} = (1.03)^{20} 1000 = 1,806$

Common sequences

TABLE 1 Some Useful Sequences.

<i>nth Term</i>	<i>First 10 Terms</i>
n^2	1, 4, 9, 16, 25, 36, 49, 64, 81, 100, ...
n^3	1, 8, 27, 64, 125, 216, 343, 512, 729, 1000, ...
n^4	1, 16, 81, 256, 625, 1296, 2401, 4096, 6561, 10000, ...
2^n	2, 4, 8, 16, 32, 64, 128, 256, 512, 1024, ...
3^n	3, 9, 27, 81, 243, 729, 2187, 6561, 19683, 59049, ...
$n!$	1, 2, 6, 24, 120, 720, 5040, 40320, 362880, 3628800, ...
f_n	1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, ...

Summations

Given a sequence $\{a_n\}$, the sum of terms $a_m, a_{m+1}, \dots, a_\ell$ is

$$a_m + a_{m+1} + \dots + a_\ell$$

Summations

Given a sequence $\{a_n\}$, the sum of terms $a_m, a_{m+1}, \dots, a_\ell$ is

$$a_m + a_{m+1} + \dots + a_\ell$$

$$\sum_{j=m}^{\ell} a_j \quad \text{or} \quad \sum_{m \leq j \leq \ell} a_j$$

Summations

Given a sequence $\{a_n\}$, the sum of terms $a_m, a_{m+1}, \dots, a_\ell$ is

$$a_m + a_{m+1} + \dots + a_\ell$$

$$\sum_{j=m}^{\ell} a_j \quad \text{or} \quad \sum_{m \leq j \leq \ell} a_j$$

The variable j is called the index of summation

More generally for an index set S

$$\sum_{j \in S} a_j$$

Useful summation formulas

TABLE 2 Some Useful Summation Formulae.

<i>Sum</i>	<i>Closed Form</i>
$\sum_{k=0}^n ar^k \ (r \neq 0)$	$\frac{ar^{n+1} - a}{r - 1}, r \neq 1$
$\sum_{k=1}^n k$	$\frac{n(n + 1)}{2}$
$\sum_{k=1}^n k^2$	$\frac{n(n + 1)(2n + 1)}{6}$
$\sum_{k=1}^n k^3$	$\frac{n^2(n + 1)^2}{4}$
$\sum_{k=0}^{\infty} x^k, x < 1$	$\frac{1}{1 - x}$
$\sum_{k=1}^{\infty} kx^{k-1}, x < 1$	$\frac{1}{(1 - x)^2}$

Products

Given a sequence $\{a_n\}$, the product of terms $a_m, a_{m+1}, \dots, a_\ell$ is

$$a_m \cdot a_{m+1} \cdot \dots \cdot a_\ell$$

$$\prod_{j=m}^{\ell} a_j \quad \text{or} \quad \prod_{m \leq j \leq \ell} a_j$$

More generally for a finite index set S one writes

$$\prod_{j \in S} a_j$$

Discrete Mathematics & Mathematical Reasoning

Cardinality

Colin Stirling

Informatics

Some slides based on ones by Myrto Arapinis and by Richard Mayr

Cardinality of Sets

Definition

- Two sets A and B have the same cardinality, $|A| = |B|$, iff there exists a bijection from A to B

Cardinality of Sets

Definition

- Two sets A and B have the same cardinality, $|A| = |B|$, iff there exists a bijection from A to B
- $|A| \leq |B|$ iff there exists an injection from A to B

Cardinality of Sets

Definition

- Two sets A and B have the same cardinality, $|A| = |B|$, iff there exists a bijection from A to B
- $|A| \leq |B|$ iff there exists an injection from A to B
- $|A| < |B|$ iff $|A| \leq |B|$ and $|A| \neq |B|$ (A smaller cardinality than B)

Cardinality of Sets

Definition

- Two sets A and B have the same cardinality, $|A| = |B|$, iff there exists a bijection from A to B
- $|A| \leq |B|$ iff there exists an injection from A to B
- $|A| < |B|$ iff $|A| \leq |B|$ and $|A| \neq |B|$ (A smaller cardinality than B)

When A and B are finite $|A| = |B|$ iff they have same size

Cardinality of Sets

Definition

- Two sets A and B have the same cardinality, $|A| = |B|$, iff there exists a bijection from A to B
- $|A| \leq |B|$ iff there exists an injection from A to B
- $|A| < |B|$ iff $|A| \leq |B|$ and $|A| \neq |B|$ (A smaller cardinality than B)

When A and B are finite $|A| = |B|$ iff they have same size

\mathbb{N} and its subset $\text{Even} = \{2n \mid n \in \mathbb{N}\}$ have the same cardinality, because $f : \mathbb{N} \rightarrow \text{Even}$ where $f(n) = 2n$ is a bijection

Countable Sets

Definition

- A set S is called countably infinite, iff it has the same cardinality as the natural numbers, $|S| = |\mathbb{N}|$

Countable Sets

Definition

- A set S is called countably infinite, iff it has the same cardinality as the natural numbers, $|S| = |\mathbb{N}|$
- A set is called countable iff it is either finite or countably infinite

Countable Sets

Definition

- A set S is called countably infinite, iff it has the same cardinality as the natural numbers, $|S| = |\mathbb{N}|$
- A set is called countable iff it is either finite or countably infinite
- A set that is not countable is called uncountable

The positive rational numbers are countable

Construct a bijection $f : \mathbb{N} \rightarrow \mathbb{Q}^+$

The positive rational numbers are countable

Construct a bijection $f : \mathbb{N} \rightarrow \mathbb{Q}^+$

List fractions p/q with $q = n$ in the n^{th} row

The positive rational numbers are countable

Construct a bijection $f : \mathbb{N} \rightarrow \mathbb{Q}^+$

List fractions p/q with $q = n$ in the n^{th} row

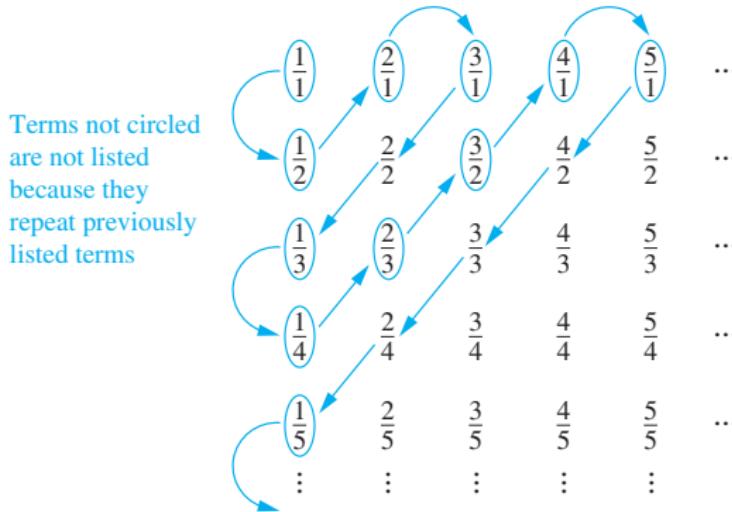
f traverses this list in the order for $m = 2, 3, 4, \dots$ visiting all p/q with $p + q = m$ (and listing only new rationals)

The positive rational numbers are countable

Construct a bijection $f : \mathbb{N} \rightarrow \mathbb{Q}^+$

List fractions p/q with $q = n$ in the n^{th} row

f traverses this list in the order for $m = 2, 3, 4, \dots$ visiting all p/q with $p + q = m$ (and listing only new rationals)



Finite strings

Theorem

The set Σ^ of all finite strings over a finite alphabet Σ is countably infinite*

Finite strings

Theorem

The set Σ^ of all finite strings over a finite alphabet Σ is countably infinite*

Proof.

- First define an (alphabetical) ordering on the symbols in Σ
Show that the strings can be listed in a sequence
 - First single string ϵ of length 0
 - Then all strings of length 1 in lexicographic order
 - Then all strings of length 2 in lexicographic order
 - etc

Finite strings

Theorem

The set Σ^ of all finite strings over a finite alphabet Σ is countably infinite*

Proof.

- First define an (alphabetical) ordering on the symbols in Σ
Show that the strings can be listed in a sequence
 - ▶ First single string ϵ of length 0
 - ▶ Then all strings of length 1 in lexicographic order
 - ▶ Then all strings of length 2 in lexicographic order
 - ▶ etc
- This implies a bijection from \mathbb{N} to Σ^*



Finite strings

Theorem

The set Σ^ of all finite strings over a finite alphabet Σ is countably infinite*

Proof.

- First define an (alphabetical) ordering on the symbols in Σ
Show that the strings can be listed in a sequence
 - ▶ First single string ϵ of length 0
 - ▶ Then all strings of length 1 in lexicographic order
 - ▶ Then all strings of length 2 in lexicographic order
 - ▶ etc
- This implies a bijection from \mathbb{N} to Σ^*



The set of Java-programs is countable; a program is just a finite string

Infinite binary strings

- An infinite length string of bits 10010...

Infinite binary strings

- An infinite length string of bits 10010...
- Such a string is a function $d : \mathbb{N} \rightarrow \{0, 1\}$

Infinite binary strings

- An infinite length string of bits 10010...
- Such a string is a function $d : \mathbb{N} \rightarrow \{0, 1\}$
- with the property $d_m = d(m)$ is the m th symbol (starting from 0)

Uncountable sets

Theorem

The set of infinite binary strings is uncountable

Uncountable sets

Theorem

The set of infinite binary strings is uncountable

Proof.

Let X be the set of infinite binary strings. For a contradiction assume that a bijection $f : \mathbb{N} \rightarrow X$ exists. So, f must be onto (surjective).

Assume that $f(i) = d^i$ for $i \in \mathbb{N}$. So, $X = \{d^0, d^1, \dots, d^m, \dots\}$. Define the infinite binary string d as follows: $d_n = (d_n^0 + 1) \bmod 2$. But for each m , $d \neq d^m$ because $d_m \neq d_m^m$. So, f is not a surjection. □

Uncountable sets

Theorem

The set of infinite binary strings is uncountable

Proof.

Let X be the set of infinite binary strings. For a contradiction assume that a bijection $f : \mathbb{N} \rightarrow X$ exists. So, f must be onto (surjective).

Assume that $f(i) = d^i$ for $i \in \mathbb{N}$. So, $X = \{d^0, d^1, \dots, d^m, \dots\}$. Define the infinite binary string d as follows: $d_n = (d_n^m + 1) \bmod 2$. But for each m , $d \neq d^m$ because $d_m \neq d_m^m$. So, f is not a surjection. □

The technique used here is called diagonalization

Uncountable sets

Theorem

The set of infinite binary strings is uncountable

Proof.

Let X be the set of infinite binary strings. For a contradiction assume that a bijection $f : \mathbb{N} \rightarrow X$ exists. So, f must be onto (surjective).

Assume that $f(i) = d^i$ for $i \in \mathbb{N}$. So, $X = \{d^0, d^1, \dots, d^m, \dots\}$. Define the infinite binary string d as follows: $d_n = (d_n^n + 1) \bmod 2$. But for each m , $d \neq d^m$ because $d_m \neq d_m^m$. So, f is not a surjection. □

The technique used here is called diagonalization

Uncountable sets

Theorem

The set of infinite binary strings is uncountable

Proof.

Let X be the set of infinite binary strings. For a contradiction assume that a bijection $f : \mathbb{N} \rightarrow X$ exists. So, f must be onto (surjective).

Assume that $f(i) = d^i$ for $i \in \mathbb{N}$. So, $X = \{d^0, d^1, \dots, d^m, \dots\}$. Define the infinite binary string d as follows: $d_n = (d_n^n + 1) \bmod 2$. But for each m , $d \neq d^m$ because $d_m \neq d_m^m$. So, f is not a surjection. □

The technique used here is called diagonalization

Similar argument shows that \mathbb{R} via $[0, 1]$ is uncountable using infinite decimal strings (see book).

Uncountable sets

Theorem

The set of infinite binary strings is uncountable

Proof.

Let X be the set of infinite binary strings. For a contradiction assume that a bijection $f : \mathbb{N} \rightarrow X$ exists. So, f must be onto (surjective).

Assume that $f(i) = d^i$ for $i \in \mathbb{N}$. So, $X = \{d^0, d^1, \dots, d^m, \dots\}$. Define the infinite binary string d as follows: $d_n = (d_n^n + 1) \bmod 2$. But for each m , $d \neq d^m$ because $d_m \neq d_m^m$. So, f is not a surjection. □

The technique used here is called diagonalization

Similar argument shows that \mathbb{R} via $[0, 1]$ is uncountable using infinite decimal strings (see book). “Most functions” are not computable!

Schröder-Bernstein Theorem

Theorem

If $|A| \leq |B|$ and $|B| \leq |A|$ then $|A| = |B|$

Schröder-Bernstein Theorem

Theorem

If $|A| \leq |B|$ and $|B| \leq |A|$ then $|A| = |B|$

- Example $|(0, 1)| = |(0, 1]|$

Schröder-Bernstein Theorem

Theorem

If $|A| \leq |B|$ and $|B| \leq |A|$ then $|A| = |B|$

- Example $|(0, 1)| = |(0, 1]|$
- $|(0, 1)| \leq |(0, 1]|$ using identity function

Schröder-Bernstein Theorem

Theorem

If $|A| \leq |B|$ and $|B| \leq |A|$ then $|A| = |B|$

- Example $|(0, 1)| = |(0, 1]|$
- $|(0, 1)| \leq |(0, 1]|$ using identity function
- $|(0, 1]| \leq |(0, 1)|$ use $f(x) = x/2$ as $(0, 1/2] \subset (0, 1)$

Cantor's theorem

Theorem

$$|A| < |\mathcal{P}(A)|$$

Cantor's theorem

Theorem

$$|A| < |\mathcal{P}(A)|$$

Proof.

Consider the injection $f : A \rightarrow \mathcal{P}(A)$ with $f(a) = \{a\}$ for any $a \in A$. Therefore, $|A| \leq |\mathcal{P}(A)|$. Next we show there is not a surjection $f : A \rightarrow \mathcal{P}(A)$. For a contradiction, assume that a surjection f exists. We define the set $B \subseteq A$: $B = \{x \in A \mid x \notin f(x)\}$. Since f is a surjection, there must exist an $a \in A$ s.t. $B = f(a)$. Now there are two cases:

- ① If $a \in B$ then, by definition of B , $a \notin f(a) = B$. Contradiction
- ② If $a \notin B$ then $a \notin f(a)$. Thus, by definition of B , $a \in B$. Contradiction



Implications of Cantor's theorem

- $\mathcal{P}(\mathbb{N})$ is not countable (in fact, $|\mathcal{P}(\mathbb{N})| = |\mathbb{R}|$)

Implications of Cantor's theorem

- $\mathcal{P}(\mathbb{N})$ is not countable (in fact, $|\mathcal{P}(\mathbb{N})| = |\mathbb{R}|$)
- The Continuum Hypothesis claims there is no set S with $|\mathbb{N}| < |S| < |\mathbb{R}|$

Implications of Cantor's theorem

- $\mathcal{P}(\mathbb{N})$ is not countable (in fact, $|\mathcal{P}(\mathbb{N})| = |\mathbb{R}|$)
- The Continuum Hypothesis claims there is no set S with $|\mathbb{N}| < |S| < |\mathbb{R}|$
- It was 1st of Hilbert's 23 open problems presented in 1900.
Shown to be independent of ZF set theory by Gödel/Cohen in 1963: cannot be proven/disproven in ZF

Implications of Cantor's theorem

- $\mathcal{P}(\mathbb{N})$ is not countable (in fact, $|\mathcal{P}(\mathbb{N})| = |\mathbb{R}|$)
- The Continuum Hypothesis claims there is no set S with $|\mathbb{N}| < |S| < |\mathbb{R}|$
- It was 1st of Hilbert's 23 open problems presented in 1900.
Shown to be independent of ZF set theory by Gödel/Cohen in 1963: cannot be proven/disproven in ZF
- There exists an infinite hierarchy of sets of ever larger cardinality

Implications of Cantor's theorem

- $\mathcal{P}(\mathbb{N})$ is not countable (in fact, $|\mathcal{P}(\mathbb{N})| = |\mathbb{R}|$)
- The Continuum Hypothesis claims there is no set S with $|\mathbb{N}| < |S| < |\mathbb{R}|$
- It was 1st of Hilbert's 23 open problems presented in 1900.
Shown to be independent of ZF set theory by Gödel/Cohen in 1963: cannot be proven/disproven in ZF
- There exists an infinite hierarchy of sets of ever larger cardinality
- $S_0 := \mathbb{N}$ and $S_{i+1} := \mathcal{P}(S_i)$

Implications of Cantor's theorem

- $\mathcal{P}(\mathbb{N})$ is not countable (in fact, $|\mathcal{P}(\mathbb{N})| = |\mathbb{R}|$)
- The Continuum Hypothesis claims there is no set S with $|\mathbb{N}| < |S| < |\mathbb{R}|$
- It was 1st of Hilbert's 23 open problems presented in 1900.
Shown to be independent of ZF set theory by Gödel/Cohen in 1963: cannot be proven/disproven in ZF
- There exists an infinite hierarchy of sets of ever larger cardinality
- $S_0 := \mathbb{N}$ and $S_{i+1} := \mathcal{P}(S_i)$
- $|S_0| < |S_1| < \dots < |S_i| < |S_{i+1}| < \dots$

Discrete Mathematics & Mathematical Reasoning

Induction

Colin Stirling

Informatics

Another proof method: Mathematical Induction

- Want to prove $\forall n \in \mathbb{N} (P(n))$

Another proof method: Mathematical Induction

- Want to prove $\forall n \in \mathbb{N} (P(n))$
- BASIS STEP show $P(0)$

Another proof method: Mathematical Induction

- Want to prove $\forall n \in \mathbb{N} (P(n))$
- BASIS STEP show $P(0)$
- INDUCTIVE STEP show $P(k) \rightarrow P(k + 1)$ for all $k \in \mathbb{N}$

Another proof method: Mathematical Induction

- Want to prove $\forall n \in \mathbb{N} (P(n))$
- BASIS STEP show $P(0)$
- INDUCTIVE STEP show $P(k) \rightarrow P(k + 1)$ for all $k \in \mathbb{N}$
- Assume k is arbitrary and $P(k)$ is true. Show $P(k + 1)$

Another proof method: Mathematical Induction

- Want to prove $\forall n \in \mathbb{Z}^+ (P(n))$
- BASIS STEP show $P(1)$
- INDUCTIVE STEP show $P(k) \rightarrow P(k + 1)$ for all $k \in \mathbb{Z}^+$
- Assume k is arbitrary and $P(k)$ is true. Show $P(k + 1)$

Another proof method: Mathematical Induction

- Want to prove $\forall n \geq m \in \mathbb{N} (P(n))$
- BASIS STEP show $P(m)$
- INDUCTIVE STEP show $P(k) \rightarrow P(k + 1)$ for all $k \geq m \in \mathbb{N}$
- Assume $k \geq m$ is arbitrary and $P(k)$ is true. Show $P(k + 1)$

Another proof method: Mathematical Induction

- Want to prove $\forall n \in \mathbb{Q}^+ (P(n))$

Another proof method: Mathematical Induction

- Want to prove $\forall n \in \mathbb{Q}^+ (P(n))$
- Can we use induction?

Another proof method: Mathematical Induction

- Want to prove $\forall n \in \mathbb{Q}^+ (P(n))$
- Can we use induction?
- Want to prove $\forall x \in \mathbb{R}^+ (P(x))$

Another proof method: Mathematical Induction

- Want to prove $\forall n \in \mathbb{Q}^+ (P(n))$
- Can we use induction?
- Want to prove $\forall x \in \mathbb{R}^+ (P(x))$
- Can we use induction?

Another proof method: Mathematical Induction

- Want to prove $\forall n \in \mathbb{Q}^+ (P(n))$
- Can we use induction?
- Want to prove $\forall x \in \mathbb{R}^+ (P(x))$
- Can we use induction?
- What justifies mathematical induction?

Another proof method: Mathematical Induction

- Want to prove $\forall n \in \mathbb{Q}^+ (P(n))$
- Can we use induction?
- Want to prove $\forall x \in \mathbb{R}^+ (P(x))$
- Can we use induction?
- What justifies mathematical induction?
- Well ordering principle: every nonempty set $S \subseteq \mathbb{N}$ has a least element

Examples

- $\sum_{j=1}^n j = \frac{n(n+1)}{2}$

Examples

- $\sum_{j=1}^n j = \frac{n(n+1)}{2}$
- $\sum_{j=0}^k ar^j = \frac{ar^{k+1}-a}{r-1}$ when $r \neq 1$

Examples

- $\sum_{j=1}^n j = \frac{n(n+1)}{2}$
- $\sum_{j=0}^k ar^j = \frac{ar^{k+1}-a}{r-1}$ when $r \neq 1$
- for all $n \in \mathbb{Z}^+$ ($n < 2^n$)

Examples

- $\sum_{j=1}^n j = \frac{n(n+1)}{2}$
- $\sum_{j=0}^k ar^j = \frac{ar^{k+1}-a}{r-1}$ when $r \neq 1$
- for all $n \in \mathbb{Z}^+$ ($n < 2^n$)
- for all integers $n \geq 4$, $2^n < n!$

Examples

- $\sum_{j=1}^n j = \frac{n(n+1)}{2}$
- $\sum_{j=0}^k ar^j = \frac{ar^{k+1}-a}{r-1}$ when $r \neq 1$
- for all $n \in \mathbb{Z}^+ (n < 2^n)$
- for all integers $n \geq 4$, $2^n < n!$
- for all $n > 1 (n^3 - n)$ is divisible by 3

Examples

- $\sum_{j=1}^n j = \frac{n(n+1)}{2}$
- $\sum_{j=0}^k ar^j = \frac{ar^{k+1}-a}{r-1}$ when $r \neq 1$
- for all $n \in \mathbb{Z}^+ (n < 2^n)$
- for all integers $n \geq 4, 2^n < n!$
- for all $n > 1 (n^3 - n)$ is divisible by 3
- for all $n \in \mathbb{N} (7^{n+2} + 8^{2n+1})$ is divisible by 57

More examples

- **Odd Pie Fights** An odd number of people stand in a room at mutually distinct distances. At the same time each person throws a pie at their nearest neighbour and hits them. Prove that at least one person is not hit by a pie

More examples

- **Odd Pie Fights** An odd number of people stand in a room at mutually distinct distances. At the same time each person throws a pie at their nearest neighbour and hits them. Prove that at least one person is not hit by a pie
- All horses have the same colour

Strong Induction

- Want to prove $\forall n \in \mathbb{N} (P(n))$

Strong Induction

- Want to prove $\forall n \in \mathbb{N} (P(n))$
- BASIS STEP show $P(0)$

Strong Induction

- Want to prove $\forall n \in \mathbb{N} (P(n))$
- BASIS STEP show $P(0)$
- INDUCTIVE STEP show $(P(0) \wedge \dots \wedge P(k)) \rightarrow P(k + 1)$ for all $k \in \mathbb{N}$

Strong Induction

- Want to prove $\forall n \in \mathbb{N} (P(n))$
- BASIS STEP show $P(0)$
- INDUCTIVE STEP show $(P(0) \wedge \dots \wedge P(k)) \rightarrow P(k + 1)$ for all $k \in \mathbb{N}$
- Assume k is arbitrary and $P(0), \dots, P(k)$ are true. Show $P(k + 1)$

Strong Induction

- Want to prove $\forall n \in \mathbb{Z}^+ (P(n))$
- BASIS STEP show $P(1)$
- INDUCTIVE STEP show $(P(1) \wedge \dots \wedge P(k)) \rightarrow P(k + 1)$ for all $k \in \mathbb{Z}^+$
- Assume k is arbitrary and $P(1), \dots, P(k)$ are true. Show $P(k + 1)$

Strong Induction

- Want to prove $\forall n \geq m \in \mathbb{N} (P(n))$
- BASIS STEP** show $P(m)$
- INDUCTIVE STEP** show $(P(m) \wedge \dots \wedge P(k)) \rightarrow P(k + 1)$ for all $k \geq m \in \mathbb{N}$
- Assume $k \geq m$ is arbitrary and $P(m), \dots, P(k)$ are true. Show $P(k + 1)$

Examples

- If $n > 1$ is an integer, then n can be written as a product of primes

Examples

- If $n > 1$ is an integer, then n can be written as a product of primes
- Prove that every amount of postage of 12p or more can be formed using just 4p and 5p stamps

Examples

- If $n > 1$ is an integer, then n can be written as a product of primes
- Prove that every amount of postage of 12p or more can be formed using just 4p and 5p stamps
- Game of matches Two players take turns removing any positive number of matches they want from one of two piles of matches. The player who removes the last match wins the game. Show that if the two piles contain the same number of matches initially then the second player can guarantee a win

Discrete Mathematics & Mathematical Reasoning

Arithmetic Modulo m

Colin Stirling

Informatics

Some slides based on ones by Myrto Arapinis

Division

Definition

If a and b are integers with $a \neq 0$, then a divides b , written $a|b$, if there exists an integer c such that $b = ac$

Division

Definition

If a and b are integers with $a \neq 0$, then a divides b , written $a|b$, if there exists an integer c such that $b = ac$

- $3 | (-12)$ $3 | 0$ $3 \nmid 7$ (where \nmid “not divides”)

Division

Definition

If a and b are integers with $a \neq 0$, then a divides b , written $a|b$, if there exists an integer c such that $b = ac$

- $3 | (-12)$ $3 | 0$ $3 \nmid 7$ (where \nmid “not divides”)
- If $a|b$ and $a|c$, then $a|(b + c)$

Division

Definition

If a and b are integers with $a \neq 0$, then a divides b , written $a|b$, if there exists an integer c such that $b = ac$

- $3 | (-12)$ $3 | 0$ $3 \nmid 7$ (where \nmid “not divides”)
- If $a|b$ and $a|c$, then $a|(b + c)$
- Proof $a|b \Leftrightarrow \exists k_b. b = k_b a$ and $a|c \Leftrightarrow \exists k_c. c = k_c a$. But then $b + c = (k_b + k_c)a$ which by definition implies that $a|(b + c)$

Division

Definition

If a and b are integers with $a \neq 0$, then a divides b , written $a|b$, if there exists an integer c such that $b = ac$

- $3 | (-12) \quad 3 | 0 \quad 3 \nmid 7$ (where \nmid “not divides”)
- If $a|b$ and $a|c$, then $a|(b + c)$
- Proof $a|b \Leftrightarrow \exists k_b. b = k_b a$ and $a|c \Leftrightarrow \exists k_c. c = k_c a$. But then $b + c = (k_b + k_c)a$ which by definition implies that $a|(b + c)$
- If $a|b$, then $a|bc$

Division

Definition

If a and b are integers with $a \neq 0$, then a divides b , written $a|b$, if there exists an integer c such that $b = ac$

- $3 | (-12) \quad 3 | 0 \quad 3 \nmid 7$ (where \nmid “not divides”)
- If $a|b$ and $a|c$, then $a|(b + c)$
- Proof $a|b \Leftrightarrow \exists k_b. b = k_b a$ and $a|c \Leftrightarrow \exists k_c. c = k_c a$. But then $b + c = (k_b + k_c)a$ which by definition implies that $a|(b + c)$
- If $a|b$, then $a|bc$
- Proof $a|b \Leftrightarrow \exists k_b. b = k_b a$. But then $bc = k_b ac$ which by definition implies that $a|bc$

Division

Definition

If a and b are integers with $a \neq 0$, then a divides b , written $a|b$, if there exists an integer c such that $b = ac$

- $3 | (-12) \quad 3 | 0 \quad 3 \nmid 7$ (where \nmid “not divides”)
- If $a|b$ and $a|c$, then $a|(b + c)$
- Proof $a|b \Leftrightarrow \exists k_b. b = k_b a$ and $a|c \Leftrightarrow \exists k_c. c = k_c a$. But then $b + c = (k_b + k_c)a$ which by definition implies that $a|(b + c)$
- If $a|b$, then $a|bc$
- Proof $a|b \Leftrightarrow \exists k_b. b = k_b a$. But then $bc = k_b ac$ which by definition implies that $a|bc$
- If $a|b$ and $b|c$, then $a|c$

Division

Definition

If a and b are integers with $a \neq 0$, then a divides b , written $a|b$, if there exists an integer c such that $b = ac$

- $3 | (-12) \quad 3 | 0 \quad 3 \nmid 7$ (where \nmid “not divides”)
- If $a|b$ and $a|c$, then $a|(b + c)$
- Proof $a|b \Leftrightarrow \exists k_b. b = k_b a$ and $a|c \Leftrightarrow \exists k_c. c = k_c a$. But then $b + c = (k_b + k_c)a$ which by definition implies that $a|(b + c)$
- If $a|b$, then $a|bc$
- Proof $a|b \Leftrightarrow \exists k_b. b = k_b a$. But then $bc = k_b ac$ which by definition implies that $a|bc$
- If $a|b$ and $b|c$, then $a|c$
- Proof $a|b \Leftrightarrow \exists k_b. b = k_b a$ and $b|c \Leftrightarrow \exists k_c. c = k_c b$. But then $c = (k_c k_b)a$ which by definition implies that $a|c$

Division algorithm (not really an algorithm!)

Theorem

If a is an integer and d a positive integer, then there are unique integers q and r , with $0 \leq r < d$, such that $a = dq + r$

Division algorithm (not really an algorithm!)

Theorem

If a is an integer and d a positive integer, then there are unique integers q and r , with $0 \leq r < d$, such that $a = dq + r$

q is quotient and r the remainder; $q = a \text{ div } d$ and $r = a \text{ mod } d$

Division algorithm (not really an algorithm!)

Theorem

If a is an integer and d a positive integer, then there are unique integers q and r , with $0 \leq r < d$, such that $a = dq + r$

q is quotient and r the remainder; $q = a \text{ div } d$ and $r = a \text{ mod } d$

Proof.

Consider the largest q such that $dq \leq a$; then $a = dq + r$ for $0 \leq r < d$: if $r \geq d$ then $d(q + 1) \leq a$ which contradicts that q is largest. So, there is at least one such q and r . Assume that there is more than one:

$a = dq_1 + r_1$, $a = dq_2 + r_2$, and $(q_1, r_1) \neq (q_2, r_2)$. If $q_1 = q_2$ then $r_1 = a - dq_1 = a - dq_2 = r_2$. Since $dq_1 + r_1 = dq_2 + r_2$, $d = \frac{r_1 - r_2}{q_2 - q_1}$ which is impossible because $r_1 - r_2 < d$.



Congruent modulo m relation

Definition

If a and b are integers and m is a positive integer, then a is congruent to b modulo m , written $a \equiv b \pmod{m}$, iff $m|(a - b)$

- $17 \equiv 5 \pmod{6}$ because 6 divides $17 - 5 = 12$

Congruent modulo m relation

Definition

If a and b are integers and m is a positive integer, then a is congruent to b modulo m , written $a \equiv b \pmod{m}$, iff $m|(a - b)$

- $17 \equiv 5 \pmod{6}$ because 6 divides $17 - 5 = 12$
- $-17 \not\equiv 5 \pmod{6}$ because 6 $\nmid (-22)$

Congruent modulo m relation

Definition

If a and b are integers and m is a positive integer, then a is congruent to b modulo m , written $a \equiv b \pmod{m}$, iff $m|(a - b)$

- $17 \equiv 5 \pmod{6}$ because 6 divides $17 - 5 = 12$
- $-17 \not\equiv 5 \pmod{6}$ because 6 $\nmid (-22)$
- $-17 \equiv 1 \pmod{6}$

Congruent modulo m relation

Definition

If a and b are integers and m is a positive integer, then a is congruent to b modulo m , written $a \equiv b \pmod{m}$, iff $m|(a - b)$

- $17 \equiv 5 \pmod{6}$ because 6 divides $17 - 5 = 12$
- $-17 \not\equiv 5 \pmod{6}$ because 6 $\nmid (-22)$
- $-17 \equiv 1 \pmod{6}$
- $24 \not\equiv 14 \pmod{6}$ because 6 $\nmid 10$

Congruence is an equivalence relation

Theorem

$a \equiv b \pmod{m}$ iff $a \bmod m = b \bmod m$

Congruence is an equivalence relation

Theorem

$$a \equiv b \pmod{m} \text{ iff } a \bmod m = b \bmod m$$

Proof.

Assume $a \equiv b \pmod{m}$; so $m|(a - b)$. If $a = q_1m + r_1$ and $b = q_2m + r_2$ where $0 \leq r_1 < m$ and $0 \leq r_2 < m$ it follows that $r_1 = r_2$ and so $a \bmod m = b \bmod m$. If $a \bmod m = b \bmod m$ then a and b have the same remainder so $a = q_1m + r$ and $b = q_2m + r$; therefore $a - b = (q_1 - q_2)m$, and so $m|(a - b)$.



Congruence is an equivalence relation

Theorem

$$a \equiv b \pmod{m} \text{ iff } a \bmod m = b \bmod m$$

Proof.

Assume $a \equiv b \pmod{m}$; so $m|(a - b)$. If $a = q_1m + r_1$ and $b = q_2m + r_2$ where $0 \leq r_1 < m$ and $0 \leq r_2 < m$ it follows that $r_1 = r_2$ and so $a \bmod m = b \bmod m$. If $a \bmod m = b \bmod m$ then a and b have the same remainder so $a = q_1m + r$ and $b = q_2m + r$; therefore $a - b = (q_1 - q_2)m$, and so $m|(a - b)$.

□

- $\equiv \pmod{m}$ is an equivalence relation on integers

A simple theorem of congruence

Theorem

$a \equiv b \pmod{m}$ iff there is an integer k such that $a = b + km$

A simple theorem of congruence

Theorem

$a \equiv b \pmod{m}$ iff there is an integer k such that $a = b + km$

Proof.

If $a \equiv b \pmod{m}$, then by the definition of congruence $m|(a - b)$. Hence, there is an integer k such that $a - b = km$ and equivalently $a = b + km$. If there is an integer k such that $a = b + km$, then $km = a - b$. Hence, $m|(a - b)$ and $a \equiv b \pmod{m}$. □

Congruences of sums, differences, and products

Theorem

If $a \equiv b \pmod{m}$ and $c \equiv d \pmod{m}$, then $a + c \equiv b + d \pmod{m}$,
 $a - c \equiv b - d \pmod{m}$, and $ac \equiv bd \pmod{m}$

Congruences of sums, differences, and products

Theorem

If $a \equiv b \pmod{m}$ and $c \equiv d \pmod{m}$, then $a + c \equiv b + d \pmod{m}$,
 $a - c \equiv b - d \pmod{m}$, and $ac \equiv bd \pmod{m}$

Proof.

Since $a \equiv b \pmod{m}$ and $c \equiv d \pmod{m}$, by the previous theorem,
there are integers s and t with $b = a + sm$ and $d = c + tm$. Therefore,
 $b + d = (a + sm) + (c + tm) = (a + c) + m(s + t)$, and
 $bd = (a + sm)(c + tm) = ac + m(at + cs + stm)$. Hence,
 $a + c \equiv b + d \pmod{m}$ and $ac \equiv bd \pmod{m}$



Congruences of sums, differences, and products

Theorem

If $a \equiv b \pmod{m}$ and $c \equiv d \pmod{m}$, then $a + c \equiv b + d \pmod{m}$,
 $a - c \equiv b - d \pmod{m}$, and $ac \equiv bd \pmod{m}$

Proof.

Since $a \equiv b \pmod{m}$ and $c \equiv d \pmod{m}$, by the previous theorem,
there are integers s and t with $b = a + sm$ and $d = c + tm$. Therefore,
 $b + d = (a + sm) + (c + tm) = (a + c) + m(s + t)$, and
 $bd = (a + sm)(c + tm) = ac + m(at + cs + stm)$. Hence,
 $a + c \equiv b + d \pmod{m}$ and $ac \equiv bd \pmod{m}$ □

Corollary

- $(a + b) \pmod{m} = ((a \pmod{m}) + (b \pmod{m})) \pmod{m}$
- $ab \pmod{m} = ((a \pmod{m})(b \pmod{m})) \pmod{m}$

Arithmetic modulo m

- $\mathbb{Z}_m = \{0, 1, \dots, m - 1\}$

Arithmetic modulo m

- $\mathbb{Z}_m = \{0, 1, \dots, m - 1\}$
- $+_m$ on \mathbb{Z}_m is $a +_m b = (a + b) \bmod m$

Arithmetic modulo m

- $\mathbb{Z}_m = \{0, 1, \dots, m - 1\}$
- $+_m$ on \mathbb{Z}_m is $a +_m b = (a + b) \bmod m$
- \cdot_m on \mathbb{Z}_m is define $a \cdot_m b = (a \cdot b) \bmod m$

Arithmetic modulo m

- $\mathbb{Z}_m = \{0, 1, \dots, m - 1\}$
- $+_m$ on \mathbb{Z}_m is $a +_m b = (a + b) \text{ mod } m$
- \cdot_m on \mathbb{Z}_m is define $a \cdot_m b = (a \cdot b) \text{ mod } m$
- Find $7 +_{11} 9$ and $-7 \cdot_{11} 9$

Arithmetic modulo m

- $\mathbb{Z}_m = \{0, 1, \dots, m - 1\}$
- $+_m$ on \mathbb{Z}_m is $a +_m b = (a + b) \text{ mod } m$
- \cdot_m on \mathbb{Z}_m is define $a \cdot_m b = (a \cdot b) \text{ mod } m$
- Find $7 +_{11} 9$ and $-7 \cdot_{11} 9$
- $7 +_{11} 9 = (7 + 9) \text{ mod } 11 = 16 \text{ mod } 11 = 5$

Arithmetic modulo m

- $\mathbb{Z}_m = \{0, 1, \dots, m - 1\}$
- $+_m$ on \mathbb{Z}_m is $a +_m b = (a + b) \bmod m$
- \cdot_m on \mathbb{Z}_m is define $a \cdot_m b = (a \cdot b) \bmod m$
- Find $7 +_{11} 9$ and $-7 \cdot_{11} 9$
- $7 +_{11} 9 = (7 + 9) \bmod 11 = 16 \bmod 11 = 5$
- $-7 \cdot_{11} 9 = (-7 \cdot 9) \bmod 11 = -63 \bmod 11 = 3$

Arithmetic modulo m

The operations $+_m$ and \cdot_m satisfy many of the same properties as ordinary addition and multiplication

Arithmetic modulo m

The operations $+_m$ and \cdot_m satisfy many of the same properties as ordinary addition and multiplication

Closure If $a, b \in \mathbb{Z}_m$, then $a +_m b$ and $a \cdot_m b$ belong to \mathbb{Z}_m

Arithmetic modulo m

The operations $+_m$ and \cdot_m satisfy many of the same properties as ordinary addition and multiplication

Closure If $a, b \in \mathbb{Z}_m$, then $a +_m b$ and $a \cdot_m b$ belong to \mathbb{Z}_m

Associativity If $a, b, c \in \mathbb{Z}_m$, then $(a +_m b) +_m c = a +_m (b +_m c)$ and $(a \cdot_m b) \cdot_m c = a \cdot_m (b \cdot_m c)$

Arithmetic modulo m

The operations $+_m$ and \cdot_m satisfy many of the same properties as ordinary addition and multiplication

Closure If $a, b \in \mathbb{Z}_m$, then $a +_m b$ and $a \cdot_m b$ belong to \mathbb{Z}_m

Associativity If $a, b, c \in \mathbb{Z}_m$, then $(a +_m b) +_m c = a +_m (b +_m c)$ and $(a \cdot_m b) \cdot_m c = a \cdot_m (b \cdot_m c)$

Commutativity If $a, b \in \mathbb{Z}_m$, then $a +_m b = b +_m a$ and $a \cdot_m b = b \cdot_m a$

Arithmetic modulo m

The operations $+_m$ and \cdot_m satisfy many of the same properties as ordinary addition and multiplication

Closure If $a, b \in \mathbb{Z}_m$, then $a +_m b$ and $a \cdot_m b$ belong to \mathbb{Z}_m

Associativity If $a, b, c \in \mathbb{Z}_m$, then $(a +_m b) +_m c = a +_m (b +_m c)$ and $(a \cdot_m b) \cdot_m c = a \cdot_m (b \cdot_m c)$

Commutativity If $a, b \in \mathbb{Z}_m$, then $a +_m b = b +_m a$ and $a \cdot_m b = b \cdot_m a$

Identity elements The elements 0 and 1 are identity elements for addition and multiplication modulo m , respectively. If $a \in \mathbb{Z}_m$ then $a +_m 0 = a$ and $a \cdot_m 1 = a$

Arithmetic modulo m

The operations $+_m$ and \cdot_m satisfy many of the same properties as ordinary addition and multiplication

Closure If $a, b \in \mathbb{Z}_m$, then $a +_m b$ and $a \cdot_m b$ belong to \mathbb{Z}_m

Associativity If $a, b, c \in \mathbb{Z}_m$, then $(a +_m b) +_m c = a +_m (b +_m c)$ and $(a \cdot_m b) \cdot_m c = a \cdot_m (b \cdot_m c)$

Commutativity If $a, b \in \mathbb{Z}_m$, then $a +_m b = b +_m a$ and $a \cdot_m b = b \cdot_m a$

Identity elements The elements 0 and 1 are identity elements for addition and multiplication modulo m , respectively. If $a \in \mathbb{Z}_m$ then $a +_m 0 = a$ and $a \cdot_m 1 = a$

Additive inverses If $0 \neq a \in \mathbb{Z}_m$, then $m - a$ is the additive inverse of a modulo m . Moreover, 0 is its own additive inverse $a +_m (m - a) = 0$ and $0 +_m 0 = 0$

Arithmetic modulo m

The operations $+_m$ and \cdot_m satisfy many of the same properties as ordinary addition and multiplication

Closure If $a, b \in \mathbb{Z}_m$, then $a +_m b$ and $a \cdot_m b$ belong to \mathbb{Z}_m

Associativity If $a, b, c \in \mathbb{Z}_m$, then $(a +_m b) +_m c = a +_m (b +_m c)$ and $(a \cdot_m b) \cdot_m c = a \cdot_m (b \cdot_m c)$

Commutativity If $a, b \in \mathbb{Z}_m$, then $a +_m b = b +_m a$ and $a \cdot_m b = b \cdot_m a$

Identity elements The elements 0 and 1 are identity elements for addition and multiplication modulo m , respectively. If $a \in \mathbb{Z}_m$ then $a +_m 0 = a$ and $a \cdot_m 1 = a$

Additive inverses If $0 \neq a \in \mathbb{Z}_m$, then $m - a$ is the additive inverse of a modulo m . Moreover, 0 is its own additive inverse $a +_m (m - a) = 0$ and $0 +_m 0 = 0$

Distributivity If $a, b, c \in \mathbb{Z}_m$, then $a \cdot_m (b +_m c) = (a \cdot_m b) +_m (a \cdot_m c)$ and $(a +_m b) \cdot_m c = (a \cdot_m c) +_m (b \cdot_m c)$

Multiplicative inverses

- Over the reals, dividing by a number x is the same as multiplying by $y = 1/x$, so $xy = 1$

Multiplicative inverses

- Over the reals, dividing by a number x is the same as multiplying by $y = 1/x$, so $xy = 1$
- Similarly for $x \bmod m$, we wish to find $y \bmod m$ such that $xy \equiv 1 \pmod{m}$

Multiplicative inverses

- Over the reals, dividing by a number x is the same as multiplying by $y = 1/x$, so $xy = 1$
- Similarly for $x \bmod m$, we wish to find $y \bmod m$ such that $xy \equiv 1 \pmod{m}$
- $x = 8$ and $m = 15$. Then $2x = 16 \equiv 1 \pmod{15}$, so 2 is a multiplicative inverse of 8 (mod 15)

Multiplicative inverses

- Over the reals, dividing by a number x is the same as multiplying by $y = 1/x$, so $xy = 1$
- Similarly for $x \bmod m$, we wish to find $y \bmod m$ such that $xy \equiv 1 \pmod{m}$
- $x = 8$ and $m = 15$. Then $2x = 16 \equiv 1 \pmod{15}$, so 2 is a multiplicative inverse of 8 (mod 15)
- $x = 12$ and $m = 15$

Multiplicative inverses

- Over the reals, dividing by a number x is the same as multiplying by $y = 1/x$, so $xy = 1$
- Similarly for $x \bmod m$, we wish to find $y \bmod m$ such that $xy \equiv 1 \pmod{m}$
- $x = 8$ and $m = 15$. Then $2x = 16 \equiv 1 \pmod{15}$, so 2 is a multiplicative inverse of 8 (mod 15)
- $x = 12$ and $m = 15$
The sequence $\{ax \pmod{m} \mid a = 0, 1, 2, \dots\}$ is periodic, and takes on the values $\{0, 12, 9, 6, 3\}$. So, 12 has no multiplicative inverse mod 15

Multiplicative inverses

- Over the reals, dividing by a number x is the same as multiplying by $y = 1/x$, so $xy = 1$
- Similarly for $x \bmod m$, we wish to find $y \bmod m$ such that $xy \equiv 1 \pmod{m}$
- $x = 8$ and $m = 15$. Then $2x = 16 \equiv 1 \pmod{15}$, so 2 is a multiplicative inverse of 8 (mod 15)
- $x = 12$ and $m = 15$
The sequence $\{ax \pmod{m} \mid a = 0, 1, 2, \dots\}$ is periodic, and takes on the values $\{0, 12, 9, 6, 3\}$. So, 12 has no multiplicative inverse mod 15

Not all integers have an inverse mod m . Return to this later

Discrete Mathematics & Mathematical Reasoning

Primes and Greatest Common Divisors

Colin Stirling

Informatics

Some slides based on ones by Myrto Arapinis

Primes

Definition

A positive integer $p > 1$ is called prime iff the only positive factors of p are 1 and p . Otherwise it is called composite

Primes

Definition

A positive integer $p > 1$ is called prime iff the only positive factors of p are 1 and p . Otherwise it is called composite

Theorem (Fundamental Theorem of Arithmetic)

Every positive integer greater than 1 can be written uniquely as a prime or as the product of its prime factors, written in order of nondecreasing size

Primes

Definition

A positive integer $p > 1$ is called prime iff the only positive factors of p are 1 and p . Otherwise it is called composite

Theorem (Fundamental Theorem of Arithmetic)

Every positive integer greater than 1 can be written uniquely as a prime or as the product of its prime factors, written in order of nondecreasing size

$$765 = 3 \cdot 3 \cdot 5 \cdot 17 = 3^2 \cdot 5 \cdot 17$$

Proof of fundamental theorem

Theorem (Fundamental Theorem of Arithmetic)

Every positive integer greater than 1 can be written uniquely as a prime or as the product of its prime factors, written in order of nondecreasing size

Proof of fundamental theorem

Theorem (Fundamental Theorem of Arithmetic)

Every positive integer greater than 1 can be written uniquely as a prime or as the product of its prime factors, written in order of nondecreasing size

Showed by induction if $n > 1$ is an integer then n can be written as a product of primes

Proof of fundamental theorem

Theorem (Fundamental Theorem of Arithmetic)

Every positive integer greater than 1 can be written uniquely as a prime or as the product of its prime factors, written in order of nondecreasing size

Showed by induction if $n > 1$ is an integer then n can be written as a product of primes

Missing is uniqueness

Proof of fundamental theorem

Theorem (Fundamental Theorem of Arithmetic)

Every positive integer greater than 1 can be written uniquely as a prime or as the product of its prime factors, written in order of nondecreasing size

Showed by induction if $n > 1$ is an integer then n can be written as a product of primes

Missing is uniqueness

Lemma if p is prime and $p|a_1 a_2 \dots a_n$ where each a_i is an integer, then $p|a_j$ for some $1 \leq j \leq n$

Proof of fundamental theorem

Theorem (Fundamental Theorem of Arithmetic)

Every positive integer greater than 1 can be written uniquely as a prime or as the product of its prime factors, written in order of nondecreasing size

Showed by induction if $n > 1$ is an integer then n can be written as a product of primes

Missing is uniqueness

Lemma if p is prime and $p|a_1 a_2 \dots a_n$ where each a_i is an integer, then $p|a_j$ for some $1 \leq j \leq n$

By induction too

Proof of fundamental theorem

Theorem (Fundamental Theorem of Arithmetic)

Every positive integer greater than 1 can be written uniquely as a prime or as the product of its prime factors, written in order of nondecreasing size

Showed by induction if $n > 1$ is an integer then n can be written as a product of primes

Missing is uniqueness

Lemma if p is prime and $p|a_1 a_2 \dots a_n$ where each a_i is an integer, then $p|a_j$ for some $1 \leq j \leq n$

By induction too

Now result follows

There are infinitely many primes

There are infinitely many primes

Lemma Every natural number greater than one is either prime or it has a prime divisor

There are infinitely many primes

Lemma Every natural number greater than one is either prime or it has a prime divisor

Follows from fundamental theorem

There are infinitely many primes

Lemma Every natural number greater than one is either prime or it has a prime divisor

Follows from fundamental theorem

Proof Suppose towards a contradiction that there are only finitely many primes $p_1, p_2, p_3, \dots, p_k$. Consider the number $q = p_1 p_2 p_3 \dots p_k + 1$, the product of all the primes plus one. By hypothesis q cannot be prime because it is strictly larger than all the primes. Thus, by the lemma, it has a prime divisor, p . Because $p_1, p_2, p_3, \dots, p_k$ are all the primes, p must be equal to one of them, so p is a divisor of their product. So we have that p divides $p_1 p_2 p_3 \dots p_k$, and p divides q , but that means p divides their difference, which is 1. Therefore $p \leq 1$. Contradiction. Therefore there are infinitely many primes.

The Sieve of Eratosthenes

How to find all primes between 2 and n ?

The Sieve of Eratosthenes

How to find all primes between 2 and n ?

A **very inefficient** method of determining if a number n is prime

Try every integer $i \leq \sqrt{n}$ and see if n is divisible by i

- ① Write the numbers $2, \dots, n$ into a list. Let $i := 2$

The Sieve of Eratosthenes

How to find all primes between 2 and n ?

A **very inefficient** method of determining if a number n is prime

Try every integer $i \leq \sqrt{n}$ and see if n is divisible by i

- ① Write the numbers $2, \dots, n$ into a list. Let $i := 2$
- ② Remove all strict multiples of i from the list

The Sieve of Eratosthenes

How to find all primes between 2 and n ?

A **very inefficient** method of determining if a number n is prime

Try every integer $i \leq \sqrt{n}$ and see if n is divisible by i

- ① Write the numbers $2, \dots, n$ into a list. Let $i := 2$
- ② Remove all strict multiples of i from the list
- ③ Let k be the smallest number present in the list s.t. $k > i$ and let $i := k$

The Sieve of Eratosthenes

How to find all primes between 2 and n ?

A **very inefficient** method of determining if a number n is prime

Try every integer $i \leq \sqrt{n}$ and see if n is divisible by i

- ① Write the numbers $2, \dots, n$ into a list. Let $i := 2$
- ② Remove all strict multiples of i from the list
- ③ Let k be the smallest number present in the list s.t. $k > i$ and let $i := k$
- ④ If $i > \sqrt{n}$ then stop else go to step 2

The Sieve of Eratosthenes

How to find all primes between 2 and n ?

A **very inefficient** method of determining if a number n is prime

Try every integer $i \leq \sqrt{n}$ and see if n is divisible by i

- ① Write the numbers $2, \dots, n$ into a list. Let $i := 2$
- ② Remove all strict multiples of i from the list
- ③ Let k be the smallest number present in the list s.t. $k > i$ and let $i := k$
- ④ If $i > \sqrt{n}$ then stop else go to step 2

Testing if a number is prime can be done efficiently in polynomial time [Agrawal-Kayal-Saxena 2002], i.e., polynomial in the number of bits used to describe the input number. Efficient randomized tests had been available previously.

Greatest common divisor

Definition

Let $a, b \in \mathbb{Z}^+$. The largest integer d such that $d|a$ and $d|b$ is called the greatest common divisor of a and b , written $\gcd(a, b)$

Greatest common divisor

Definition

Let $a, b \in \mathbb{Z}^+$. The largest integer d such that $d|a$ and $d|b$ is called the greatest common divisor of a and b , written $\gcd(a, b)$

$$\gcd(24, 36) = 12$$

Greatest common divisor

Definition

Let $a, b \in \mathbb{Z}^+$. The largest integer d such that $d|a$ and $d|b$ is called the greatest common divisor of a and b , written $\gcd(a, b)$

$$\gcd(24, 36) = 12$$

Definition

The integers a and b are relatively prime (coprime) iff $\gcd(a, b) = 1$

Greatest common divisor

Definition

Let $a, b \in \mathbb{Z}^+$. The largest integer d such that $d|a$ and $d|b$ is called the greatest common divisor of a and b , written $\gcd(a, b)$

$$\gcd(24, 36) = 12$$

Definition

The integers a and b are relatively prime (coprime) iff $\gcd(a, b) = 1$

9 and 22 are coprime (both are composite)

Gcd by prime factorisations

Suppose that the prime factorisations of a and b are

$$a = p_1^{a_1} p_2^{a_2} \cdots p_n^{a_n}$$

$$b = p_1^{b_1} p_2^{b_2} \cdots p_n^{b_n}$$

where each exponent is a nonnegative integer (possibly zero)

Gcd by prime factorisations

Suppose that the prime factorisations of a and b are

$$a = p_1^{a_1} p_2^{a_2} \cdots p_n^{a_n} \quad b = p_1^{b_1} p_2^{b_2} \cdots p_n^{b_n}$$

where each exponent is a nonnegative integer (possibly zero)

$$\gcd(a, b) = p_1^{\min(a_1, b_1)} p_2^{\min(a_2, b_2)} \cdots p_n^{\min(a_n, b_n)}$$

Gcd by prime factorisations

Suppose that the prime factorisations of a and b are

$$a = p_1^{a_1} p_2^{a_2} \cdots p_n^{a_n} \quad b = p_1^{b_1} p_2^{b_2} \cdots p_n^{b_n}$$

where each exponent is a nonnegative integer (possibly zero)

$$\gcd(a, b) = p_1^{\min(a_1, b_1)} p_2^{\min(a_2, b_2)} \cdots p_n^{\min(a_n, b_n)}$$

This number clearly divides a and b . No larger number can divide both a and b . Proof by contradiction and the prime factorisation of a postulated larger divisor.

Gcd by prime factorisations

Suppose that the prime factorisations of a and b are

$$a = p_1^{a_1} p_2^{a_2} \cdots p_n^{a_n}$$

$$b = p_1^{b_1} p_2^{b_2} \cdots p_n^{b_n}$$

where each exponent is a nonnegative integer (possibly zero)

$$\gcd(a, b) = p_1^{\min(a_1, b_1)} p_2^{\min(a_2, b_2)} \cdots p_n^{\min(a_n, b_n)}$$

This number clearly divides a and b . No larger number can divide both a and b . Proof by contradiction and the prime factorisation of a postulated larger divisor.

Factorisation is a very inefficient method to compute gcd

Euclidian algorithm: efficient for computing gcd

Euclidian algorithm

```
algorithm gcd(x, y)
    if y = 0
        then return(x)
        else return(gcd(y, x mod y))
```

Euclidian algorithm: efficient for computing gcd

Euclidian algorithm

```
algorithm gcd(x, y)
    if y = 0
        then return(x)
        else return(gcd(y, x mod y))
```

The Euclidian algorithm relies on

$$\forall x, y \in \mathbb{Z} (x > y \rightarrow \gcd(x, y) = \gcd(y, x \text{ mod } y))$$

Euclidian algorithm (proof of correctness)

Lemma

If $a = bq + r$, where a, b, q , and r are integers, then
 $\gcd(a, b) = \gcd(b, r)$

Euclidian algorithm (proof of correctness)

Lemma

If $a = bq + r$, where a, b, q , and r are integers, then
 $\gcd(a, b) = \gcd(b, r)$

Proof.

(\Rightarrow) Suppose that d divides both a and b . Then d also divides $a - bq = r$. Hence, any common divisor of a and b must also be a common divisor of b and r .

(\Leftarrow) Suppose that d divides both b and r . Then d also divides $bq + r = a$. Hence, any common divisor of b and r must also be a common divisor of a and b .

Therefore, $\gcd(a, b) = \gcd(b, r)$



Gcd as a linear combination

Theorem (Bézout's theorem)

If x and y are positive integers, then there exist integers a and b such that $\gcd(x, y) = ax + by$

Gcd as a linear combination

Theorem (Bézout's theorem)

If x and y are positive integers, then there exist integers a and b such that $\gcd(x, y) = ax + by$

Proof.

Let S be the set of positive integers of the form $ax + by$ (where a or b may be a negative integer); clearly, S is non-empty as it includes $x + y$. By the well-ordering principle S has a least element c . So $c = ax + by$ for some a and b . If $d|x$ and $d|y$ then $d|ax$ and $d|by$ and so $d|(ax + by)$, that is $d|c$. We now show $c|x$ and $c|y$ which means that $c = \gcd(x, y)$. Assume $c \nmid x$. So $x = qc + r$ where $0 < r < c$. Now $r = x - qc = x - q(ax + by)$. That is, $r = (1 - qa)x + (-qb)y$, so $r \in S$ which contradicts that c is the least element in S as $c < r$. The same argument shows $c|y$.



Computing Bézout coefficients

$$2 = \gcd(6, 14) = (-2) \cdot 6 + 1 \cdot 14$$

Computing Bézout coefficients

$$2 = \gcd(6, 14) = (-2) \cdot 6 + 1 \cdot 14$$

Extended Euclidian algorithm

```
algorithm extended-gcd(x, y)
    if y = 0
        then return(x, 1, 0)
    else
        (d, a, b) := extended-gcd(y, x mod y)
        return((d, b, a - ((x div y) * b)))
```

Further properties

Theorem

If a, b, c are positive integers such that $\gcd(a, b) = 1$ and $a|bc$ then $a|c$

Further properties

Theorem

If a, b, c are positive integers such that $\gcd(a, b) = 1$ and $a|bc$ then $a|c$

Proof.

Because $\gcd(a, b) = 1$, by Bézout's theorem there are integers s and t such that $sa + tb = 1$. So, $sac + tbc = c$. Assume $a|bc$. Therefore, $a|tbc$ and $a|sac$, so $a|(sac + tbc)$; that is, $a|c$.



Further properties

Theorem

If a, b, c are positive integers such that $\gcd(a, b) = 1$ and $a|bc$ then $a|c$

Proof.

Because $\gcd(a, b) = 1$, by Bézout's theorem there are integers s and t such that $sa + tb = 1$. So, $sac + tbc = c$. Assume $a|bc$. Therefore, $a|tbc$ and $a|sac$, so $a|(sac + tbc)$; that is, $a|c$.



Theorem

Let m be a positive integer and let a, b, c be integers. If $ac \equiv bc \pmod{m}$ and $\gcd(c, m) = 1$ then $a \equiv b \pmod{m}$

Further properties

Theorem

If a, b, c are positive integers such that $\gcd(a, b) = 1$ and $a|bc$ then $a|c$

Proof.

Because $\gcd(a, b) = 1$, by Bézout's theorem there are integers s and t such that $sa + tb = 1$. So, $sac + tbc = c$. Assume $a|bc$. Therefore, $a|tbc$ and $a|sac$, so $a|(sac + tbc)$; that is, $a|c$. □

Theorem

Let m be a positive integer and let a, b, c be integers. If $ac \equiv bc \pmod{m}$ and $\gcd(c, m) = 1$ then $a \equiv b \pmod{m}$

Proof.

Because $ac \equiv bc \pmod{m}$, it follows $m|(ac - bc)$; so, $m|c(a - b)$. By the result above because $\gcd(c, m) = 1$, it follows that $m|(a - b)$. □

Discrete Mathematics & Mathematical Reasoning

Multiplicative Inverses and Some Cryptography

Colin Stirling

Informatics

Some slides based on ones by Myrto Arapinis

Multiplicative inverses

Theorem

If m, x are positive integers and $\gcd(m, x) = 1$ then x has a multiplicative inverse modulo m (and it is unique modulo m)

Multiplicative inverses

Theorem

If m, x are positive integers and $\gcd(m, x) = 1$ then x has a multiplicative inverse modulo m (and it is unique modulo m)

Proof.

Consider the sequence of m numbers $0, x, 2x, \dots, (m - 1)x$. We first show that these are all distinct modulo m .

To verify the above claim, suppose that $ax \bmod m = bx \bmod m$ for two distinct values a, b in the range $0 \leq a, b \leq m - 1$. Then we would have $(a - b)x \equiv 0 \pmod{m}$, or equivalently, $(a - b)x = km$ for some integer k . But since x and m are relatively prime, it follows that $a - b$ must be an integer multiple of m . This is not possible since a, b are distinct non-negative integers less than m .

Now, since there are only m distinct values modulo m , it must then be the case that $ax \equiv 1 \pmod{m}$ for exactly one a (modulo m). This a is the unique multiplicative inverse. □

Chinese remainder theorem

Theorem

Let m_1, m_2, \dots, m_n be pairwise relatively prime positive integers greater than 1 and a_1, a_2, \dots, a_n be arbitrary integers. Then the system

$$x \equiv a_1 \pmod{m_1}$$

$$x \equiv a_2 \pmod{m_2}$$

⋮

$$x \equiv a_n \pmod{m_n}$$

has a unique solution modulo $m = m_1 m_2 \cdots m_n$

Chinese remainder theorem

Theorem

Let m_1, m_2, \dots, m_n be pairwise relatively prime positive integers greater than 1 and a_1, a_2, \dots, a_n be arbitrary integers. Then the system

$$x \equiv a_1 \pmod{m_1}$$

$$x \equiv a_2 \pmod{m_2}$$

⋮

$$x \equiv a_n \pmod{m_n}$$

has a unique solution modulo $m = m_1 m_2 \cdots m_n$

Proof.

In the book



Example

$$x \equiv 2 \pmod{3}$$

$$x \equiv 3 \pmod{5}$$

$$x \equiv 5 \pmod{7}$$

Example

$$x \equiv 2 \pmod{3}$$

$$x \equiv 3 \pmod{5}$$

$$x \equiv 5 \pmod{7}$$

- $m = 3 \cdot 5 \cdot 7 = 105$

Example

$$x \equiv 2 \pmod{3}$$

$$x \equiv 3 \pmod{5}$$

$$x \equiv 5 \pmod{7}$$

- $m = 3 \cdot 5 \cdot 7 = 105$
- $M_1 = 35$ and 2 is an inverse of M_1 mod 3

Example

$$x \equiv 2 \pmod{3}$$

$$x \equiv 3 \pmod{5}$$

$$x \equiv 5 \pmod{7}$$

- $m = 3 \cdot 5 \cdot 7 = 105$
- $M_1 = 35$ and 2 is an inverse of M_1 mod 3
- $M_2 = 21$ and 1 is an inverse of M_2 mod 5

Example

$$x \equiv 2 \pmod{3}$$

$$x \equiv 3 \pmod{5}$$

$$x \equiv 5 \pmod{7}$$

- $m = 3 \cdot 5 \cdot 7 = 105$
- $M_1 = 35$ and 2 is an inverse of M_1 mod 3
- $M_2 = 21$ and 1 is an inverse of M_2 mod 5
- $M_3 = 15$ and 1 is an inverse of M_3 mod 7

Example

$$x \equiv 2 \pmod{3}$$

$$x \equiv 3 \pmod{5}$$

$$x \equiv 5 \pmod{7}$$

- $m = 3 \cdot 5 \cdot 7 = 105$
- $M_1 = 35$ and 2 is an inverse of M_1 mod 3
- $M_2 = 21$ and 1 is an inverse of M_2 mod 5
- $M_3 = 15$ and 1 is an inverse of M_3 mod 7
- $x = 2 \cdot 35 \cdot 2 + 3 \cdot 21 \cdot 1 + 5 \cdot 15 \cdot 1$

Example

$$x \equiv 2 \pmod{3}$$

$$x \equiv 3 \pmod{5}$$

$$x \equiv 5 \pmod{7}$$

- $m = 3 \cdot 5 \cdot 7 = 105$
- $M_1 = 35$ and 2 is an inverse of M_1 mod 3
- $M_2 = 21$ and 1 is an inverse of M_2 mod 5
- $M_3 = 15$ and 1 is an inverse of M_3 mod 7
- $x = 2 \cdot 35 \cdot 2 + 3 \cdot 21 \cdot 1 + 5 \cdot 15 \cdot 1$
- $x = 140 + 63 + 75 = 278 \equiv 68 \pmod{105}$

Fermat's little theorem

Theorem

If p is prime and $p \nmid a$, then $a^{p-1} \equiv 1 \pmod{p}$. Furthermore, for every integer a we have $a^p \equiv a \pmod{p}$

Fermat's little theorem

Theorem

If p is prime and $p \nmid a$, then $a^{p-1} \equiv 1 \pmod{p}$. Furthermore, for every integer a we have $a^p \equiv a \pmod{p}$

Proof.

Assume $p \nmid a$ and so, therefore, $\gcd(p, a) = 1$. Then $a, 2a, \dots, (p-1)a$ are not pairwise congruent modulo p ; if $ia \equiv ja \pmod{p}$ then $(i-j)a = pm$ for some m which is impossible (as then $i \equiv j \pmod{p}$ using last result from slides of Lecture 11). Therefore, each element $ja \pmod{p}$ is a distinct element in the set $\{1, \dots, p-1\}$. This means that the product $a \cdot 2a \cdots (p-1)a \equiv 1 \cdot 2 \cdots p-1 \pmod{p}$. Therefore, $(p-1)!a^{p-1} \equiv (p-1)! \pmod{p}$. Now because $\gcd(p, q) = 1$ for $1 \leq q \leq p-1$ it follows that $a^{p-1} \equiv 1 \pmod{p}$. Therefore, also $a^p \equiv a \pmod{p}$ and when $p|a$ then clearly $a^p \equiv a \pmod{p}$.



Computing the remainders modulo prime p

- Find $7^{222} \bmod 11$

Computing the remainders modulo prime p

- Find $7^{222} \bmod 11$
- By Fermat's little theorem, we know that $7^{10} \equiv 1 \pmod{11}$, and so $(7^{10})^k \equiv 1 \pmod{11}$ for every positive integer k . Therefore,
 $7^{222} = 7^{22 \cdot 10 + 2} = (7^{10})^{22} 7^2 \equiv 1^{22} 49 \equiv 5 \pmod{11}$. Hence,
 $7^{222} \bmod 11 = 5$

Computing the remainders modulo prime p

- Find $7^{222} \bmod 11$
- By Fermat's little theorem, we know that $7^{10} \equiv 1 \pmod{11}$, and so $(7^{10})^k \equiv 1 \pmod{11}$ for every positive integer k . Therefore,
 $7^{222} = 7^{22 \cdot 10 + 2} = (7^{10})^{22} 7^2 \equiv 1^{22} 49 \equiv 5 \pmod{11}$. Hence,
 $7^{222} \bmod 11 = 5$
- $2^{340} \equiv 1 \pmod{11}$ because $2^{10} \equiv 1 \pmod{11}$

Private key cryptography

- Bob wants to send Alice a secret message M

Private key cryptography

- Bob wants to send Alice a secret message M
- Alice sends Bob a private key En (which has an inverse De)

Private key cryptography

- Bob wants to send Alice a secret message M
- Alice sends Bob a private key En (which has an inverse De)
- Bob encrypts M and sends Alice $En(M)$

Private key cryptography

- Bob wants to send Alice a secret message M
- Alice sends Bob a private key En (which has an inverse De)
- Bob encrypts M and sends Alice $En(M)$
- Alice decrypts $En(M)$, $De(En(M))$

Private key cryptography

- Bob wants to send Alice a secret message M
- Alice sends Bob a private key En (which has an inverse De)
- Bob encrypts M and sends Alice $En(M)$
- Alice decrypts $En(M)$, $De(En(M))$
- Important property $De(En(M)) = M$

Private key cryptography

- Bob wants to send Alice a secret message M
- Alice sends Bob a private key En (which has an inverse De)
- Bob encrypts M and sends Alice $En(M)$
- Alice decrypts $En(M)$, $De(En(M))$
- Important property $De(En(M)) = M$
- Alice and Bob share a secret which could be intercepted by a third party

Private key cryptography

- Bob wants to send Alice a secret message M
- Alice sends Bob a private key En (which has an inverse De)
- Bob encrypts M and sends Alice En(M)
- Alice decrypts En(M), De(En(M))
- Important property $De(En(M)) = M$
- Alice and Bob share a secret which could be intercepted by a third party
- Example use $En(p) = (p + 3) \text{ mod } 26$

Private key cryptography

- Bob wants to send Alice a secret message M
- Alice sends Bob a private key En (which has an inverse De)
- Bob encrypts M and sends Alice En(M)
- Alice decrypts En(M), De(En(M))
- Important property $De(En(M)) = M$
- Alice and Bob share a secret which could be intercepted by a third party
- Example use $En(p) = (p + 3) \text{ mod } 26$
- What is WKLV LV D VHFSHW ?

Public key cryptography

- Bob wants to send Alice a secret message M

Public key cryptography

- Bob wants to send Alice a secret message M
- Without Alice and Bob sharing a secret

Public key cryptography

- Bob wants to send Alice a secret message M
- Without Alice and Bob sharing a secret
- Alice sends Bob a public key En (and keeps her inverse private key De secret from everyone including Bob)

Public key cryptography

- Bob wants to send Alice a secret message M
- Without Alice and Bob sharing a secret
- Alice sends Bob a public key En (and keeps her inverse private key De secret from everyone including Bob)
- Bob encrypts M and sends Alice $En(M)$

Public key cryptography

- Bob wants to send Alice a secret message M
- Without Alice and Bob sharing a secret
- Alice sends Bob a public key En (and keeps her inverse private key De secret from everyone including Bob)
- Bob encrypts M and sends Alice $En(M)$
- Alice decrypts $En(M)$, $De(En(M))$

Public key cryptography

- Bob wants to send Alice a secret message M
- Without Alice and Bob sharing a secret
- Alice sends Bob a public key En (and keeps her inverse private key De secret from everyone including Bob)
- Bob encrypts M and sends Alice $En(M)$
- Alice decrypts $En(M)$, $De(En(M))$
- Important property $De(En(M)) = M$

Public key cryptography

- Bob wants to send Alice a secret message M
- Without Alice and Bob sharing a secret
- Alice sends Bob a public key En (and keeps her inverse private key De secret from everyone including Bob)
- Bob encrypts M and sends Alice $En(M)$
- Alice decrypts $En(M)$, $De(En(M))$
- Important property $De(En(M)) = M$
- The challenge: De can't be feasibly computed from En ; and given $En(M)$ one can't feasibly compute M

RSA Cryptosystem

- Named after 3 researchers: Rivest, Shamir and Adleman

RSA Cryptosystem

- Named after 3 researchers: Rivest, Shamir and Adleman
- There are quick algorithms for testing whether a large integer is prime

RSA Cryptosystem

- Named after 3 researchers: Rivest, Shamir and Adleman
- There are quick algorithms for testing whether a large integer is prime
- There is no known quick algorithm that can factorise a large integer

RSA Cryptosystem

- Named after 3 researchers: Rivest, Shamir and Adleman
- There are quick algorithms for testing whether a large integer is prime
- There is no known quick algorithm that can factorise a large integer
- Very significant open problem: how hard is it to factorise integers?

RSA: key generation

- Choose two distinct prime numbers p and q
- Let $n = pq$ and $k = (p - 1)(q - 1)$
- Choose integer e where $1 < e < k$ and $\gcd(e, k) = 1$
- (n, e) is released as the public key
- Let d be the multiplicative inverse of e modulo k , so $de \equiv 1 \pmod{k}$
- (n, d) is the private key and kept secret

RSA: encryption and decryption

Alice transmits her public key (n, e) to Bob and keeps the private key secret

Encryption If Bob wishes to send message M to Alice.

- ① He turns M into an integer m , such that $0 \leq m < n$ by using an agreed-upon reversible protocol known as a padding scheme
- ② He computes the ciphertext c corresponding to $c = m^e \bmod n$.
(This can be done quickly)
- ③ Bob transmits c to Alice.

Decryption Alice can recover m from c by

- ① Using her private key exponent d via computing $m = c^d \bmod n$
- ② Given m , she can recover the original message M by reversing the padding scheme

Unrealistic example

- $n = 43 \cdot 59 = 2537$

Unrealistic example

- $n = 43 \cdot 59 = 2537$
- $\gcd(13, 42 \cdot 58) = 1$, so public key is $(2537, 13)$

Unrealistic example

- $n = 43 \cdot 59 = 2537$
- $\gcd(13, 42 \cdot 58) = 1$, so public key is $(2537, 13)$
- $d = 937$ is inverse of 13 modulo $2436 = 42 \cdot 58$; private key $(2537, 937)$

Unrealistic example

- $n = 43 \cdot 59 = 2537$
- $\gcd(13, 42 \cdot 58) = 1$, so public key is $(2537, 13)$
- $d = 937$ is inverse of 13 modulo $2436 = 42 \cdot 58$; private key $(2537, 937)$
- Encrypt STOP as two blocks 1819 for ST and 1415 for OP (padding scheme)

Unrealistic example

- $n = 43 \cdot 59 = 2537$
- $\gcd(13, 42 \cdot 58) = 1$, so public key is $(2537, 13)$
- $d = 937$ is inverse of 13 modulo $2436 = 42 \cdot 58$; private key $(2537, 937)$
- Encrypt STOP as two blocks 1819 for ST and 1415 for OP (padding scheme)
- So, $1819^{13} \bmod 2537 = 2081$ and $1415^{13} \bmod 2537 = 2182$

Unrealistic example

- $n = 43 \cdot 59 = 2537$
- $\gcd(13, 42 \cdot 58) = 1$, so public key is $(2537, 13)$
- $d = 937$ is inverse of 13 modulo $2436 = 42 \cdot 58$; private key $(2537, 937)$
- Encrypt STOP as two blocks 1819 for ST and 1415 for OP (padding scheme)
- So, $1819^{13} \bmod 2537 = 2081$ and $1415^{13} \bmod 2537 = 2182$
- So encrypted message is 2081 2182

RSA: correctness of decryption

Given that $c = m^e \text{ mod } n$, is $m = c^d \text{ mod } n$?

$$c^d = (m^e)^d \equiv m^{ed} \pmod{n}$$

By construction, d and e are each others multiplicative inverses modulo k , i.e. $ed \equiv 1 \pmod{k}$. Also $k = (p - 1)(q - 1)$. Thus $ed - 1 = h(p - 1)(q - 1)$ for some integer h . We consider $m^{ed} \text{ mod } p$
If $p \nmid m$ then

$m^{ed} = m^{h(p-1)(q-1)}m = (m^{p-1})^{h(q-1)}m \equiv 1^{h(q-1)}m \equiv m \pmod{p}$ (by Fermat's little theorem)

Otherwise $m^{ed} \equiv 0 \equiv m \pmod{p}$

Symmetrically, $m^{ed} \equiv m \pmod{q}$

Since p, q are distinct primes, we have $m^{ed} \equiv m \pmod{pq}$. Since $n = pq$, we have $c^d = m^{ed} \equiv m \pmod{n}$

Discrete Mathematics & Mathematical Reasoning Algorithms

Colin Stirling

Informatics

Some slides based on ones by Myrto Arapinis

Algorithms

Definition

An algorithm is a finite sequence of precise instructions for performing a computation or for solving a problem

Algorithms

Definition

An algorithm is a finite sequence of precise instructions for performing a computation or for solving a problem

Euclidian algorithm

```
algorithm gcd(x, y)
    if y = 0
        then return(x)
        else return(gcd(y, x mod y))
```

Properties of an algorithm

Input it has input values from specified sets

Properties of an algorithm

Input it has input values from specified sets

Output from the input values, it produces the output values from specified sets which are the solution

Properties of an algorithm

Input it has input values from specified sets

Output from the input values, it produces the output values from specified sets which are the solution

Correctness it should produce the correct output values for each set of input values

Properties of an algorithm

Input it has input values from specified sets

Output from the input values, it produces the output values from specified sets which are the solution

Correctness it should produce the correct output values for each set of input values

Finiteness it should produce the output after a finite number of steps for any input

Properties of an algorithm

Input it has input values from specified sets

Output from the input values, it produces the output values from specified sets which are the solution

Correctness it should produce the correct output values for each set of input values

Finiteness it should produce the output after a finite number of steps for any input

Effectiveness it must be possible to perform each step correctly and in a finite amount of time

Properties of an algorithm

Input it has input values from specified sets

Output from the input values, it produces the output values from specified sets which are the solution

Correctness it should produce the correct output values for each set of input values

Finiteness it should produce the output after a finite number of steps for any input

Effectiveness it must be possible to perform each step correctly and in a finite amount of time

Generality it should work for all problems of the desired form

Euclidian algorithm (proof of correctness)

Lemma

If $a = bq + r$, where a, b, q , and r are positive integers, then
 $\gcd(a, b) = \gcd(b, r)$

Euclidian algorithm (proof of correctness)

Lemma

If $a = bq + r$, where a, b, q , and r are positive integers, then $\gcd(a, b) = \gcd(b, r)$

Proof.

(\Rightarrow) Suppose that d divides both a and b . Then d also divides $a - bq = r$. Hence, any common divisor of a and b must also be a common divisor of b and r .

(\Leftarrow) Suppose that d divides both b and r . Then d also divides $bq + r = a$. Hence, any common divisor of b and r must also be a common divisor of a and b .

Therefore, $\gcd(a, b) = \gcd(b, r)$



Description of algorithms in pseudocode

- Intermediate step between English prose and formal coding in a programming language

Description of algorithms in pseudocode

- Intermediate step between English prose and formal coding in a programming language
- Focus on the fundamental operation of the program, instead of peculiarities of a given programming language

Description of algorithms in pseudocode

- Intermediate step between English prose and formal coding in a programming language
- Focus on the fundamental operation of the program, instead of peculiarities of a given programming language
- Analyze the time required to solve a problem using an algorithm, independent of the actual programming language

Maximum

Describe an algorithm for finding the maximum value in a finite sequence of integers

Maximum

Describe an algorithm for finding the maximum value in a finite sequence of integers

Input finite sequence of integers a_1, \dots, a_n

Maximum

Describe an algorithm for finding the maximum value in a finite sequence of integers

Input finite sequence of integers a_1, \dots, a_n

Output a_k , $k \in \{1, \dots, n\}$, where for all $j \in \{1, \dots, n\}$, $a_j \leq a_k$

Maximum

Describe an algorithm for finding the maximum value in a finite sequence of integers

Input finite sequence of integers a_1, \dots, a_n

Output a_k , $k \in \{1, \dots, n\}$, where for all $j \in \{1, \dots, n\}$, $a_j \leq a_k$

```
procedure maximum(a1, ..., an)
max:=a1
for i:=2 to n
    if max<ai
        then max:=ai
return max
```

Linear search

Describe an algorithm for locating an item in a sequence of integers

Input integer x and finite sequence of integers a_1, \dots, a_n

Linear search

Describe an algorithm for locating an item in a sequence of integers

Input integer x and finite sequence of integers a_1, \dots, a_n

Output integer $i \in \{0, \dots, n\}$ where $a_i = x$ or $i = 0$ if $x \neq a_j$ for all a_j

Linear search

Describe an algorithm for locating an item in a sequence of integers

Input integer x and finite sequence of integers a_1, \dots, a_n

Output integer $i \in \{0, \dots, n\}$ where $a_i = x$ or $i = 0$ if $x \neq a_j$ for all a_j

```
procedure linear_search(x, a1, ..., an)
    i := 1
    while i ≤ n and x ≠ ai
        i := i + 1
    if i ≤ n
        then location := i
    else location := 0
    return location
```

Binary search

Describe an algorithm for locating an item in an ordered sequence of integers

Binary search

Describe an algorithm for locating an item in an ordered sequence of integers

Input integer x and finite ordered sequence of integers a_1, \dots, a_n

Binary search

Describe an algorithm for locating an item in an ordered sequence of integers

Input integer x and finite ordered sequence of integers a_1, \dots, a_n

Output integer $i \in \{0, \dots, n\}$ where $a_i = x$ or $i = 0$ if $x \neq a_j$ for all a_j

Binary search

Describe an algorithm for locating an item in an ordered sequence of integers

Input integer x and finite ordered sequence of integers a_1, \dots, a_n

Output integer $i \in \{0, \dots, n\}$ where $a_i = x$ or $i = 0$ if $x \neq a_j$ for all a_j

- The algorithm begins by comparing x with the middle element

Binary search

Describe an algorithm for locating an item in an ordered sequence of integers

Input integer x and finite ordered sequence of integers a_1, \dots, a_n

Output integer $i \in \{0, \dots, n\}$ where $a_i = x$ or $i = 0$ if $x \neq a_j$ for all a_j

- The algorithm begins by comparing x with the middle element
- If the middle element is strictly smaller than x , then the search proceeds with the upper half of the list

Binary search

Describe an algorithm for locating an item in an ordered sequence of integers

Input integer x and finite ordered sequence of integers a_1, \dots, a_n

Output integer $i \in \{0, \dots, n\}$ where $a_i = x$ or $i = 0$ if $x \neq a_j$ for all a_j

- The algorithm begins by comparing x with the middle element
- If the middle element is strictly smaller than x , then the search proceeds with the upper half of the list
- Otherwise the search proceeds with the lower half of the list (including the middle element)

Binary search

Describe an algorithm for locating an item in an ordered sequence of integers

Input integer x and finite ordered sequence of integers a_1, \dots, a_n

Output integer $i \in \{0, \dots, n\}$ where $a_i = x$ or $i = 0$ if $x \neq a_j$ for all a_j

- The algorithm begins by comparing x with the middle element
- If the middle element is strictly smaller than x , then the search proceeds with the upper half of the list
- Otherwise the search proceeds with the lower half of the list (including the middle element)
- Repeat this process until we have a list of size 1

Binary search

Describe an algorithm for locating an item in an ordered sequence of integers

Input integer x and finite ordered sequence of integers a_1, \dots, a_n

Output integer $i \in \{0, \dots, n\}$ where $a_i = x$ or $i = 0$ if $x \neq a_j$ for all a_j

- The algorithm begins by comparing x with the middle element
- If the middle element is strictly smaller than x , then the search proceeds with the upper half of the list
- Otherwise the search proceeds with the lower half of the list (including the middle element)
- Repeat this process until we have a list of size 1
- If x is equal to the single element in the list, then its position is returned

Binary search

Describe an algorithm for locating an item in an ordered sequence of integers

Input integer x and finite ordered sequence of integers a_1, \dots, a_n

Output integer $i \in \{0, \dots, n\}$ where $a_i = x$ or $i = 0$ if $x \neq a_j$ for all a_j

- The algorithm begins by comparing x with the middle element
- If the middle element is strictly smaller than x , then the search proceeds with the upper half of the list
- Otherwise the search proceeds with the lower half of the list (including the middle element)
- Repeat this process until we have a list of size 1
- If x is equal to the single element in the list, then its position is returned
- Otherwise 0 is returned to indicate that the element was not found

Binary search

```
procedure binary_search(x, a1, ..., an)
i := 1
j := n
while i < j
    m := ⌊(i + j)/2⌋
    if x > am
        then i := m + 1
    else j := m
if x = ai
then location := i
else location := 0
return location
```

Big-O notation for function growth

Definition

Let $f, g : \mathbb{N} \rightarrow \mathbb{R}$ or $f, g : \mathbb{R} \rightarrow \mathbb{R}$. Then f is $O(g)$ if there is a constant k and a positive constant c such that

$$\forall x > k (|f(x)| \leq c|g(x)|)$$

Big-O notation for function growth

Definition

Let $f, g : \mathbb{N} \rightarrow \mathbb{R}$ or $f, g : \mathbb{R} \rightarrow \mathbb{R}$. Then f is $O(g)$ if there is a constant k and a positive constant c such that

$$\forall x > k (|f(x)| \leq c|g(x)|)$$

- c and k are witnesses to the relationship between f and g

Big-O notation for function growth

Definition

Let $f, g : \mathbb{N} \rightarrow \mathbb{R}$ or $f, g : \mathbb{R} \rightarrow \mathbb{R}$. Then f is $O(g)$ if there is a constant k and a positive constant c such that

$$\forall x > k (|f(x)| \leq c|g(x)|)$$

- c and k are witnesses to the relationship between f and g
- $O(g)$ is the set of all functions f that satisfy the condition above: it would be formally correct to write $f \in O(g)$

Big-O notation for function growth

Definition

Let $f, g : \mathbb{N} \rightarrow \mathbb{R}$ or $f, g : \mathbb{R} \rightarrow \mathbb{R}$. Then f is $O(g)$ if there is a constant k and a positive constant c such that

$$\forall x > k (|f(x)| \leq c|g(x)|)$$

- c and k are witnesses to the relationship between f and g
- $O(g)$ is the set of all functions f that satisfy the condition above: it would be formally correct to write $f \in O(g)$
- Often the condition is: $\forall x > k (f(x) \leq cg(x))$

Examples

- $f(x) = x^2 + 2x + 1$

Examples

- $f(x) = x^2 + 2x + 1$
 - Show $f(x)$ is $O(g)$ where $g(x) = x^2$

Examples

- $f(x) = x^2 + 2x + 1$
- Show $f(x)$ is $O(g)$ where $g(x) = x^2$
- Show $f(x)$ is also $O(g)$ where $g(x) = x^3$

Examples

- $f(x) = x^2 + 2x + 1$
- Show $f(x)$ is $O(g)$ where $g(x) = x^2$
- Show $f(x)$ is also $O(g)$ where $g(x) = x^3$
- Show $f(x)$ is not $O(h)$ where $h(x) = x$

Examples

- $f(x) = a_nx^n + a_{n-1}x^{n-1} + \dots + a_1x + a_0$ is $O(x^n)$

Examples

- $f(x) = a_nx^n + a_{n-1}x^{n-1} + \dots + a_1x + a_0$ is $O(x^n)$
- $f(x) = 1 + 2 + \dots + x$ is $O(x^2)$

Examples

- $f(x) = a_nx^n + a_{n-1}x^{n-1} + \dots + a_1x + a_0$ is $O(x^n)$
- $f(x) = 1 + 2 + \dots + x$ is $O(x^2)$
- $\log(n)$ is $O(n)$

Examples

- $f(x) = a_nx^n + a_{n-1}x^{n-1} + \dots + a_1x + a_0$ is $O(x^n)$
- $f(x) = 1 + 2 + \dots + x$ is $O(x^2)$
- $\log(n)$ is $O(n)$
- $n! = 1 \times 2 \times \dots \times n$ is $O(n^n)$

Examples

- $f(x) = a_nx^n + a_{n-1}x^{n-1} + \dots + a_1x + a_0$ is $O(x^n)$
- $f(x) = 1 + 2 + \dots + x$ is $O(x^2)$
- $\log(n)$ is $O(n)$
- $n! = 1 \times 2 \times \dots \times n$ is $O(n^n)$
- $\log(n!)$ is $O(n \log(n))$

Big-Omega notation for function growth

Definition

Let $f, g : \mathbb{N} \rightarrow \mathbb{R}$ or $f, g : \mathbb{R} \rightarrow \mathbb{R}$. Then f is $\Omega(g)$ if there is a constant k and a positive constant c such that

$$\forall x > k (|f(x)| \geq c|g(x)|)$$

Big-Omega notation for function growth

Definition

Let $f, g : \mathbb{N} \rightarrow \mathbb{R}$ or $f, g : \mathbb{R} \rightarrow \mathbb{R}$. Then f is $\Omega(g)$ if there is a constant k and a positive constant c such that

$$\forall x > k (|f(x)| \geq c|g(x)|)$$

- c and k are witnesses to the relationship between f and g

Big-Omega notation for function growth

Definition

Let $f, g : \mathbb{N} \rightarrow \mathbb{R}$ or $f, g : \mathbb{R} \rightarrow \mathbb{R}$. Then f is $\Omega(g)$ if there is a constant k and a positive constant c such that

$$\forall x > k (|f(x)| \geq c|g(x)|)$$

- c and k are witnesses to the relationship between f and g
- Big- O gives an upper bound on the growth of a function, while Big-Omega gives a lower bound

Big-Omega notation for function growth

Definition

Let $f, g : \mathbb{N} \rightarrow \mathbb{R}$ or $f, g : \mathbb{R} \rightarrow \mathbb{R}$. Then f is $\Omega(g)$ if there is a constant k and a positive constant c such that

$$\forall x > k (|f(x)| \geq c|g(x)|)$$

- c and k are witnesses to the relationship between f and g
- Big- O gives an upper bound on the growth of a function, while Big-Omega gives a lower bound
- Often the condition is: $\forall x > k (f(x) \geq cg(x))$

Big-Omega notation for function growth

Definition

Let $f, g : \mathbb{N} \rightarrow \mathbb{R}$ or $f, g : \mathbb{R} \rightarrow \mathbb{R}$. Then f is $\Omega(g)$ if there is a constant k and a positive constant c such that

$$\forall x > k (|f(x)| \geq c|g(x)|)$$

- c and k are witnesses to the relationship between f and g
- Big- O gives an upper bound on the growth of a function, while Big-Omega gives a lower bound
- Often the condition is: $\forall x > k (f(x) \geq cg(x))$
- f is $\Omega(g)$ if and only if g is $O(f)$

Big-Theta notation for function growth

Definition

Let $f, g : \mathbb{N} \rightarrow \mathbb{R}$ or $f, g : \mathbb{R} \rightarrow \mathbb{R}$. Then f is $\Theta(g)$ iff f is $O(g)$ and $\Omega(g)$

Big-Theta notation for function growth

Definition

Let $f, g : \mathbb{N} \rightarrow \mathbb{R}$ or $f, g : \mathbb{R} \rightarrow \mathbb{R}$. Then f is $\Theta(g)$ iff f is $O(g)$ and $\Omega(g)$

- f and g are of the same order

Big-Theta notation for function growth

Definition

Let $f, g : \mathbb{N} \rightarrow \mathbb{R}$ or $f, g : \mathbb{R} \rightarrow \mathbb{R}$. Then f is $\Theta(g)$ iff f is $O(g)$ and $\Omega(g)$

- f and g are of the same order
- f is $\Theta(g)$ iff there exists constants c_1, c_2 and k such that

for all $x > k(c_1|g(x)| \leq |f(x)| \leq c_2|g(x)|)$

Big-Theta notation for function growth

Definition

Let $f, g : \mathbb{N} \rightarrow \mathbb{R}$ or $f, g : \mathbb{R} \rightarrow \mathbb{R}$. Then f is $\Theta(g)$ iff f is $O(g)$ and $\Omega(g)$

- f and g are of the same order
- f is $\Theta(g)$ iff there exists constants c_1, c_2 and k such that

$$\text{for all } x > k(c_1|g(x)| \leq |f(x)| \leq c_2|g(x)|)$$

- $f(x) = a_nx^n + a_{n-1}x^{n-1} + \dots + a_1x + a_0$ is $\Theta(x^n)$ if $a_n \neq 0$

Big-Theta notation for function growth

Definition

Let $f, g : \mathbb{N} \rightarrow \mathbb{R}$ or $f, g : \mathbb{R} \rightarrow \mathbb{R}$. Then f is $\Theta(g)$ iff f is $O(g)$ and $\Omega(g)$

- f and g are of the same order
- f is $\Theta(g)$ iff there exists constants c_1, c_2 and k such that

$$\text{for all } x > k(c_1|g(x)| \leq |f(x)| \leq c_2|g(x)|)$$

- $f(x) = a_nx^n + a_{n-1}x^{n-1} + \dots + a_1x + a_0$ is $\Theta(x^n)$ if $a_n \neq 0$
- $f(x) = 1 + 2 + \dots + x$ is $\Theta(x^2)$

Complexity of algorithms

- Given an algorithm, how efficient is it for solving the problem relative to input size?

Complexity of algorithms

- Given an algorithm, how efficient is it for solving the problem relative to input size?
- How much time does it take or how much computer memory does it need

Complexity of algorithms

- Given an algorithm, how efficient is it for solving the problem relative to input size?
- How much time does it take or how much computer memory does it need
- We measure time complexity in terms of the number of basic operations executed and use big- O and big-Theta notation to estimate it

Complexity of algorithms

- Given an algorithm, how efficient is it for solving the problem relative to input size?
- How much time does it take or how much computer memory does it need
- We measure time complexity in terms of the number of basic operations executed and use big- O and big-Theta notation to estimate it
- Focus on worst-case time complexity. Derive an upper bound on the number of operations it uses to solve a problem with input of particular size (as opposed to the average-case complexity)

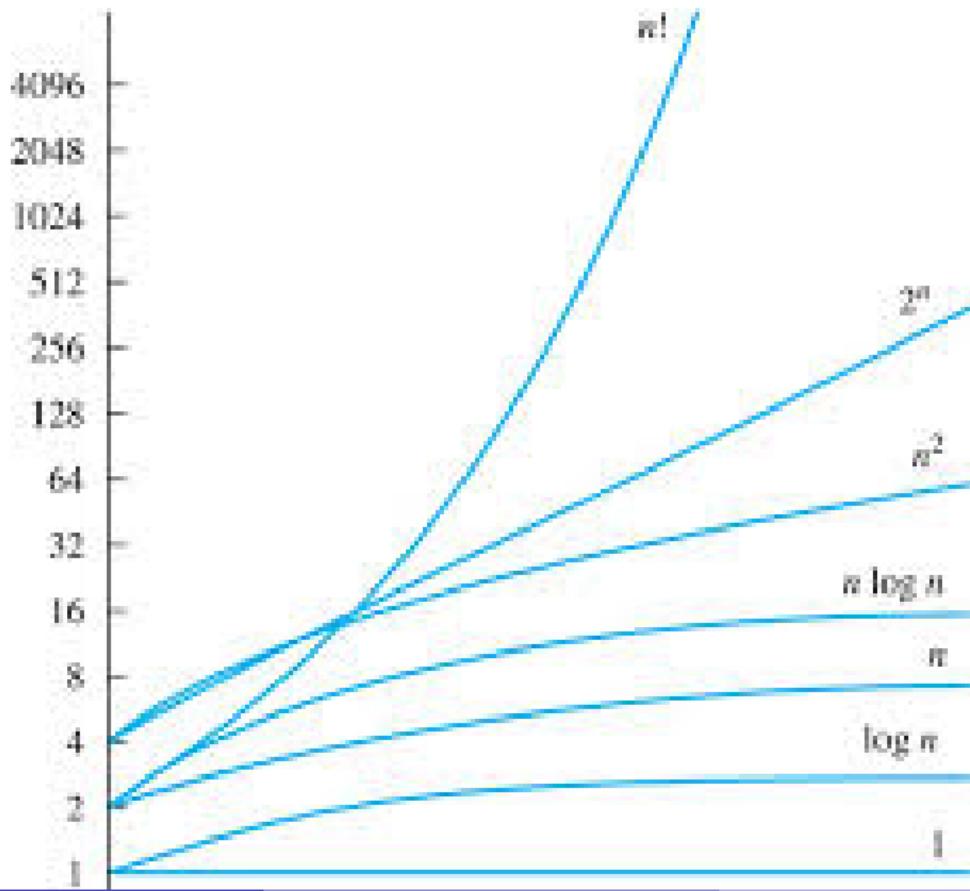
Complexity of algorithms

- Given an algorithm, how efficient is it for solving the problem relative to input size?
- How much time does it take or how much computer memory does it need
- We measure time complexity in terms of the number of basic operations executed and use big- O and big-Theta notation to estimate it
- Focus on worst-case time complexity. Derive an upper bound on the number of operations it uses to solve a problem with input of particular size (as opposed to the average-case complexity)
- Compute an $f(n)$ as worst case for input size n

Complexity of algorithms

- Given an algorithm, how efficient is it for solving the problem relative to input size?
- How much time does it take or how much computer memory does it need
- We measure time complexity in terms of the number of basic operations executed and use big- O and big-Theta notation to estimate it
- Focus on worst-case time complexity. Derive an upper bound on the number of operations it uses to solve a problem with input of particular size (as opposed to the average-case complexity)
- Compute an $f(n)$ as worst case for input size n
- Compare efficiency of different algorithms for the same problem

Growth



Linear search

```
procedure linear_search(x, a1, ..., an)
i := 1
while i ≤ n and x ≠ ai
    i := i + 1
if i ≤ n
then location := i
else location := 0
return location
```

Worst-Case complexity of linear search

- Count the number of comparisons

Worst-Case complexity of linear search

- Count the number of comparisons
- at each step two comparisons are made $i \leq n$ and $x \neq a_i$

Worst-Case complexity of linear search

- Count the number of comparisons
- at each step two comparisons are made $i \leq n$ and $x \neq a_i$
- to end the loop, one comparison $i \leq n$ is made

Worst-Case complexity of linear search

- Count the number of comparisons
- at each step two comparisons are made $i \leq n$ and $x \neq a_i$
- to end the loop, one comparison $i \leq n$ is made
- after the loop, one more $i \leq n$ comparison is made

Worst-Case complexity of linear search

- Count the number of comparisons
- at each step two comparisons are made $i \leq n$ and $x \neq a_i$
- to end the loop, one comparison $i \leq n$ is made
- after the loop, one more $i \leq n$ comparison is made
- If $x = a_i$, $2i + 1$ comparisons are used

Worst-Case complexity of linear search

- Count the number of comparisons
- at each step two comparisons are made $i \leq n$ and $x \neq a_i$
- to end the loop, one comparison $i \leq n$ is made
- after the loop, one more $i \leq n$ comparison is made
- If $x = a_i$, $2i + 1$ comparisons are used
- If x is not in the list, $2n + 2$ comparisons are made which is the worst case

Worst-Case complexity of linear search

- Count the number of comparisons
- at each step two comparisons are made $i \leq n$ and $x \neq a_i$
- to end the loop, one comparison $i \leq n$ is made
- after the loop, one more $i \leq n$ comparison is made
- If $x = a_i$, $2i + 1$ comparisons are used
- If x is not in the list, $2n + 2$ comparisons are made which is the worst case
- This means that the complexity is $\Theta(n)$

Binary search

```
procedure binary_search(x, a1, ..., an)
i := 1
j := n
while i < j
    m := ⌊(i + j)/2⌋
    if x > am
        then i := m + 1
    else j := m
if x = ai
then location := i
else location := 0
return location
```

Worst-Case complexity of binary search

- Assume (for simplicity) $n = 2^k$; so $k = \log_2 n$

Worst-Case complexity of binary search

- Assume (for simplicity) $n = 2^k$; so $k = \log_2 n$
- Two comparisons are made at each stage $i < j$ and $x > a_m$

Worst-Case complexity of binary search

- Assume (for simplicity) $n = 2^k$; so $k = \log_2 n$
- Two comparisons are made at each stage $i < j$ and $x > a_m$
- At the first iteration the size of the list is 2^k ; after the first iteration it is 2^{k-1} . Then 2^{k-2} and so on until the size of the list is $2^1 = 2$

Worst-Case complexity of binary search

- Assume (for simplicity) $n = 2^k$; so $k = \log_2 n$
- Two comparisons are made at each stage $i < j$ and $x > a_m$
- At the first iteration the size of the list is 2^k ; after the first iteration it is 2^{k-1} . Then 2^{k-2} and so on until the size of the list is $2^1 = 2$
- At the last step, a comparison tells us that the size of the list is $2^0 = 1$ and the element is compared with the single remaining element

Worst-Case complexity of binary search

- Assume (for simplicity) $n = 2^k$; so $k = \log_2 n$
- Two comparisons are made at each stage $i < j$ and $x > a_m$
- At the first iteration the size of the list is 2^k ; after the first iteration it is 2^{k-1} . Then 2^{k-2} and so on until the size of the list is $2^1 = 2$
- At the last step, a comparison tells us that the size of the list is $2^0 = 1$ and the element is compared with the single remaining element
- Hence, at most $2k + 2 = 2\log_2 n + 2$ comparisons are made

Worst-Case complexity of binary search

- Assume (for simplicity) $n = 2^k$; so $k = \log_2 n$
- Two comparisons are made at each stage $i < j$ and $x > a_m$
- At the first iteration the size of the list is 2^k ; after the first iteration it is 2^{k-1} . Then 2^{k-2} and so on until the size of the list is $2^1 = 2$
- At the last step, a comparison tells us that the size of the list is $2^0 = 1$ and the element is compared with the single remaining element
- Hence, at most $2k + 2 = 2\log_2 n + 2$ comparisons are made
- This means that complexity is $\Theta(\log n)$

Computer time

TABLE 2 The Computer Time Used by Algorithms.

Problem Size	Bit Operations Used					
n	$\log n$	n	$n \log n$	n^2	2^n	$n!$
10	3×10^{-9} sec	10^{-8} sec	3×10^{-8} sec	10^{-7} sec	10^{-6} sec	3×10^{-3} sec
10^2	7×10^{-9} sec	10^{-7} sec	7×10^{-7} sec	10^{-5} sec	4×10^{13} yr	*
10^3	1.0×10^{-8} sec	10^{-6} sec	1×10^{-5} sec	10^{-3} sec	*	*
10^4	1.3×10^{-8} sec	10^{-5} sec	1×10^{-4} sec	10^{-1} sec	*	*
10^5	1.7×10^{-8} sec	10^{-4} sec	2×10^{-3} sec	10 sec	*	*
10^6	2×10^{-8} sec	10^{-3} sec	2×10^{-2} sec	17 min	*	*

However, the time required for an algorithm to solve a problem of a specified size can be determined if all operations can be reduced to the bit operations used by the computer. Table 2 displays the time needed to solve problems of various sizes with an algorithm using the indicated number of bit operations. Times of more than 10^{100} years are indicated with an asterisk. (In Section 2.4 the number of bit operations

15/12/2009

Further topics

- An algorithm is polynomial time if for some k it is $\Theta(n^k)$

Further topics

- An algorithm is polynomial time if for some k it is $\Theta(n^k)$
- Tractable problem: there is a polynomial time algorithm that solves it. (Class P is tractable problems)

Further topics

- An algorithm is polynomial time if for some k it is $\Theta(n^k)$
- Tractable problem: there is a polynomial time algorithm that solves it. (Class P is tractable problems)
- Intractable problem: there is no polynomial time algorithm that solves it

Further topics

- An algorithm is polynomial time if for some k it is $\Theta(n^k)$
- Tractable problem: there is a polynomial time algorithm that solves it. (Class P is tractable problems)
- Intractable problem: there is no polynomial time algorithm that solves it
- Class NP with $P \subseteq NP$ and which has complete problems such as satisfiability of boolean formulas

Further topics

- An algorithm is polynomial time if for some k it is $\Theta(n^k)$
- Tractable problem: there is a polynomial time algorithm that solves it. (Class P is tractable problems)
- Intractable problem: there is no polynomial time algorithm that solves it
- Class NP with $P \subseteq NP$ and which has complete problems such as satisfiability of boolean formulas
- Open problem: $NP \subseteq P$?

Further topics

- An algorithm is polynomial time if for some k it is $\Theta(n^k)$
- Tractable problem: there is a polynomial time algorithm that solves it. (Class P is tractable problems)
- Intractable problem: there is no polynomial time algorithm that solves it
- Class NP with $P \subseteq NP$ and which has complete problems such as satisfiability of boolean formulas
- Open problem: $NP \subseteq P ?$
- If there is a polynomial time algorithm for any NP complete problem then $P = NP$

Further topics

- An algorithm is polynomial time if for some k it is $\Theta(n^k)$
- Tractable problem: there is a polynomial time algorithm that solves it. (Class P is tractable problems)
- Intractable problem: there is no polynomial time algorithm that solves it
- Class NP with $P \subseteq NP$ and which has complete problems such as satisfiability of boolean formulas
- Open problem: $NP \subseteq P ?$
- If there is a polynomial time algorithm for any NP complete problem then $P = NP$
- There are quick algorithms for testing whether a large integer is prime $O((\log n)^6)$

Further topics

- An algorithm is polynomial time if for some k it is $\Theta(n^k)$
- Tractable problem: there is a polynomial time algorithm that solves it. (Class P is tractable problems)
- Intractable problem: there is no polynomial time algorithm that solves it
- Class NP with $P \subseteq NP$ and which has complete problems such as satisfiability of boolean formulas
- Open problem: $NP \subseteq P ?$
- If there is a polynomial time algorithm for any NP complete problem then $P = NP$
- There are quick algorithms for testing whether a large integer is prime $O((\log n)^6)$
- How hard is it to factorise integers?

Further topics

- An algorithm is polynomial time if for some k it is $\Theta(n^k)$
- Tractable problem: there is a polynomial time algorithm that solves it. (Class P is tractable problems)
- Intractable problem: there is no polynomial time algorithm that solves it
- Class NP with $P \subseteq NP$ and which has complete problems such as satisfiability of boolean formulas
- Open problem: $NP \subseteq P ?$
- If there is a polynomial time algorithm for any NP complete problem then $P = NP$
- There are quick algorithms for testing whether a large integer is prime $O((\log n)^6)$
- How hard is it to factorise integers?
- We don't know if it belongs to P (it is in NP)

Further topics

- An algorithm is polynomial time if for some k it is $\Theta(n^k)$
- Tractable problem: there is a polynomial time algorithm that solves it. (Class P is tractable problems)
- Intractable problem: there is no polynomial time algorithm that solves it
- Class NP with $P \subseteq NP$ and which has complete problems such as satisfiability of boolean formulas
- Open problem: $NP \subseteq P ?$
- If there is a polynomial time algorithm for any NP complete problem then $P = NP$
- There are quick algorithms for testing whether a large integer is prime $O((\log n)^6)$
- How hard is it to factorise integers?
- We don't know if it belongs to P (it is in NP)
- It is very unlikely to be NP complete

Discrete Mathematics & Mathematical Reasoning

Chapter 6: Counting

Kousha Etessami

U. of Edinburgh, UK

Chapter Summary

- The Basics of Counting
- The Pigeonhole Principle
- Permutations and Combinations
- Binomial Coefficients and Identities
- Generalized Permutations and Combinations

Basic Counting: The Product Rule

Recall: For a set A , $|A|$ is the **cardinality** of A (# of elements of A).

For a pair of sets A and B , $A \times B$ denotes their **cartesian product**:

$$A \times B = \{(a, b) \mid a \in A \wedge b \in B\}$$

Product Rule

If A and B are finite sets, then: $|A \times B| = |A| \cdot |B|$.

Proof: Obvious, but prove it yourself by induction on $|A|$. □

general Product Rule

If A_1, A_2, \dots, A_m are finite sets, then

$$|A_1 \times A_2 \times \dots \times A_m| = |A_1| \cdot |A_2| \cdot \dots \cdot |A_m|$$

Proof: By induction on m , using the (basic) product rule. □

Product Rule: examples

Example 1: How many bit strings of length seven are there?

Product Rule: examples

Example 1: How many bit strings of length seven are there?

Solution: Since each bit is either 0 or 1, applying the product rule, the answer is $2^7 = 128$.



Product Rule: examples

Example 1: How many bit strings of length seven are there?

Solution: Since each bit is either 0 or 1, applying the product rule, the answer is $2^7 = 128$. □

Example 2: How many different car license plates can be made if each plate contains a sequence of three uppercase English letters followed by three digits?

Product Rule: examples

Example 1: How many bit strings of length seven are there?

Solution: Since each bit is either 0 or 1, applying the product rule, the answer is $2^7 = 128$. □

Example 2: How many different car license plates can be made if each plate contains a sequence of three uppercase English letters followed by three digits?

Solution: $26 \cdot 26 \cdot 26 \cdot 10 \cdot 10 \cdot 10 = 17,576,000$. □

Counting Subsets

Number of Subsets of a Finite Set

A finite set, S , has $2^{|S|}$ distinct subsets.

Proof: Suppose $S = \{s_1, s_2, \dots, s_m\}$.

There is a one-to-one correspondence (bijection), between subsets of S and bit strings of length $m = |S|$.

The bit string of length $|S|$ we associate with a subset $A \subseteq S$ has a 1 in position i if $s_i \in A$, and 0 in position i if $s_i \notin A$, for all $i \in \{1, \dots, m\}$.

$$\{s_2, s_4, s_5, \dots, s_m\} \cong \underbrace{\begin{array}{ccccccc} 0 & 1 & 0 & 1 & 1 & \dots & 1 \end{array}}_m$$

By the product rule, there are $2^{|S|}$ such bit strings. □

Counting Functions

Number of Functions

For all finite sets A and B , the number of distinct functions, $f : A \rightarrow B$, mapping A to B is:

$$|B|^{|A|}$$

Proof: Suppose $A = \{a_1, \dots, a_m\}$.

There is a one-to-one correspondence between functions $f : A \rightarrow B$ and strings (sequences) of length $m = |A|$ over an alphabet of size $n = |B|$:

$$(f : A \rightarrow B) \quad \cong \quad \boxed{f(a_1) \mid f(a_2) \mid f(a_3) \mid \dots \mid f(a_m)}$$

By the product rule, there are n^m such strings of length m . □

Sum Rule

Sum Rule

If A and B are finite sets that are disjoint (meaning $A \cap B = \emptyset$), then

$$|A \cup B| = |A| + |B|$$

Proof. Obvious. (If you must, prove it yourself by induction on $|A|$). □

general Sum Rule

If A_1, \dots, A_m are finite sets that are pairwise disjoint, meaning $A_i \cap A_j = \emptyset$, for all $i, j \in \{1, \dots, m\}$, then

$$|A_1 \cup A_2 \cup \dots \cup A_m| = |A_1| + |A_2| + \dots + |A_m|$$

Sum Rule: Examples

Example 1: Suppose variable names in a programming language can be either a single uppercase letter or an uppercase letter followed by a digit. Find the number of possible variable names.

Sum Rule: Examples

Example 1: Suppose variable names in a programming language can be either a single uppercase letter or an uppercase letter followed by a digit. Find the number of possible variable names.

Solution: Use the sum and product rules: $26 + 26 \cdot 10 = 286$.



Sum Rule: Examples

Example 1: Suppose variable names in a programming language can be either a single uppercase letter or an uppercase letter followed by a digit. Find the number of possible variable names.

Solution: Use the sum and product rules: $26 + 26 \cdot 10 = 286$. □

Example 2: Each user on a computer system has a password which must be six to eight characters long.

Each character is an uppercase letter or digit.

Each password must contain at least one digit.

How many possible passwords are there?

Sum Rule: Examples

Example 1: Suppose variable names in a programming language can be either a single uppercase letter or an uppercase letter followed by a digit. Find the number of possible variable names.

Solution: Use the sum and product rules: $26 + 26 \cdot 10 = 286$. □

Example 2: Each user on a computer system has a password which must be six to eight characters long.

Each character is an uppercase letter or digit.

Each password must contain at least one digit.

How many possible passwords are there?

Solution: Let P be the total number of passwords, and let P_6, P_7, P_8 be the number of passwords of lengths 6, 7, and 8, respectively.

- By the sum rule $P = P_6 + P_7 + P_8$.
- $P_6 = 36^6 - 26^6$; $P_7 = 36^7 - 26^7$; $P_8 = 36^8 - 26^8$.
- So, $P = P_6 + P_7 + P_8 = \sum_{i=6}^8 (36^i - 26^i)$.

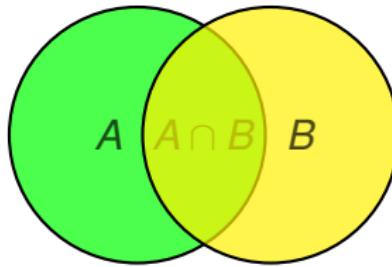
Subtraction Rule (Inclusion-Exclusion for two sets)

Subtraction Rule

For any finite sets A and B (not necessarily disjoint),

$$|A \cup B| = |A| + |B| - |A \cap B|$$

Proof: Venn Diagram:



$|A| + |B|$ overcounts (twice) exactly those elements in $A \cap B$. □

Subtraction Rule: Example

Example: How many bit strings of length 8 either start with a 1 bit or end with the two bits 00?

Subtraction Rule: Example

Example: How many bit strings of length 8 either start with a 1 bit or end with the two bits 00?

Solution:

- Number of bit strings of length 8 that start with 1: $2^7 = 128$.
- Number of bit strings of length 8 that end with 00: $2^6 = 64$.
- Number of bit strings of length 8 that start with 1 and end with 00: $2^5 = 32$.

Applying the subtraction rule, the number is $128 + 64 - 32 = 160$. □

The Pigeonhole Principle

Pigeonhole Principle

For any positive integer k , if $k + 1$ objects (pigeons) are placed in k boxes (pigeonholes), then at least one box contains two or more objects.

Proof: Suppose no box has more than 1 object. Sum up the number of objects in the k boxes. There can't be more than k .
Contradiction. □

Pigeonhole Principle (rephrased more formally)

If a function $f : A \rightarrow B$ maps a finite set A with $|A| = k + 1$ to a finite set B , with $|B| = k$, then f is **not** one-to-one.

(Recall: a function $f : A \rightarrow B$ is called **one-to-one** if $\forall a_1, a_2 \in A$, if $a_1 \neq a_2$ then $f(a_1) \neq f(a_2)$.)

Pigeonhole Principle: Examples

Example 1: At least two students registered for this course will receive **exactly the same** final exam mark. Why?

Pigeonhole Principle: Examples

Example 1: At least two students registered for this course will receive **exactly the same** final exam mark. Why?

Reason: There are at least 102 students registered for DMMR (suppose the actual number is 145), so, at least 102 objects. Final exam marks are integers in the range 0-100 (so, exactly 101 boxes). □

Generalized Pigeonhole Principle

Generalized Pigeonhole Principle (GPP)

If $N \geq 0$ objects are placed in $k \geq 1$ boxes, then at least one box contains at least $\lceil \frac{N}{k} \rceil$ objects.

Proof: Suppose no box has more than $\lceil \frac{N}{k} \rceil - 1$ objects. Sum up the number of objects in the k boxes. It is at most

$$k \cdot (\left\lceil \frac{N}{k} \right\rceil - 1) < k \cdot \left(\frac{N}{k} + 1 \right) - 1 = N$$

Thus, there must be fewer than N . Contradiction.

(We are using the fact that $\lceil \frac{N}{k} \rceil < \frac{N}{k} + 1$.)



Exercise: Rephrase GPP as a statement about functions $f : A \rightarrow B$ that map a finite set A with $|A| = N$ to a finite set B , with $|B| = k$.

Generalized Pigeonhole Principle: Examples

Example 1: Consider the following statement:

"At least d students in this course were born in the same month." (1)

Suppose the actual number of students registered for DMMR is 145.
What is the maximum number d for which **it is certain** that statement (1) is true?

Generalized Pigeonhole Principle: Examples

Example 1: Consider the following statement:

"At least d students in this course were born in the same month." (1)

Suppose the actual number of students registered for DMMR is 145. What is the maximum number d for which it is certain that statement (1) is true?

Solution: Since we are assuming there are 145 registered students in DMMR.

$$\lceil \frac{145}{12} \rceil = 13, \text{ so by GPP we know statement (1) is true for } d = 13.$$

Statement (1) need not be true for $d = 14$, because if 145 students are distributed *as evenly as possible* into 12 months, the maximum number of students in any month is 13, with other months having only 12. □

Generalized Pigeonhole Principle: Examples

Example 1: Consider the following statement:

"At least d students in this course were born in the same month." (1)

Suppose the actual number of students registered for DMMR is 145. What is the maximum number d for which **it is certain** that statement (1) is true?

Solution: Since we are assuming there are 145 registered students in DMMR.

$\lceil \frac{145}{12} \rceil = 13$, so by GPP we know statement (1) is true for $d = 13$.

Statement (1) need not be true for $d = 14$, because if 145 students are distributed **as evenly as possible** into 12 months, the maximum number of students in any month is 13, with other months having only 12. □

(In **probability theory** you will learn that nevertheless **it is highly probable**, assuming birthdays are **randomly** distributed, that at least 14 of you (and more) were indeed born in the same month.)

GPP: more Examples

Example 2: How many cards must be selected from a standard deck of 52 cards to guarantee that at least three cards of the same suit are chosen?

GPP: more Examples

Example 2: How many cards must be selected from a standard deck of 52 cards to guarantee that at least three cards of the same suit are chosen?

Solution: There are 4 suits. (In a standard deck of 52 cards, every card has exactly one suit. There are no jokers.) So, we need to choose N cards, such that $\lceil \frac{N}{4} \rceil \geq 3$. The smallest integer N such that $\lceil \frac{N}{4} \rceil \geq 3$ is $2 \cdot 4 + 1 = 9$.



Permutations

Permutation

A **permutation** of a set S is an ordered arrangement of the elements of S .

In other words, it is a sequence containing every element of S exactly once.

Example: Consider the set $S = \{1, 2, 3\}$.

The sequence $(3, 1, 2)$ is one permutation of S .

There are 6 different permutations of S . They are:

$(1, 2, 3)$, $(1, 3, 2)$, $(2, 1, 3)$, $(2, 3, 1)$, $(3, 1, 2)$, $(3, 2, 1)$

Permutations (an alternative view)

A permutation of a set S can alternatively be viewed as a **bijection (a one-to-one and onto function)**, $\pi : S \rightarrow S$, from S to itself.

Specifically, if the finite set is $S = \{s_1, \dots, s_m\}$, then by fixing the ordering s_1, \dots, s_m , we can **uniquely** associate to each bijection $\pi : S \rightarrow S$ a sequence ordering $\{\pi(s_1), \dots, \pi(s_m)\}$ as follows:

$$(\pi : S \rightarrow S) \quad \cong \quad \boxed{\pi(s_1) \mid \pi(s_2) \mid \pi(s_3) \mid \dots \mid \pi(s_m)}$$

Note that π is a bijection **if and only if** the sequence on the right containing every element of S exactly once.

r -Permutation

r -Permutation

An **r -permutation** of a set S , is an ordered arrangement (sequence) of r distinct elements of S .

(For this to be well-defined, r needs to be an integer with $0 \leq r \leq |S|$.)

Examples:

There is only one 0-permutation of any set: the empty sequence () .

For the set $S = \{1, 2, 3\}$, the sequence (3, 1) is a 2-permutation.

(3, 2, 1) is both a permutation and 3-permutation of S (since $|S| = 3$).

There are 6 different different 2-permutations of S . They are:

$$(1, 2), (1, 3), (2, 1), (2, 3), (3, 1), (3, 2)$$

Question: How many r -permutations of an n -element set are there?

r -Permutations (an alternative view)

An r -permutation of a set S , with $1 \leq r \leq |S|$, can alternatively be viewed as a **one-to-one function**, $f : \{1, \dots, r\} \rightarrow S$.

Specifically, we can uniquely associate to each one-to-one function $f : \{1, \dots, r\} \rightarrow S$, an r -permutation of S as follows:

$$(f : \{1, \dots, r\} \rightarrow S) \quad \cong \quad \boxed{f(1) \mid f(2) \mid f(3) \mid \dots \mid f(r)}$$

Note that f is one-to-one **if and only if** the sequence on the right is an r -permutation of S .

So, for a set S with $|S| = n$, the number of r -permutations of S , $1 \leq r \leq n$, is equal to the number of **one-to-one functions**:

$$f : \{1, \dots, r\} \rightarrow \{1, \dots, n\}$$

Formula for # of permutations, and # of r -permutations

Let $P(n, r)$ denote the number of **r -permutations** of an n -element set.

$P(n, 0) = 1$, because the only 0-permutation is the empty sequence.

Theorem

For all integers $n \geq 1$, and all integers r such that $1 \leq r \leq n$:

$$P(n, r) = n \cdot (n - 1) \cdot (n - 2) \dots (n - r + 1) = \frac{n!}{(n - r)!}$$

Proof. There are n different choices for the first element of the sequence. For each of those choices, there are $n - 1$ remaining choices for the second element. For every combination of the first two choices, there are $n - 2$ choices for the third element, and so forth. \square

Corollary: the number of **permutations** of an n element set is:

$$n! = n \cdot (n - 1) \cdot (n - 2) \dots 2 \cdot 1 = P(n, n)$$

Example: a simple counting problem

Example: How many permutations of the letters ABCDEFGH contain the string ABC as a (consecutive) substring?

Example: a simple counting problem

Example: How many permutations of the letters ABCDEFGH contain the string ABC as a (consecutive) substring?

Solution: We solve this by noting that this number is the same as the number of permutations of the following **six** objects:
ABC, D, E, F, G, and H. So the answer is:

$$6! = 720.$$



How big is $n!$?

The factorial function, $n!$, is fundamental in combinatorics and discrete maths. So it is important to get a good handle on how fast $n!$ grows.

Questions:

Which is bigger $n!$ or 2^n ?

Which is bigger $n!$ or n^n ?

How big is $n!$?

The factorial function, $n!$, is fundamental in combinatorics and discrete maths. So it is important to get a good handle on how fast $n!$ grows.

Questions:

Which is bigger $n!$ or 2^n ?

Which is bigger $n!$ or n^n ?

Answers (easy)

- ① $n! \leq n^n$, for all $n \geq 0$. (Note $0^0 = 1$ and $0! = 1$, by definition.)
- ② $2^n < n!$, for all $n \geq 4$.

So, $2^n \leq n! \leq n^n$, but that's a **big gap** between growth 2^n and n^n .

Question: Is there a really good formula for approximating $n!$?

How big is $n!$?

The factorial function, $n!$, is fundamental in combinatorics and discrete maths. So it is important to get a good handle on how fast $n!$ grows.

Questions:

Which is bigger $n!$ or 2^n ?

Which is bigger $n!$ or n^n ?

Answers (easy)

- ① $n! \leq n^n$, for all $n \geq 0$. (Note $0^0 = 1$ and $0! = 1$, by definition.)
- ② $2^n < n!$, for all $n \geq 4$.

So, $2^n \leq n! \leq n^n$, but that's a **big gap** between growth 2^n and n^n .

Question: Is there a really good formula for approximating $n!$?

Yes! A brilliant Scottish mathematician discovered it in 1730!



Grave of **James Stirling** (1692-1770), in **Greyfriar's kirkyard**, Edinburgh.

Stirling's Approximation Formula

Stirling's approximation formula

$$n! \sim \sqrt{2\pi n} \cdot \left(\frac{n}{e}\right)^n$$

In other words: $\lim_{n \rightarrow \infty} \frac{n!}{\sqrt{2\pi n} \cdot \left(\frac{n}{e}\right)^n} = 1$.

(e denotes the base of the natural logarithm.)

Unfortunately, we won't prove this. (The proof needs calculus.)

It is often more useful to have explicit lower and upper bounds on $n!$:

Stirling's approximation (with lower and upper bounds)

For all $n \geq 1$,

$$\sqrt{2\pi n} \cdot \left(\frac{n}{e}\right)^n \cdot e^{\frac{1}{12n+1}} \leq n! \leq \sqrt{2\pi n} \cdot \left(\frac{n}{e}\right)^n \cdot e^{\frac{1}{12n}}$$

For a proof of this see, e.g., [Feller, Vol.1, 1968].

Combinations

r -Combinations

An **r -combination** of a set S is an **unordered** collection of r elements of S . In other words, it is simply a subset of S of size r .

Example: Consider the set $S = \{1, 2, 3, 4, 5\}$.

The set $\{2, 5\}$ is a 2-combination of S .

There are 10 different 2-combinations of S . They are:

$\{1, 2\}$, $\{1, 3\}$, $\{1, 4\}$, $\{1, 5\}$,
 $\{2, 3\}$, $\{2, 4\}$, $\{2, 5\}$,
 $\{3, 4\}$, $\{3, 5\}$,
 $\{4, 5\}$

Question: How many r -combinations of an n -element set are there?

Formula for the number of r -combinations

Let $\mathbf{C}(n, r)$ denote the number of r -combinations of an n -element set.
Another notation for $C(n, r)$ is:

$$\binom{n}{r}$$

These are called **binomial coefficients**, and are read as “ n choose r ”.

Theorem

For all integers $n \geq 1$, and all integers r such that $0 \leq r \leq n$:

$$C(n, r) \doteq \binom{n}{r} = \frac{n!}{r! \cdot (n-r)!} = \frac{n \cdot (n-1) \cdot \dots \cdot (n-r+1)}{r!}$$

Proof. We can see that $P(n, r) = C(n, r) \cdot P(r, r)$. (To get an r -permutation: first choose r elements, then order them.) Thus

$$C(n, r) = \frac{P(n, r)}{P(r, r)} = \frac{n!/(n-r)!}{r!/(r-r)!} = \frac{n!}{r! \cdot (n-r)!}$$



Some simple approximations and bounds for $\binom{n}{r}$

Using basic considerations and Stirling's approximation formula, one can easily establish the following bounds and approximations for $\binom{n}{r}$:

$$\left(\frac{n}{r}\right)^r \leq \binom{n}{r} \leq \left(\frac{n \cdot e}{r}\right)^r$$

$$\binom{2n}{n} \sim \frac{2^{2n}}{\sqrt{\pi n}}$$

$$\frac{2^{2n}}{2n+1} \leq \binom{2n}{n} \leq 2^{2n}$$

Combinations: examples

Example:

- ① How many different 5-card poker hands can be dealt from a deck of 52 cards?
- ② How many different 47-card poker hands can be dealt from a deck of 52 cards?

Solutions:

1

$$\binom{52}{5} = \frac{52!}{5! \cdot 47!} = \frac{52 \cdot 51 \cdot 50 \cdot 49 \cdot 48}{5 \cdot 4 \cdot 3 \cdot 2 \cdot 1} = 2,598,960$$

2

$$\binom{52}{47} = \frac{52!}{47! \cdot 5!} = \frac{52 \cdot 51 \cdot 50 \cdot 49 \cdot 48}{5 \cdot 4 \cdot 3 \cdot 2 \cdot 1} = 2,598,960$$

Question: Why are these numbers the same?

Combinations: an identity

Theorem

For all integers $n \geq 1$, and all integers r , $1 \leq r \leq n$:

$$\binom{n}{r} = \binom{n}{n-r}$$

Proof:

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

□

We can also give a **combinatorial proof**: Suppose $|S| = n$. A function, f , that maps each r -element subset A of S to the $(n-r)$ -element subset $(S - A)$ is a **bijection**.

Any two finite sets having a bijection between them must have exactly the same number of elements.

□

Binomial Coefficients

Consider the polynomial in two variables, x and y , given by:

$$(x + y)^n = \underbrace{(x + y) \cdot (x + y) \cdots (x + y)}_n$$

By multiplying out the n terms, we can expand this polynomial and write it in a standard sum-of-monomials form:

$$(x + y)^n = \sum_{j=0}^n c_j x^{n-j} y^j$$

Question: What are the coefficients c_j ? (These are called binomial coefficients.)

Examples:

$$(x + y)^2 = x^2 + 2xy + y^2$$

$$(x + y)^3 = x^3 + 3x^2y + 3xy^2 + y^3$$

The Binomial Theorem

Binomial Theorem

For all $n \geq 0$:

$$(x + y)^n = \sum_{j=0}^n \binom{n}{j} x^{n-j} y^j = \binom{n}{0} x^n + \binom{n}{1} x^{n-1} y + \dots + \binom{n}{n} y^n$$

Proof: What is the coefficient of $x^{n-j} y^j$?

To obtain a term $x^{n-j} y^j$ in the expansion of the product

$$(x + y)^n = \underbrace{(x + y)(x + y) \dots (x + y)}_n$$

we have to choose exactly $n - j$ copies of x and (thus) j copies of y .

How many ways are there to do this? Answer: $\binom{n}{j} = \binom{n}{n-j}$. □

Corollary: $\sum_{j=0}^n \binom{n}{j} = 2^n$.

Proof: By the binomial theorem, $2^n = (1 + 1)^n = \sum_{j=0}^n \binom{n}{j}$.

Pascal's Identity

Theorem (Pascal's Identity)

For all integers $n \geq 0$, and all integers r , $0 \leq r \leq n + 1$:

$$\binom{n+1}{r} = \binom{n}{r-1} + \binom{n}{r}$$

Proof: Suppose $S = \{s_0, s_1, \dots, s_n\}$. We wish to choose a subset $A \subseteq S$ such that $|A| = r$. We can do this in two ways. We can either:

- (I) choose a subset A such that $s_0 \in A$, or
- (II) choose a subset A such that $s_0 \notin A$.

There are $\binom{n}{r-1}$ sets of the first kind,
and there are $\binom{n}{r}$ sets of the second kind.

So, $\binom{n+1}{r} = \binom{n}{r-1} + \binom{n}{r}$.



Pascal's Triangle

Copyright ©The McGraw-Hill Companies, Inc. Permission required for reproduction or display.
Kenneth H. Rosen, *Discrete Mathematics and Its Applications*, 7e

$\binom{0}{0}$		1							
$\binom{1}{0}$	$\binom{1}{1}$	1 1							
$\binom{2}{0}$	$\binom{2}{1}$	$\binom{2}{2}$	By Pascal's identity:	1 2 1					
$\binom{3}{0}$	$\binom{3}{1}$	$\binom{3}{2}$	$\binom{3}{3}$	$\binom{6}{4} + \binom{6}{5} = \binom{7}{5}$	1 3 3 1				
$\binom{4}{0}$	$\binom{4}{1}$	$\binom{4}{2}$	$\binom{4}{3}$	$\binom{4}{4}$	1 4 6 4 1				
$\binom{5}{0}$	$\binom{5}{1}$	$\binom{5}{2}$	$\binom{5}{3}$	$\binom{5}{4}$	$\binom{5}{5}$	1 5 10 10 5 1			
$\binom{6}{0}$	$\binom{6}{1}$	$\binom{6}{2}$	$\binom{6}{3}$	$\binom{6}{4}$	$\binom{6}{5}$	$\binom{6}{6}$	1		
$\binom{7}{0}$	$\binom{7}{1}$	$\binom{7}{2}$	$\binom{7}{3}$	$\binom{7}{4}$	$\binom{7}{5}$	$\binom{7}{6}$	$\binom{7}{7}$	1	
$\binom{8}{0}$	$\binom{8}{1}$	$\binom{8}{2}$	$\binom{8}{3}$	$\binom{8}{4}$	$\binom{8}{5}$	$\binom{8}{6}$	$\binom{8}{7}$	$\binom{8}{8}$	1
								...	

Many other useful identities...

Vandermonde's Identity

For $m, n, r \geq 0$, $r \leq m$, and $r \leq n$, we have

$$\binom{m+n}{r} = \sum_{k=0}^r \binom{m}{r-k} \binom{n}{k}$$

Proof: Suppose we have two disjoint sets A and B , with $|A| = m$ and $|B| = n$, and thus $|A \cup B| = m + n$. We want to choose r elements out of $A \cup B$. We can do this by either:

- (1) choosing 0 elements from A and r elements from B , or
- (2) choosing 1 element from A and $r - 1$ elements from B , or

...

- (r) choosing r elements from A and 0 elements from B .

There are $\binom{m}{r-k} \binom{n}{k}$ possible choices of kind (k).

So, in total, there are $\sum_{k=0}^r \binom{m}{r-k} \binom{n}{k}$ r -element subsets of an $(n+m)$ -element set. So $\binom{n+m}{r} = \sum_{k=0}^r \binom{m}{r-k} \binom{n}{k}$.



r -Combinations with repetition (with replaced)

Sometimes, we want to choose r elements **with repetition allowed** from a set of size n . In how many ways can we do this?

Example: How many different ways are there to place 12 colored balls in a bag, when each ball should be either Red, Green, or Blue?

Let us first formally phrase the general problem.

A **multi-set** over a set S is an **unordered** collection (bag) of copies of elements of S **with possible repetition**. The **size** of a multi-set is the number of copies of all elements in it (counting repetitions).

For example, if $S = \{\text{R, G, B}\}$, then the following two multi-sets over S both have size 4:

$$[\text{G, G, B, B}] \quad [\text{R, G, G, B}]$$

Note that *ordering doesn't matter* in multi-sets, so $[\text{R, R, B}] = [\text{R, B, R}]$.

Definition: an r -Combination with repetition (r -comb-w.r.) from a set S is simply a multi-set of size r over S .

Formula for # of r -Combinations with repetition

Theorem

For all integers $n, r \geq 1$, the number of r -combs-w.r. from a set S of size n is:

$$\binom{n+r-1}{r} = \binom{n+r-1}{n-1}$$

Proof: Each r -combination with repetition can be associated uniquely with a string of length $n + r - 1$ consisting of $n - 1$ bars and r stars, and vice versa.

The bars partition the string into n different segments, and the number of stars in each segment denotes the number of copies of the corresponding element of S in the multi-set.

For example, for $S = \{\text{R, G, B, Y}\}$, then with the multiset

$[\text{R, R, B, B}]$ we associate the string $\star\star||\star\star|$

How many strings of length $n + r - 1$ with $n - 1$ bars and r stars are there? Answer: $\binom{n+r-1}{r} = \binom{n+r-1}{n-1}$.



Example

Example

How many different solutions in non-negative integers x_1 , x_2 , and x_3 , does the following equation have?

$$x_1 + x_2 + x_3 = 11$$

Solution: We have to place 11 “pebbles” into three different “bins”, x_1 , x_2 , and x_3 .

This is equivalent to choosing an 11-comb-w.r. from a set of size 3, so the answer is

$$\binom{11+3-1}{11} = \binom{13}{2} = \frac{13 \cdot 12}{2 \cdot 1} = 78.$$



Permutations with indistinguishable objects

Question: How many different strings can be made by reordering the letters of the word “SUCCESS”?

Theorem: The number of permutations of n objects, with n_1 indistinguishable objects of Type 1, n_2 indistinguishable objects of Type 2, ..., and n_k indistinguishable objects of Type k , is:

$$\frac{n!}{n_1! \cdot n_2! \cdots n_k!}$$

Proof: First, the n_1 objects of Type 1 can be placed among the n positions in $\binom{n}{n_1}$ ways. Next, the n_2 objects of Type 2 can be placed in the remaining $n - n_1$ positions in $\binom{n-n_1}{n_2}$ ways, and so on... We get:

$$\binom{n}{n_1} \cdot \binom{n-n_1}{n_2} \cdot \binom{n-n_1-n_2}{n_3} \cdots \binom{n-n_1-n_2-\cdots-n_{k-1}}{n_k} =$$
$$\frac{n!}{n_1!(n-n_1)!} \frac{(n-n_1)!}{n_2!(n-n_1-n_2)!} \cdots \frac{(n-n_1-\cdots-n_{k-1})!}{n_k!0!} = \frac{n!}{n_1!n_2!\cdots n_k!}$$



Multinomial Coefficients

Multinomial coefficients

For integers $n, n_1, n_2, \dots, n_k \geq 0$, such that $n = n_1 + n_2 + \dots + n_k$, let:

$$\binom{n}{n_1, n_2, \dots, n_k} = \frac{n!}{n_1! n_2! \dots n_k!}$$

Multinomial Theorem

For all $n \geq 0$, and all $k \geq 1$:

$$(x_1 + x_2 + \dots + x_k)^n = \sum_{\substack{0 \leq n_1, n_2, \dots, n_k \leq n \\ n_1 + n_2 + \dots + n_k = n}} \binom{n}{n_1, n_2, \dots, n_k} x_1^{n_1} x_2^{n_2} \dots x_k^{n_k}$$

Note: the Binomial Theorem is the special case of this where $k = 2$.

Question: In how many ways can the elements of a set S , $|S| = n$, be partitioned into k distinguishable boxes, such that Box 1 gets n_1 elements, ..., Box k gets n_k elements? **Answer:** $\binom{n}{n_1, n_2, \dots, n_k}$. □

Discrete Mathematics & Mathematical Reasoning

Chapter 10: Graphs

Kousha Etessami

U. of Edinburgh, UK

Overview

- Graphs and Graph Models
- Graph Terminology and Special Types of Graphs
- Representations of Graphs, and Graph Isomorphism
- Connectivity
- Euler and Hamiltonian Paths
- Brief look at other topics like graph coloring

What is a Graph?

Informally, a **graph** consists of a non-empty set of **vertices** (or **nodes**), and a set E of **edges** that connect (pairs of) nodes.

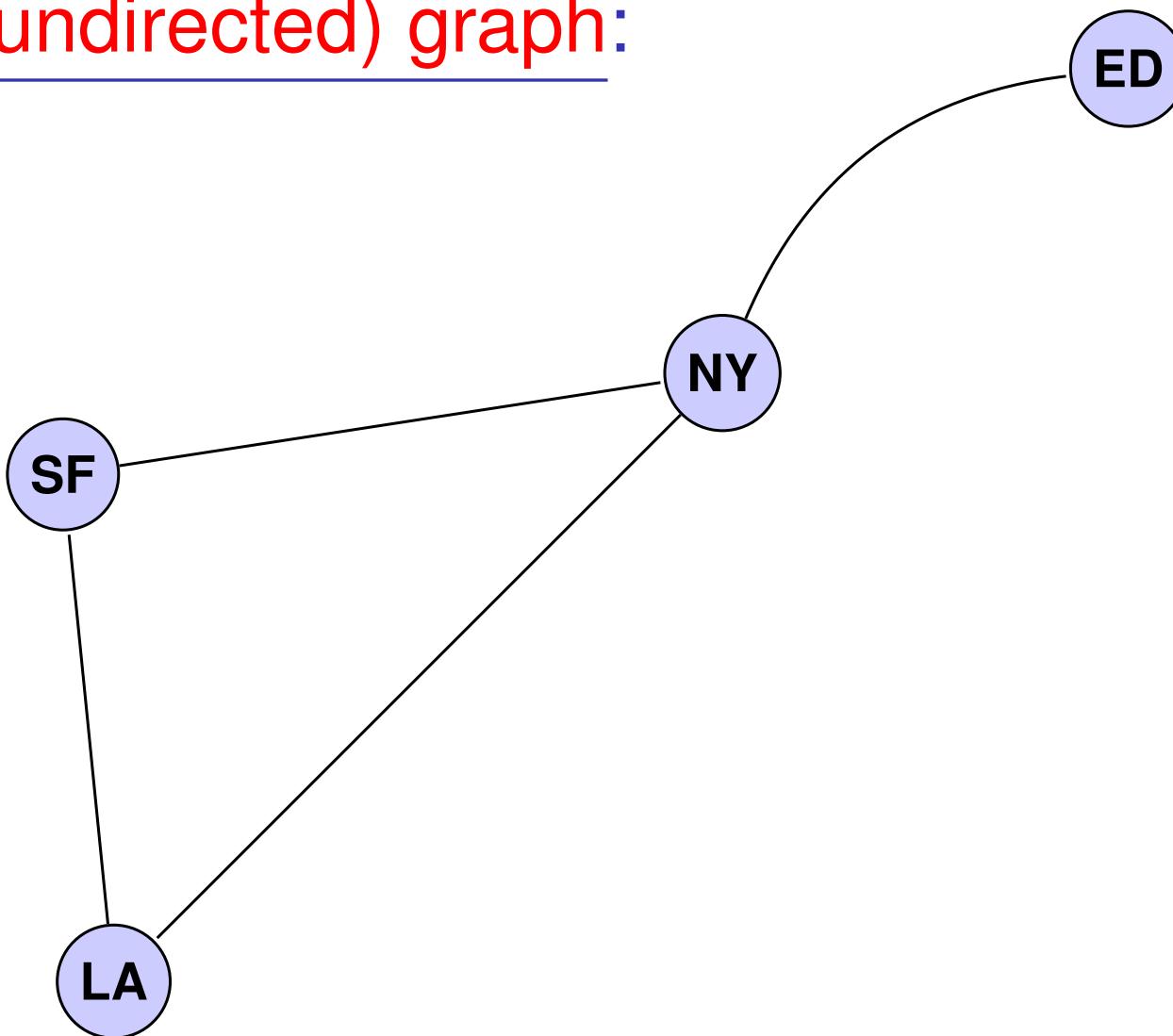
But different types of graphs (*undirected*, *directed*, *simple*, *multigraph*, ...) have different formal definitions, depending on what kinds of edges are allowed.

This creates a lot of (often inconsistent) terminology.

Before formalizing, let's see some examples....

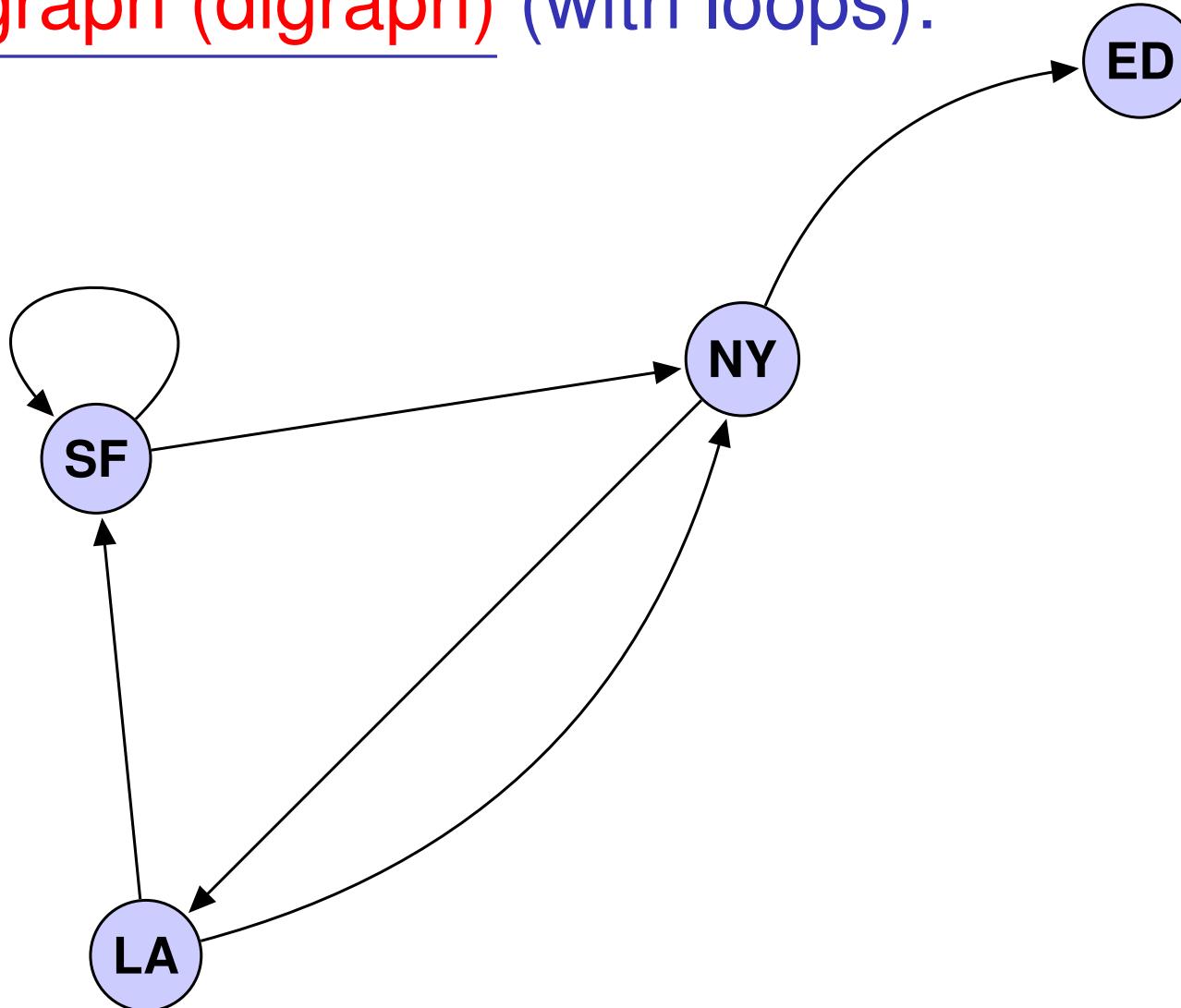
During this course, we focus almost exclusively on standard **(undirected) graphs** and **directed graphs**, which are our first two examples.

A (simple undirected) graph:



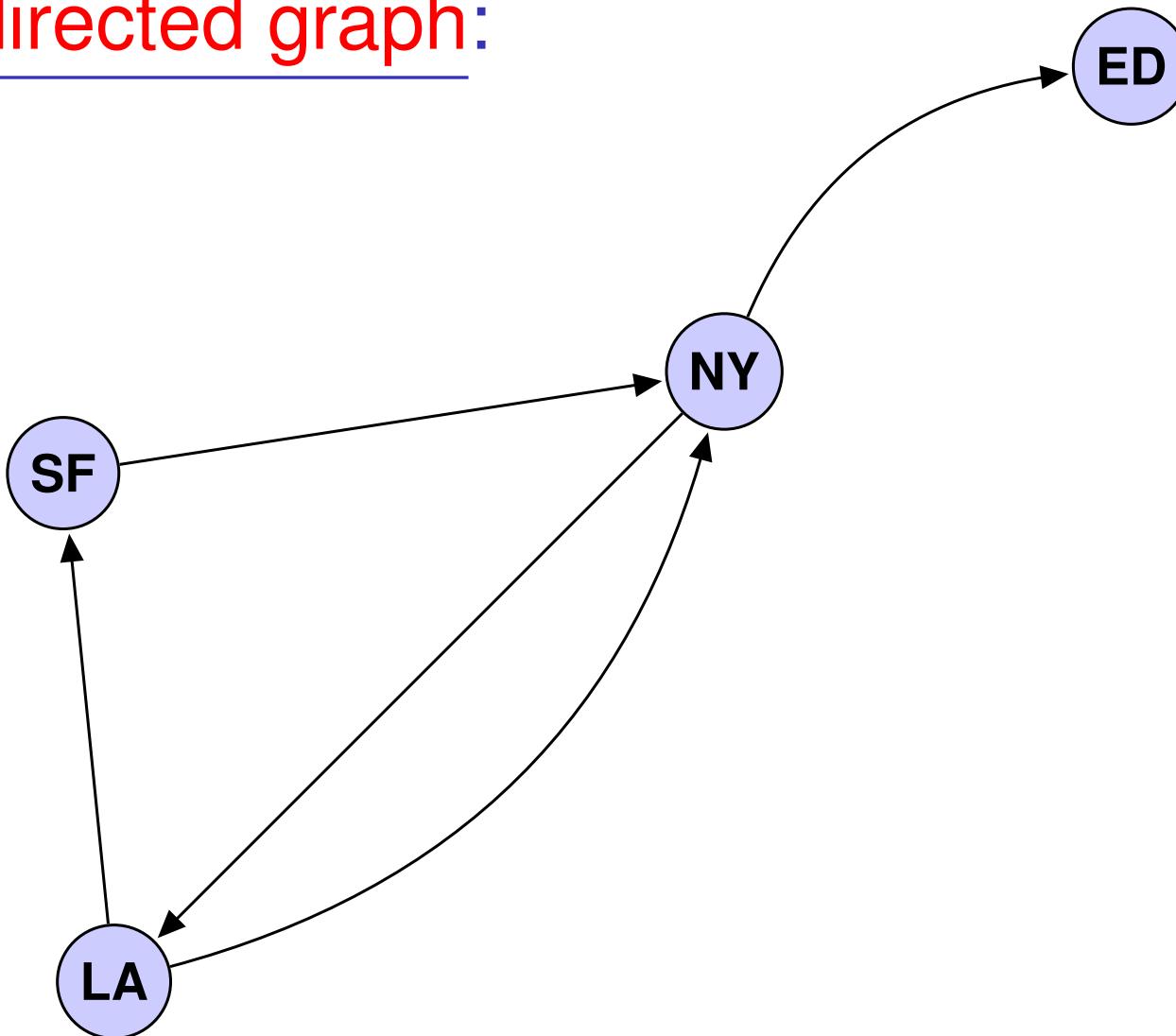
Only **undirected** edges; **at most one edge** between any pair of distinct nodes; and **no loops** (edges between a node and itself).

A directed graph (digraph) (with loops):



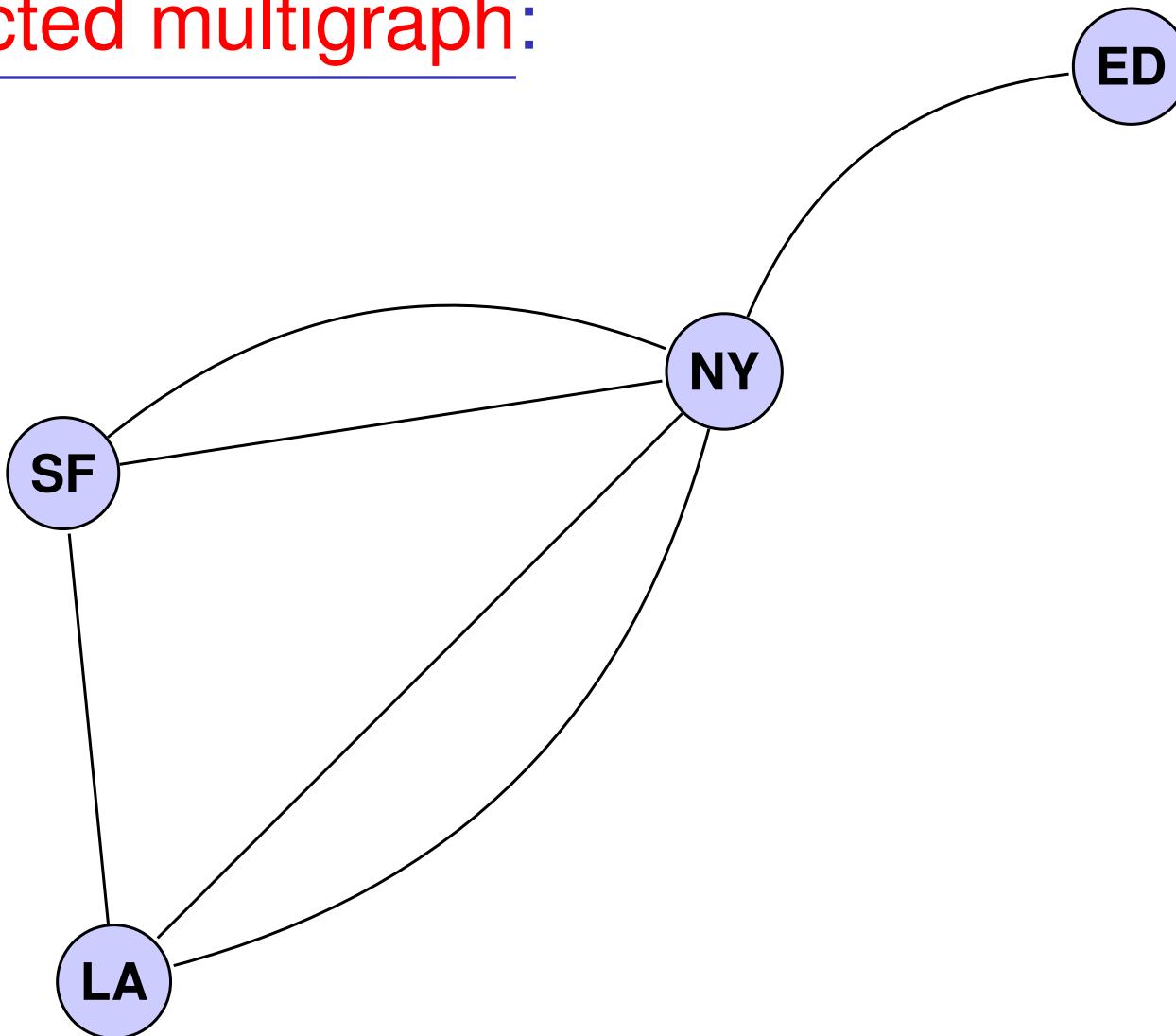
Only **directed** edges; at most one directed edge from any node to any node; and **loops** are allowed.

A simple directed graph:



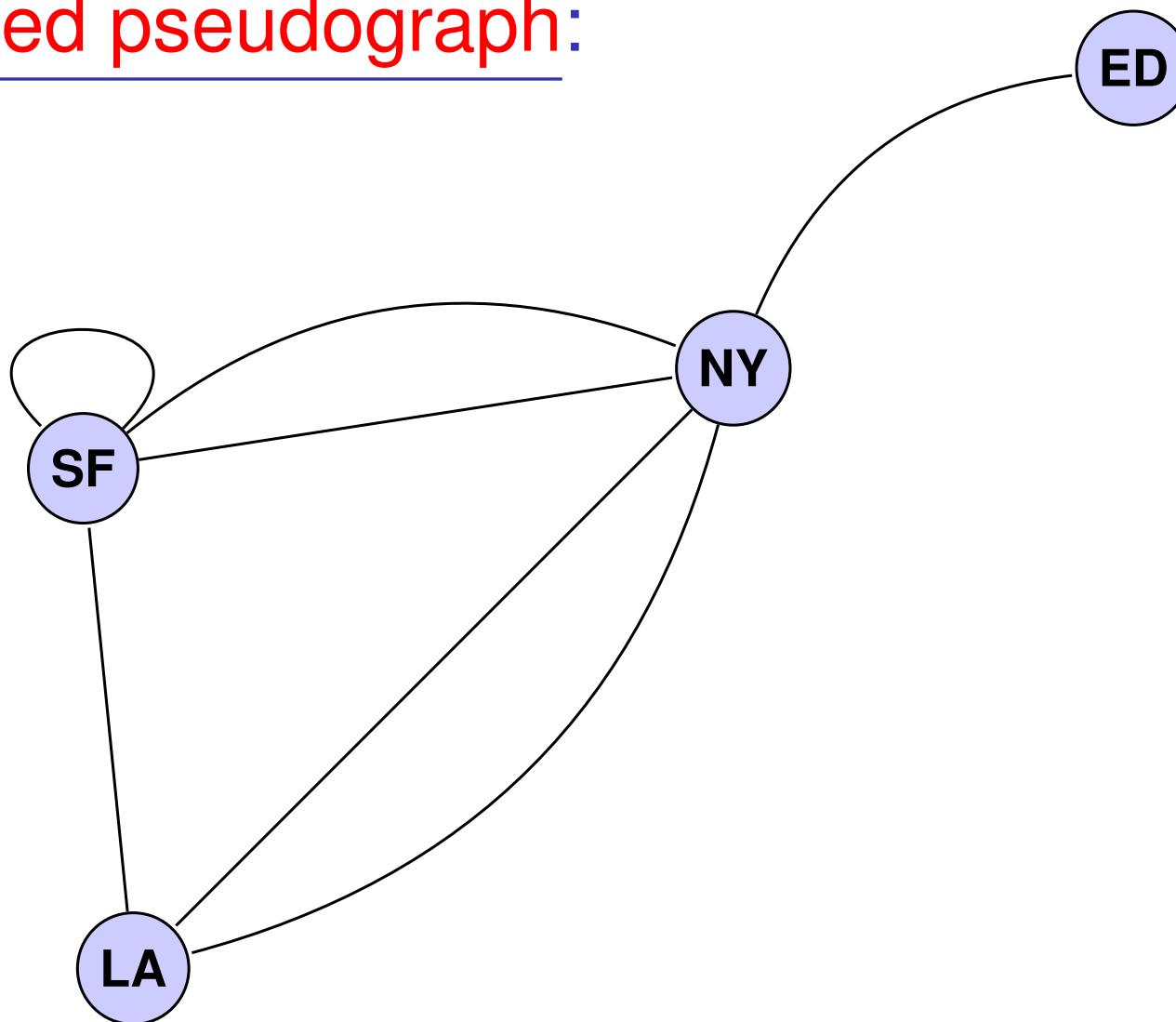
Only **directed** edges; **at most one directed edge** from any node to any other node; and **no loops allowed**.

An undirected multigraph:



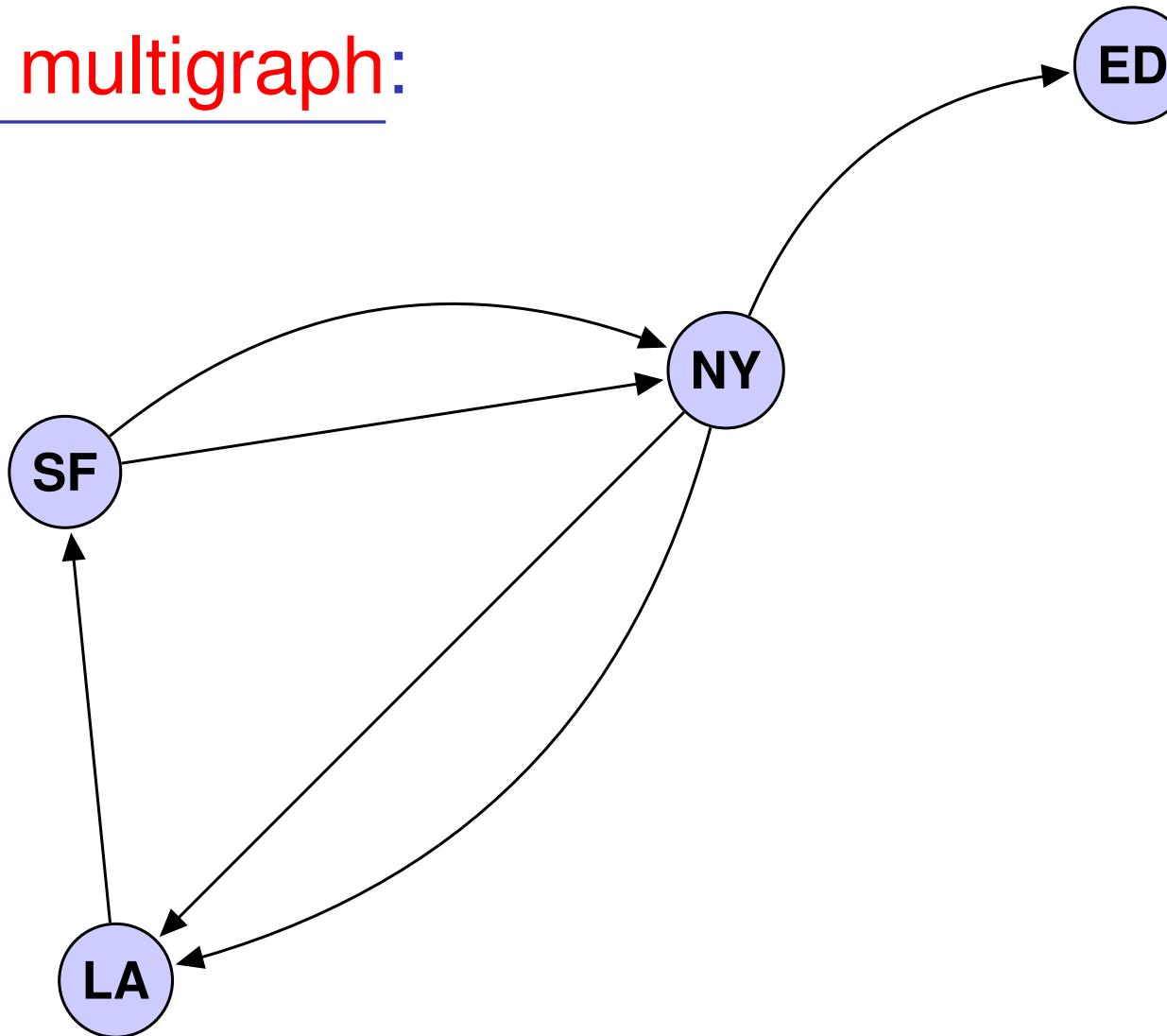
Only **undirected** edges; may contain **multiple edges** between a pair of nodes; but **no loops**.

An undirected pseudograph:



Only **undirected** edges; may contain **multiple edges** between a pair of nodes; and **may contain loops** (even multiple loops on the same node).

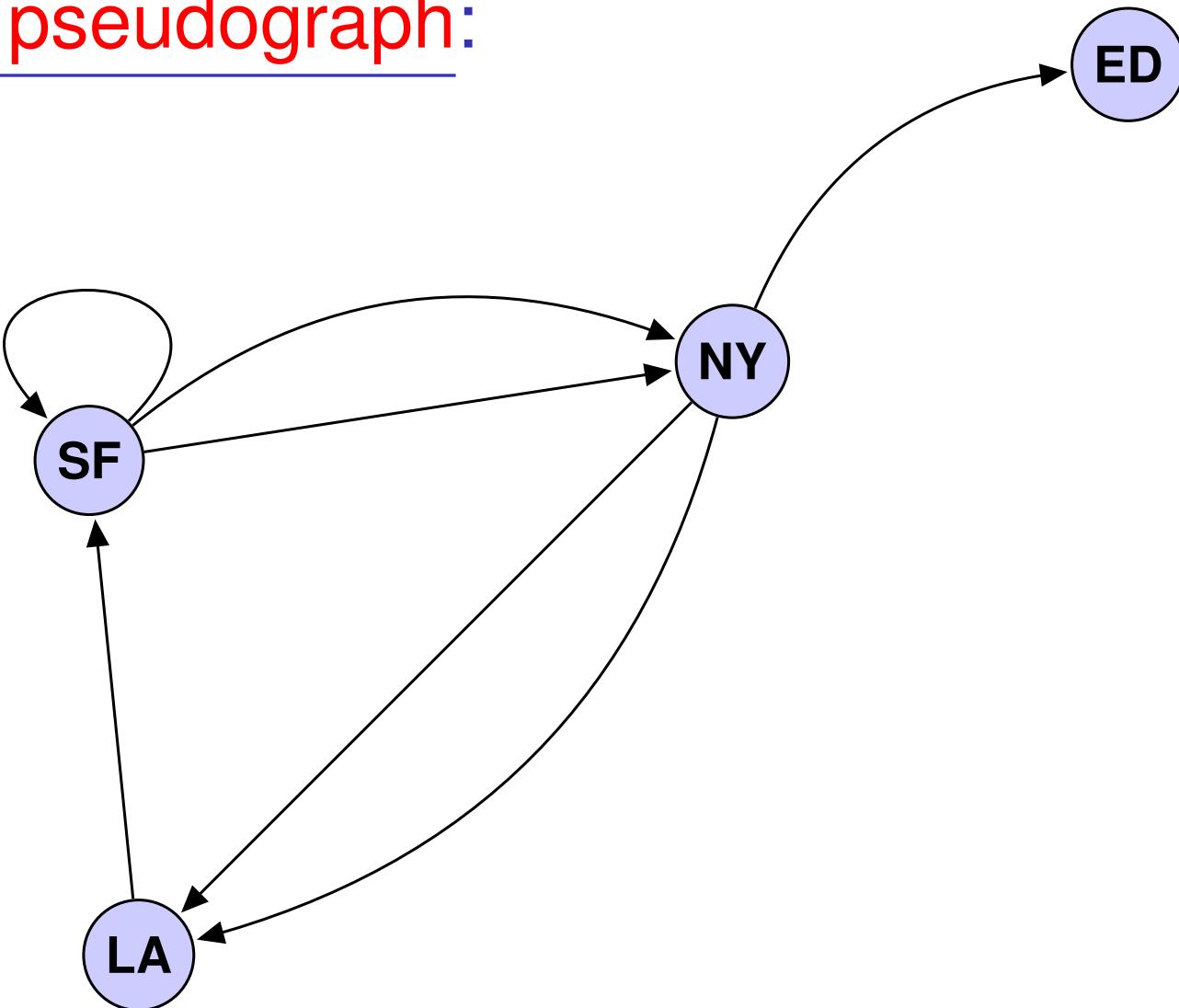
A directed multigraph:



Only **directed** edges; may contain **multiple edges** from one node to another; but **no loops allowed**.

Warning: this differs slightly from the Rosen book terminology. The book's notion of "directed multigraph" would allow loops.

An directed pseudograph:



Only **directed** edges; may contain **multiple edges** from one node to another; and **may contain loops** (even multiple loops on one node).

Graph Terminology Zoo (ridiculous)

	Type	Edges	Multi-Edges?	Loops?
1.	(simple undirected) graph	Undirected	No	No
2.	(undirected) multigraph	Undirected	Yes	No
3.	(undirected) pseudograph	Undirected	Yes	Yes
4.	directed graph	Directed	No	Yes
5.	simple directed graph	Directed	No	No
6.	directed multigraph	Directed	Yes	No ¹
7.	directed pseudograph	Directed	Yes	Yes
8.	mixed graph	Both	Yes	Yes

We will focus on the two most standard types:

- (1.) graphs (simple undirected), and
- (4.) directed graphs (also known as digraphs).

¹differs from book.

Formal Definition of Directed Graphs

A **directed graph (digraph)**, $G = (V, E)$, consists of a non-empty set, V , of **vertices** (or **nodes**), and a set $E \subseteq V \times V$ of **directed edges** (or **arcs**). Each directed edge $(u, v) \in E$ has a **start (tail)** vertex u , and a **end (head)** vertex v .

Note: a directed graph $G = (V, E)$ is simply a set V together with a **binary relation** E on V .

Definition of (Undirected) Graphs

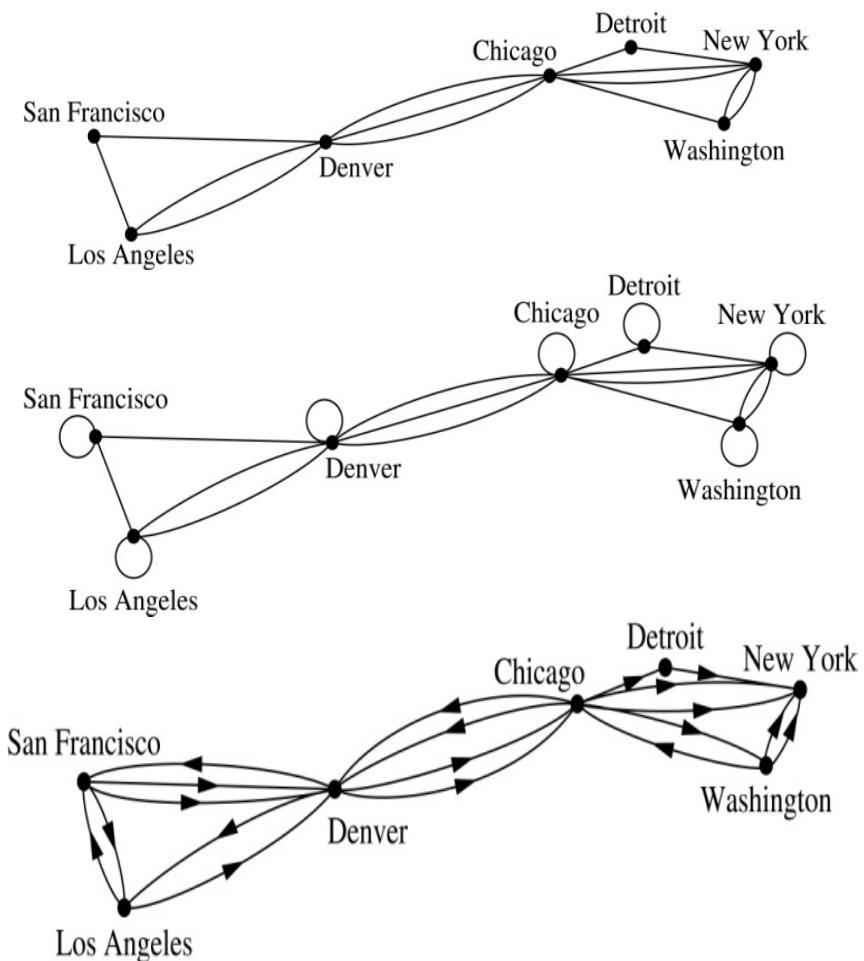
For a set V , let $[V]^k$ denote the set of k -element subsets of V .
(Equivalently, $[V]^k$ is the set of all k -combinations of V .)

A (simple,undirected) **graph**, $G = (V, E)$, consists of a non-empty set V of **vertices** (or **nodes**), and a set $E \subseteq [V]^2$ of (undirected) **edges**. Every edge $\{u, v\} \in E$ has two distinct vertices $u \neq v$ as **endpoints**, and such vertices u and v are then said to be **adjacent** in the graph G .

Note: the above definitions allow for infinite graphs, where $|V| = \infty$. In this course we will focus on finite graphs.

Graph Models: Computer Networks

- network where we care about the number of links: we use a multigraph.
- diagnostic self-links at data centers: we use a pseudograph.
- network with multiple one-way links, we use a directed (multi)graph.



Applications of Graphs

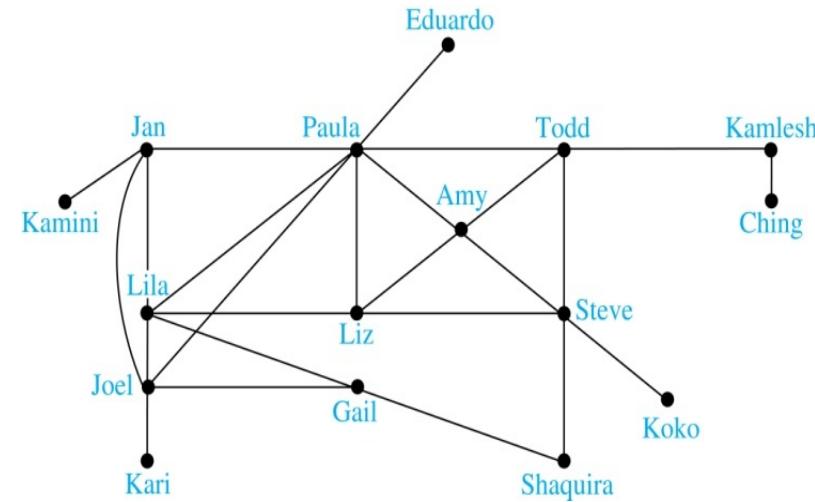
- “EVERYTHING IS A GRAPH”
(labeled, directed, etc., ...)
- graph theory can be used in modelling of:
 - Social networks
 - Communications networks
 - Information networks
 - Software design
 - Transportation networks
 - Biological networks
 -

Graph Models: Social Networks

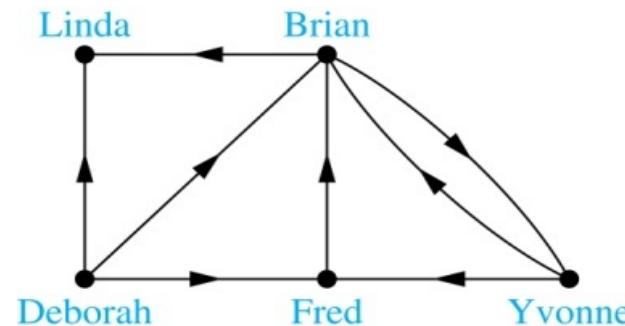
- model social structures: relationships between people or groups.
- vertices represent individuals or organizations, edges represent relationships between them.
- Useful graph models of social networks include:
 - *friendship graphs* - undirected graphs where two people are connected if they are friends (e.g., on Facebook)
 - *collaboration graphs* - undirected graphs where two people are connected if they collaborate in a specific way
 - *influence graphs* - directed graphs where there is an edge from one person to another if the first person can influence the second

Graph Models: Social Networks

Example: A friendship graph: two people are connected if they are Facebook friends.



Example: An influence graph



Information Networks

- In a *web graph*, web pages are represented by vertices and links are represented by directed edges.
- In a *citation network*:
 - Research papers are represented by vertices.
 - When paper A cites paper B, there is an edge from the vertex representing paper A to the vertex representing paper B.

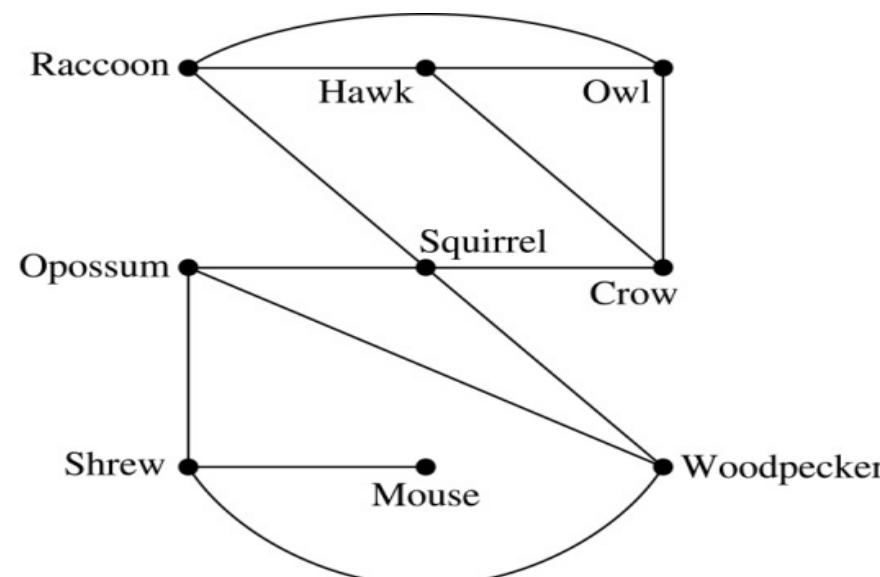
Transportation Graphs

- Graph models are extensively used to study transportation networks.
- Airline networks can be modeled using directed multigraphs, where:
 - airports are represented by vertices
 - each flight is represented by a directed edge from the vertex representing the departure airport to the vertex representing the destination airport
- Road networks modeled using graphs

Biological Applications

- Graph models are used extensively in many areas of the biological science.
- *Niche overlap graphs* model competition between species in an ecosystem:

Example: niche overlap graph for a forest ecosystem.



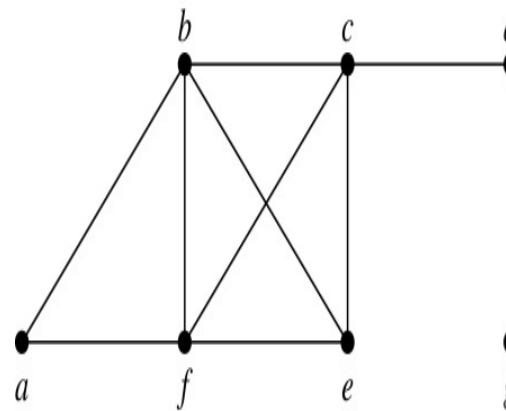
Degree and neighborhood of a vertex

Definition 3. The *degree of a vertex v in a undirected graph* is the number of edges incident with it. The degree of the vertex v is denoted by $\deg(v)$.

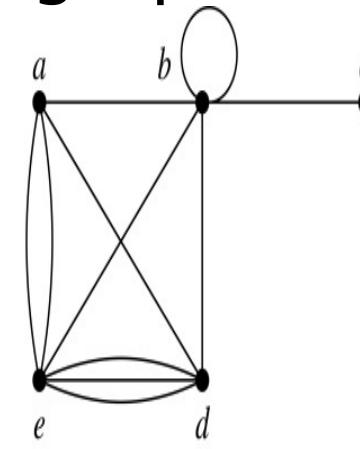
Definition 3. The *neighborhood (neighbor set) of a vertex v in a undirected graph, denoted $N(v)$* is the set of vertices adjacent to v .

Degrees and Neighborhoods of Vertices

Example: What are the degrees and neighborhoods of the vertices in the graphs G and H ?



G



H

Solution: $\deg(a) = 2$, $\deg(b) = 4$, $\deg(d) = 1$, $N(a) = \{b, f\}$, $N(b) = \{a, c, e, f\}$, $N(d) = \{c\}$.

Handshaking Theorem

THEOREM 1 (Handshaking Lemma): If $G=(V,E)$ is a undirected graph with m edges, then:

$$2m = \sum_{v \in V} \deg(v)$$

Proof:

Each edge contributes twice to the degree count of all vertices. Hence, both the left-hand and right-hand sides of this equation equal twice the number of edges. QED

Degree of Vertices (*continued*)

Theorem 2: An undirected graph has an even number of vertices of odd degree.

Proof: Let V_1 be the vertices of even degree and V_2 be the vertices of odd degree in graph $G = (V, E)$ with m edges. Then

$$\text{even} \rightarrow 2m = \sum_{v \in V} \deg(v) = \sum_{v \in V_1} \deg(v) + \sum_{v \in V_2} \deg(v).$$

must be even since $\deg(v)$ is even for each $v \in V_1$

must be even because $2m$ is even and the sum of degrees of vertices of even degree is even. Thus, since this is the sum of degrees of all vertices of odd degree, there must be an even number of them.

Handshaking Theorem: Examples

Example: How many edges are there in a graph with 10 vertices, each having degree six?

Solution: the sum of the degrees of the vertices is $6 \cdot 10 = 60$. The handshaking theorem says $2m = 60$.

So the number of edges is $m = 30$.

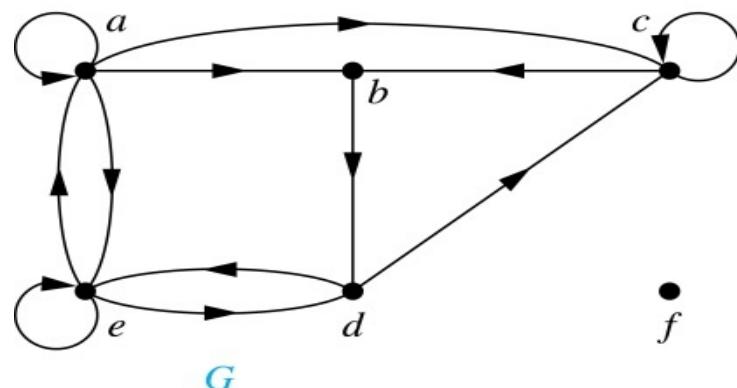
Example: If a graph has 5 vertices, can each vertex have degree 3?

Solution: This is not possible by the handshaking theorem, because the sum of the degrees of the vertices $3 \cdot 5 = 15$ is odd.

Directed Graphs

Definition: The *in-degree* of a vertex v , denoted $\deg^-(v)$, is the number of edges directed into v . The *out-degree* of v , denoted $\deg^+(v)$, is the number of edges directed out of v . Note that a loop at a vertex contributes 1 to both in-degree and out-degree.

Example: In the graph G we have



$$\begin{aligned}\deg^-(a) &= 2, \deg^-(b) = 2, \\ \deg^-(c) &= 3, \deg^-(d) = 2, \\ \deg^-(e) &= 3, \deg^-(f) = 0.\end{aligned}$$

Directed Graphs (*continued*)

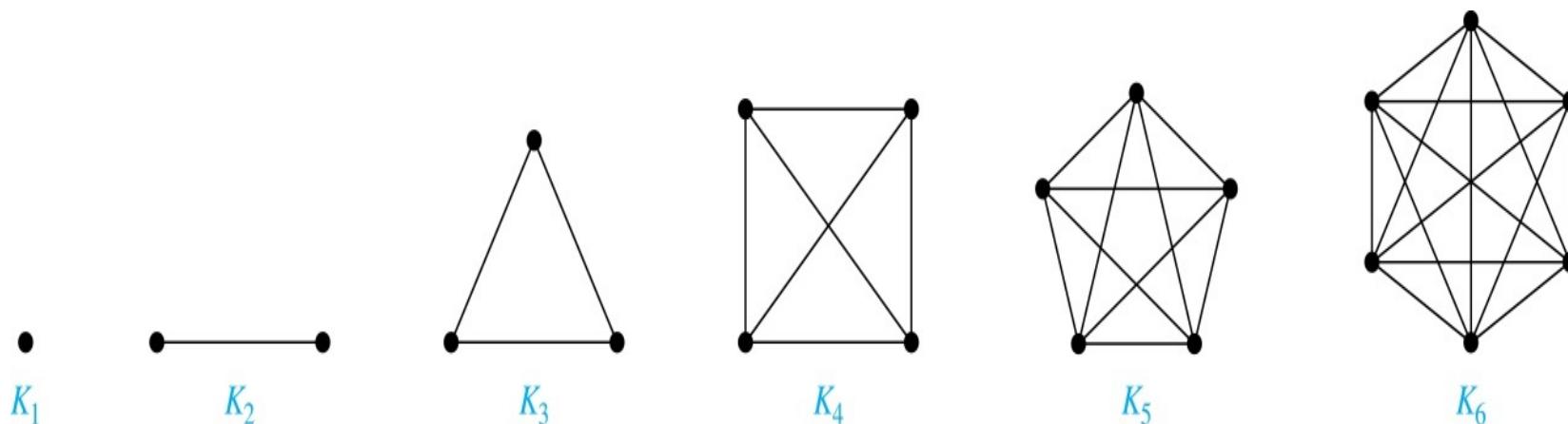
Theorem 3: Let $G = (V, E)$ be a directed graph.
Then:

$$|E| = \sum_{v \in V} \deg^-(v) = \sum_{v \in V} \deg^+(v).$$

Proof: The first sum counts the number of outgoing edges over all vertices and the second sum counts the number of incoming edges over all vertices. Both sums must be $|E|$.

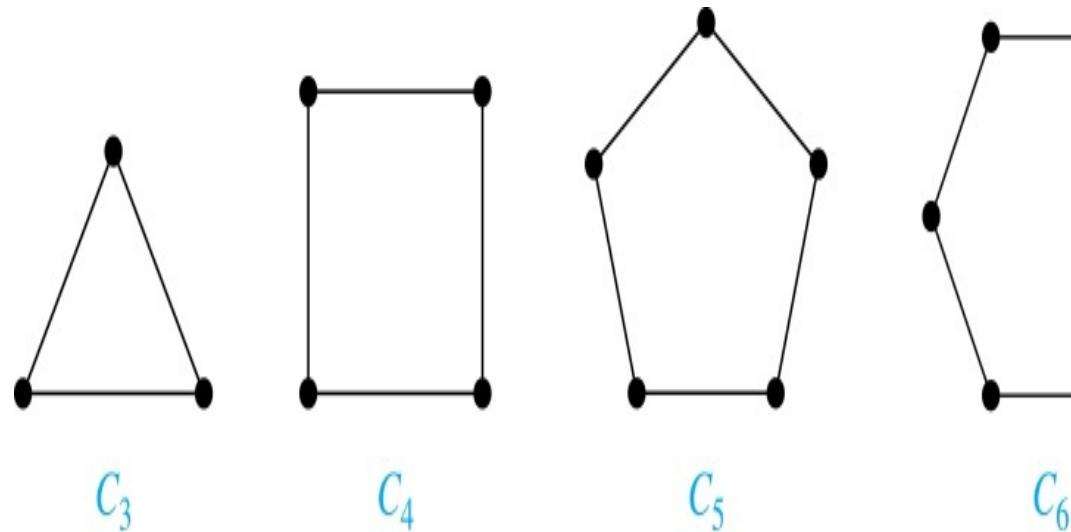
Special Types of Graphs: Complete Graphs

A *complete graph on n vertices*, denoted by K_n , is the simple graph that contains exactly one edge between each pair of distinct vertices.



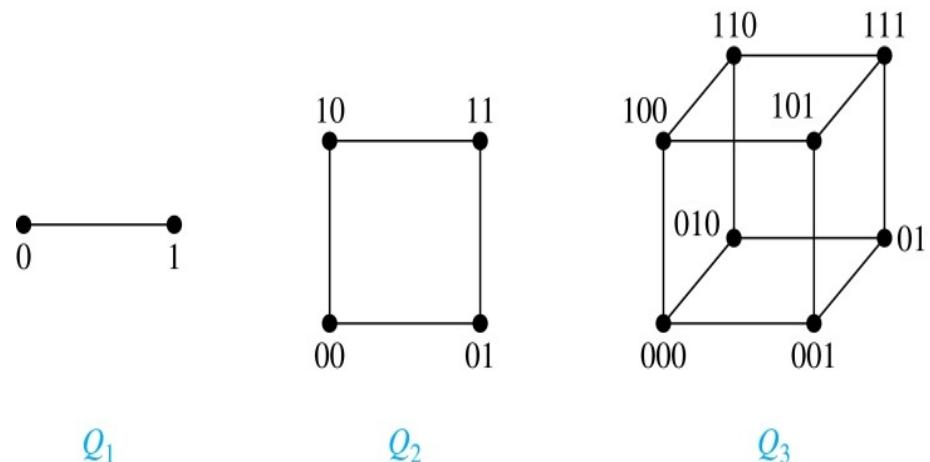
Special Types of Graphs: Cycles

A *cycle* C_n for $n \geq 3$ consists of n vertices v_1, v_2, \dots, v_n , and edges $\{v_1, v_2\}, \{v_2, v_3\}, \dots, \{v_{n-1}, v_n\}, \{v_n, v_1\}$.



Special Types of Simple Graphs: n -Cubes

An *n -dimensional hypercube*, or *n -cube*, is a graph with 2^n vertices representing all bit strings of length n , where there is an edge between two vertices if and only if they differ in exactly one bit position.

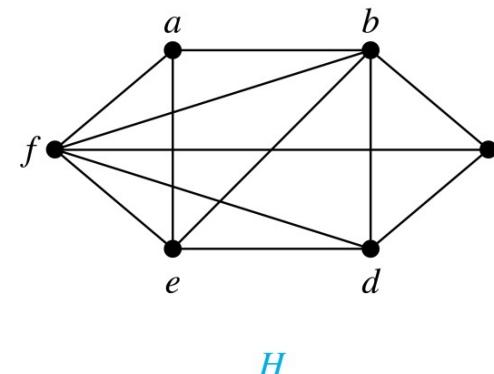
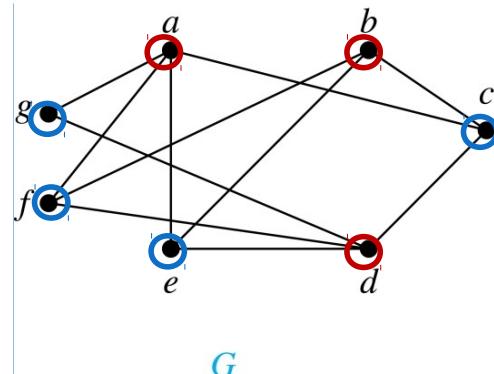


Bipartite Graphs

Definition:

An equivalent definition of a bipartite graph is one where it is possible to **color** the vertices either red or blue so that no two adjacent vertices are the same color.

G is bipartite

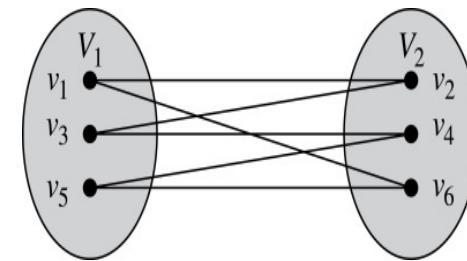


H is **not** bipartite: if we color a red, then its neighbors f and b must be blue. But f and b are adjacent.

Bipartite Graphs (*continued*)

Example: Show that C_6 is bipartite.

Solution: Partition the vertex set into $V1 = \{v1, v3, v5\}$ and $V2 = \{v2, v4, v6\}$:

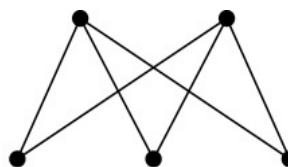
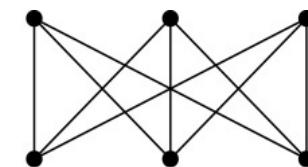
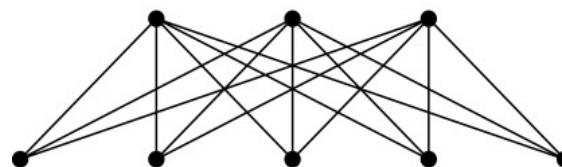
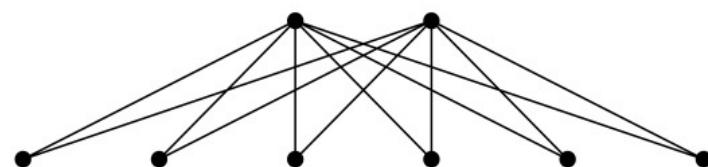


Example: Show that C_3 is not bipartite.

Solution: If we partition vertices of C_3 into two nonempty sets, one set must contain two vertices. But every vertex is connected to every other. So, the two vertices in the same partition are connected. Hence, C_3 is not bipartite.

Complete Bipartite Graphs

Definition: A *complete bipartite graph* is a graph that has its vertex set partitioned into two subsets V_1 of size m and V_2 of size n such that there is an edge from every vertex in V_1 to every vertex in V_2 .

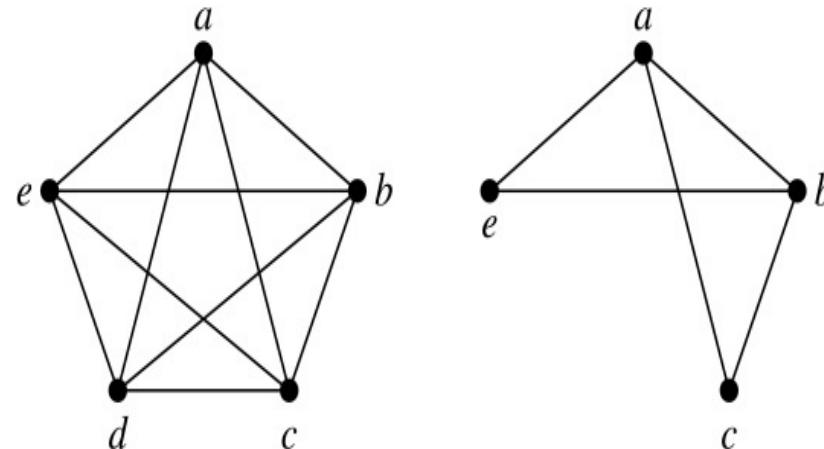
 $K_{2,3}$  $K_{3,3}$  $K_{3,5}$  $K_{2,6}$

Examples:

Subgraphs

Definition: A *subgraph* of a graph $G = (V, E)$ is a graph (W, F) , where $W \subseteq V$ and $F \subseteq E$. A subgraph H of G is a **proper subgraph** of G if $H \neq G$.

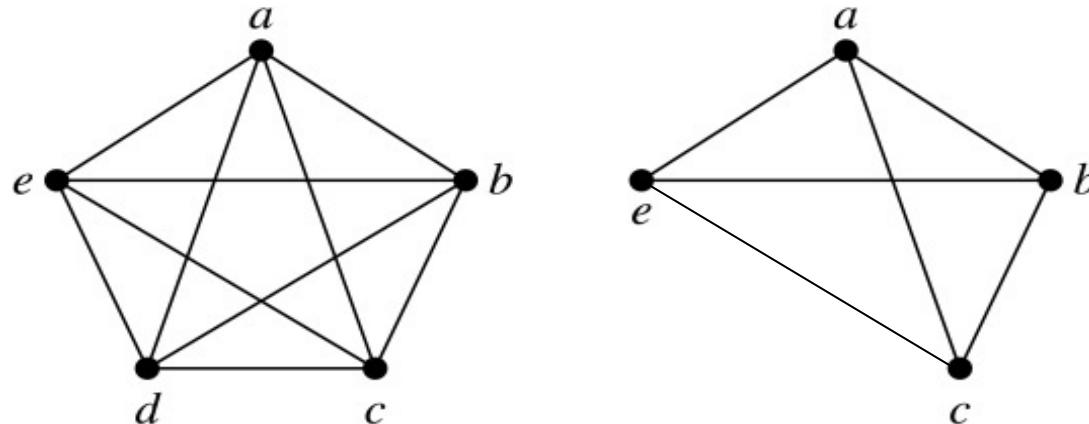
Example: here is K_5 and one of its (proper) subgraphs:



Induced Subgraphs

Definition: Let $G = (V, E)$ be a graph. The *subgraph induced* by a subset W of the vertex set V is the graph $H = (W, F)$, whose edge set F contains an edge in E if and only if both endpoints are in W .

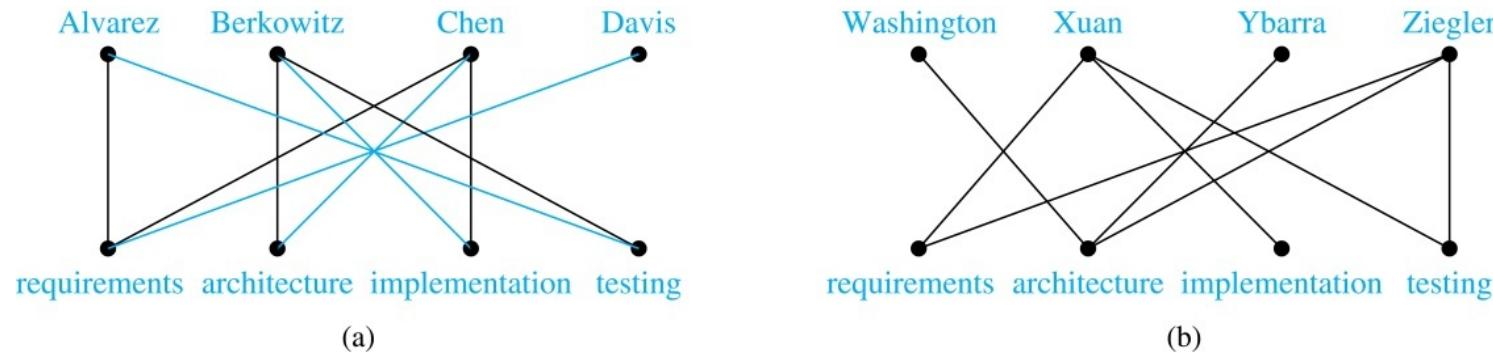
Example: Here is K_5 and its induced subgraph induced by $W = \{a, b, c, e\}$.



Bipartite Graphs and Matchings

Bipartite graphs used extensively in app's involving matching elements of two sets:

Job assignments - vertices represent the jobs and the employees, edges link employees with jobs they are qualified for. Maximize # of employees matched to jobs.



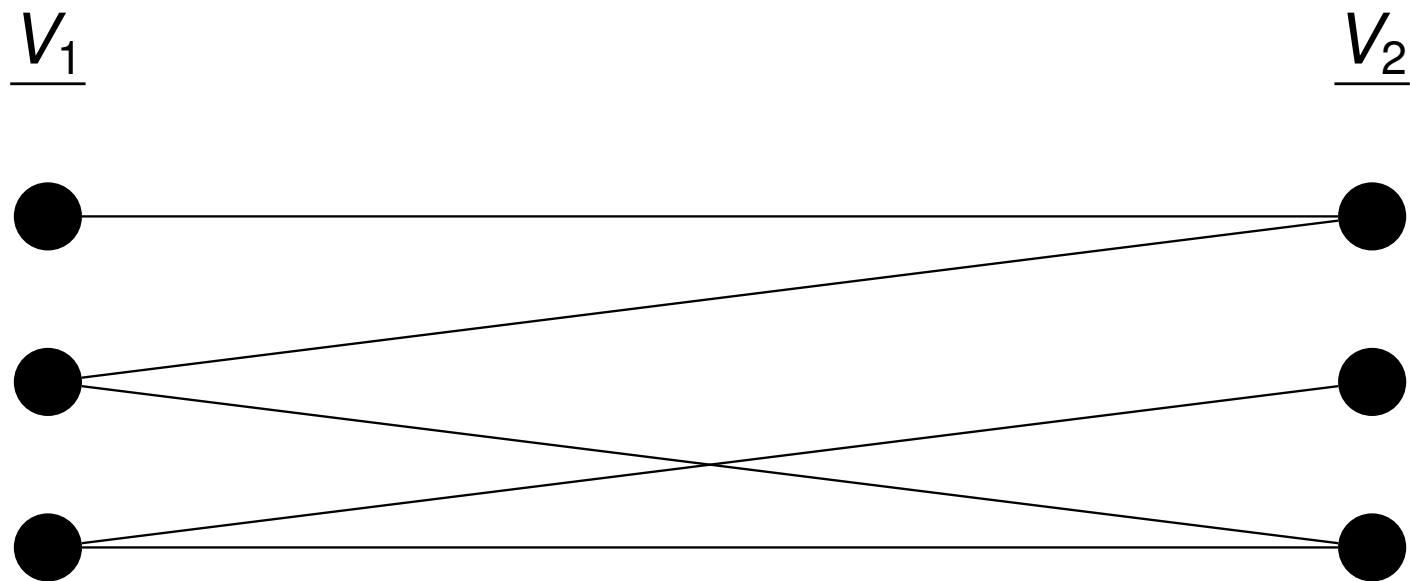
Marriage/dating - vertices represent men & women and edges link a man & woman if they are acceptable to each other as partners.

Bipartite graphs

A **bipartite graph** is a (undirected) graph $G = (V, E)$ whose vertices can be partitioned into two disjoint sets (V_1, V_2) , with $V_1 \cap V_2 = \emptyset$ and $V_1 \cup V_2 = V$, such that for every edge $e \in E$, $e = \{u, v\}$ such that $u \in V_1$ and $v \in V_2$. In other words, every edge connects a vertex in V_1 with a vertex in V_2 .

Equivalently, a graph is **bipartite if and only if** it is possible to color each vertex red or blue such that no two adjacent vertices are the same color.

Example of a Bipartite Graph



Matchings in Bipartite Graphs

A **matching**, M , in a graph, $G = (V, E)$, is a subset of edges, $M \subseteq E$, such that there does not exist two distinct edges in M that are incident on the same vertex. In other words, if $\{u, v\}, \{w, z\} \in M$, then either $\{u, v\} = \{w, z\}$ or $\{u, v\} \cap \{w, z\} = \emptyset$.

A **maximum matching** in graph G is a matching in G with the maximum possible number of edges.

Perfect/complete matchings

For a graph $G = (V, E)$, we say that a subset of edges, $W \subseteq E$, **covers** a subset of vertices, $A \subseteq V$, if for all vertices $u \in A$, there exists an edge $e \in W$, such that e is incident on u , i.e., such that $e = \{u, v\}$, for some vertex v .

In a bipartite graph $G = (V, E)$ with bipartition (V_1, V_2) , a **complete matching** with respect to V_1 , is a matching $M' \subseteq E$ that covers V_1 , and a **perfect matching** is a matching, $M^* \subseteq E$, that covers V .

Question: When does a bipartite graph have a perfect matching?

Hall's Marriage Theorem

For a bipartite graph $G = (V, E)$, with bipartition (V_1, V_2) , there exists a matching $M \subseteq E$ that covers V_1 if and only if for all $S \subseteq V_1$, $|S| \leq |N_G(S)|$.

Proof: For $G = (V, E)$, with $A \subseteq V$, let $N_G(A)$ denote the neighbors of A in G .

First, “only if” direction: Suppose there is a matching M in G that covers V_1 . We show that $\forall S \subseteq V_1$, $|S| \leq |N_G(S)|$. Suppose, for contradiction, that there is a subset $S \subseteq V_1$ such that $|S| > |N_G(S)|$. Then no matching M could possibly cover S , because there aren't enough neighbors $N_G(S)$. Done.

The “if” direction of the proof is harder...

proof of Hall's Theorem, continued...

“If” direction: Suppose $\forall S \subseteq V_1$, $|S| \leq |N_G(S)|$. Then we prove a matching M exists which covers V_1 , by **induction** on the size $|V_1|$.

Base case: $|V_1| = 1$. Since $|V_1| \leq |N_G(V_1)|$, there must be an edge covering the vertex u in $V_1 = \{u\}$.

Inductive step: Suppose (by **inductive hypothesis**) that the claim holds for bipartite graphs G' with $|V'_1| = j \leq k$. Suppose $|V_1| = k + 1$.

proof of Hall's Theorem (continued)

Case 1: Suppose that for every nonempty strict subset $S \subset V_1$, we have $|S| \leq |N_G(S)| - 1$. Take any $\{u, v\} \in E$, with $u \in V_1$. Remove u and v (and the edges incident on them) from G . Call the resulting bipartite graph G' , with bipartition $(V_1 - \{u\}, V_2 - \{v\})$.

By the induction hypothesis, there must exist a matching M' in G' that covers $V_1 - \{u\}$, because for every subset $S \subseteq V_1 - \{u\}$, $N_G(S) \subseteq N_{G'}(S) \cup \{v\}$, and thus $|N_{G'}(S)| \geq |N_G(S)| - 1 \geq |S|$. But then $M = M' \cup \{\{u, v\}\}$ is a matching in G which covers V_1 .

Case 2: Suppose, on the contrary, that there exists a nonempty strict subset $S \subset V_1$ with $|S| = |N_G(S)|$.

Any matching that covers V_1 must match S to $N_G(S)$.

By the induction hypothesis, there is a matching M' covering S on the bipartite subgraph G' of G induced by $S \cup N_G(S)$. And furthermore, the bipartite subgraph G'' of G induced by $(V_1 - S) \cup (V_2 - N_G(S))$ also satisfies the condition, and contains a matching M'' that covers $(V_1 - S)$. This is because if

$A \subseteq V_1 - S$ has $|A| > |N_{G''}(A)|$, this implies

$|A \cup S| > |N_G(A \cup S)|$, which violates the assumption about G .

Letting $M = M' \cup M''$, M defines a matching in G that covers V_1 . □

More on Matchings

Corollary A bipartite graph $G = (V, E)$ with bipartition (V_1, V_2) has a **perfect** matching if and only if $|V_1| = |V_2|$ and $\forall S \subseteq V_1$, $|S| \leq |N_G(S)|$.

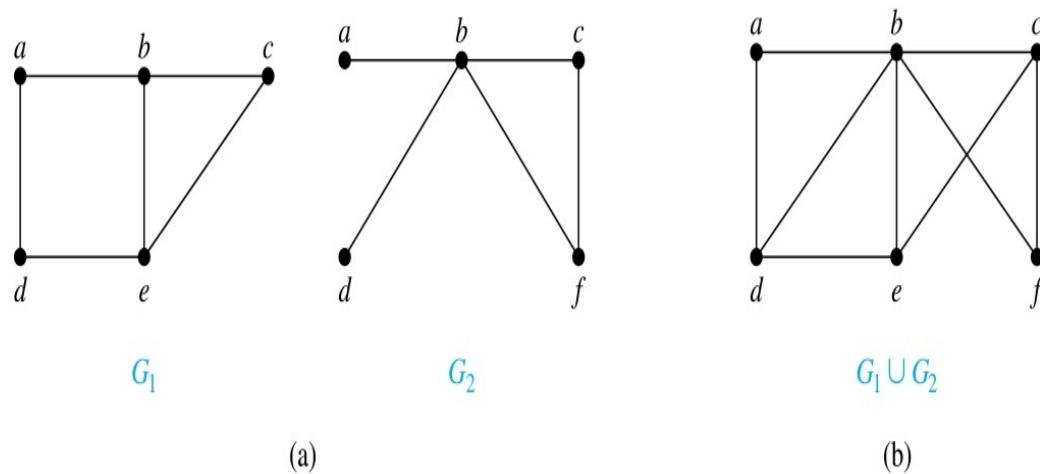
Unfortunately, the proof we have given is **not constructive enough**: it doesn't yield an (efficient) algorithm to compute a maximum matching in a bipartite graph.

An alternative proof of Hall's theorem (which we do not give) based on **alternating paths** and **augmenting paths**, is constructive & yields an efficient (polynomial time) algorithm for computing a maximum matching.

New Graphs from Old

Definition: The *union* of two simple graphs $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ is the simple graph with vertex set $V_1 \cup V_2$ and edge set $E_1 \cup E_2$. The union of G_1 and G_2 is denoted by $G_1 \cup G_2$.

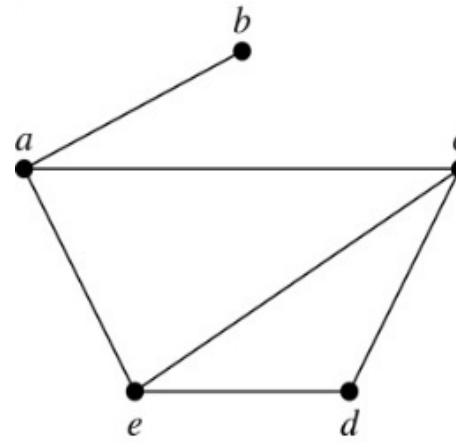
Example:



Representing Graphs: Adjacency Lists

Definition: An *adjacency list* represents a graph (with no multiple edges) by specifying the vertices that are adjacent to each vertex.

Example:



Example:

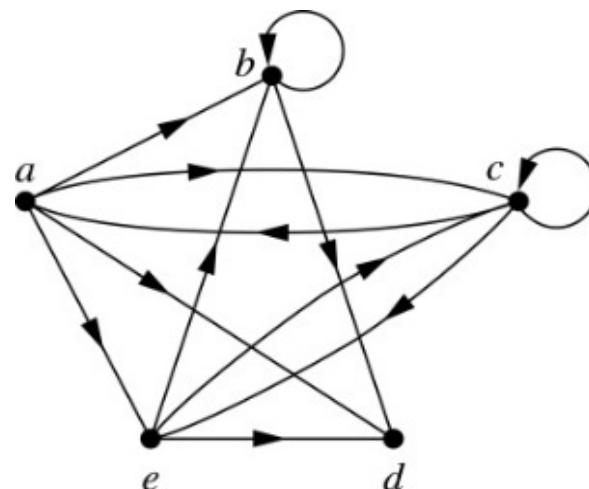


TABLE 1 An Adjacency List for a Simple Graph.

Vertex	Adjacent Vertices
a	b, c, e
b	a
c	a, d, e
d	c, e
e	a, c, d

TABLE 2 An Adjacency List for a Directed Graph.

Initial Vertex	Terminal Vertices
a	b, c, d, e
b	b, d
c	a, c, e
d	
e	b, c, d

Representation of Graphs: Adjacency Matrices

Definition: Suppose that $G = (V, E)$ is a simple graph where $|V| = n$. Arbitrarily list the vertices of G as v_1, v_2, \dots, v_n .

The *adjacency matrix*, \mathbf{A} , of G , **with respect to this listing of vertices**, is the $n \times n$ 0-1 matrix with its (i, j) th entry = 1 when v_i and v_j are adjacent, and =0 when they are not adjacent.

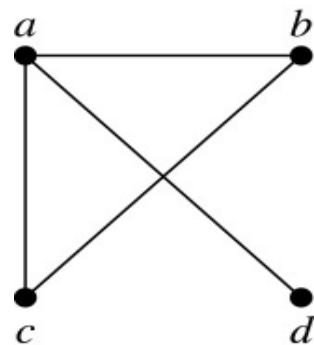
- In other words: $A = [a_{ij}]$ and:

$$a_{ij} = \begin{cases} 1 & \text{if } \{v_i, v_j\} \text{ is an edge of } G, \\ 0 & \text{otherwise.} \end{cases}$$

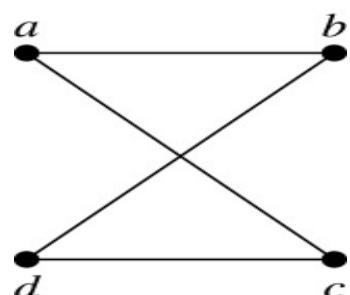
Adjacency Matrices (*continued*)

Example:

The vertex ordering is
is a, b, c, d .



$$\begin{bmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$



$$\begin{bmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix}$$

A **sparse** graph has few edges relative to the number of possible edges. Sparse graphs are more efficient to represent using an adjacency list than an adjacency matrix. But for a **dense** graph, an adjacency matrix is often preferable.

Note: The adjacency matrix of an undirected graph is **symmetric**:

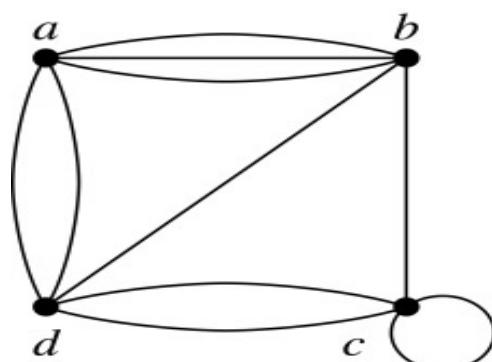
$$a_{ij} = a_{ji}, \forall i, j$$

Also, since there are no loops, each diagonal entry is zero: $a_{ii} = 0, \forall i$

Adjacency Matrices (*continued*)

- Adjacency matrices can also be used to represent graphs with loops and multi-edges.
- When multiple edges connect vertices vi and vj , (or if multiple loops present at the same vertex), the (i, j) th entry equals the number of edges connecting the pair of vertices.

Example: Adjacency matrix of a pseudograph, using vertex ordering a, b, c, d :



$$\begin{bmatrix} 0 & 3 & 0 & 2 \\ 3 & 0 & 1 & 1 \\ 0 & 1 & 1 & 2 \\ 2 & 1 & 2 & 0 \end{bmatrix}$$

Adjacency Matrices (*continued*)

- Adjacency matrices can represent directed graphs in exactly the same way. The matrix A for a directed graph $G = (V, E)$ has a 1 in its (i, j) th position if there is an edge from v_i to v_j , where v_1, v_2, \dots, v_n is a list of the vertices.
 - In other words,

$$\begin{aligned} a_{ij} &= 1 \quad \text{if } (i, j) \in E \\ a_{ij} &= 0 \quad \text{if } (i, j) \notin E \end{aligned}$$

- Note: the adjacency matrix for a directed graph need not be symmetric.

Isomorphism of Graphs

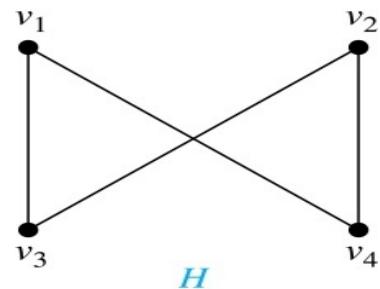
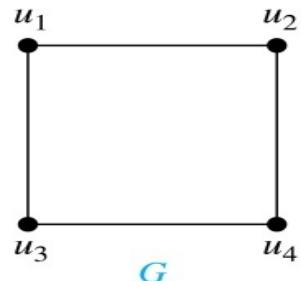
Definition: Two (undirected) graphs $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ are *isomorphic* if there is a bijection, $f: V_1 \rightarrow V_2$, with the property that for all vertices $a, b \in V_1$

$$\{a, b\} \in E_1 \quad \text{if and only if} \quad \{f(a), f(b)\} \in E_2$$

Such a function f is called an *isomorphism*. Intuitively, isomorphic graphs are “THE SAME”, except for “renamed” vertices.

Isomorphism of Graphs (cont.)

Example: Show that the graphs $G = (V, E)$ and $H = (W, F)$ are isomorphic.



Solution: The function f with $f(u_1) = v_1$, $f(u_2) = v_4$, $f(u_3) = v_3$, and $f(u_4) = v_2$ is a one-to-one correspondence between V and W .

Isomorphism of Graphs (cont.)

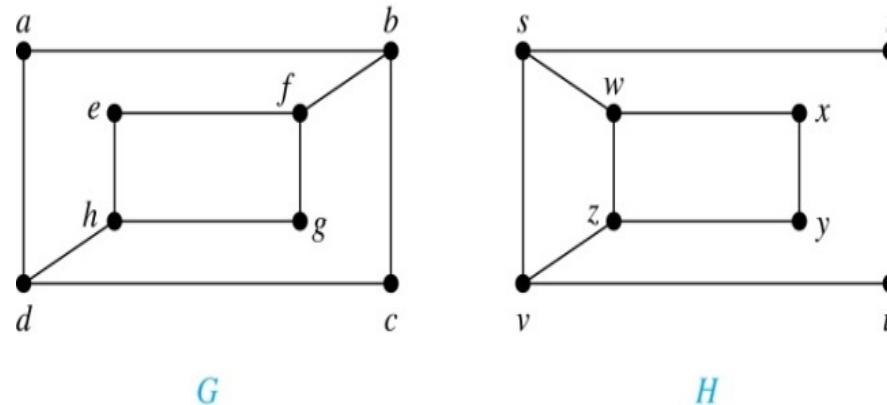
It is difficult to determine whether two graphs are isomorphic by brute force: there are $n!$ bijections between vertices of two n -vertex graphs.

Often, we can show two graphs are not isomorphic by finding a property that only one of the two graphs has. Such a property is called *graph invariant*:

- e.g., number of vertices of given degree, the degree sequence (list of the degrees),

Isomorphism of Graphs (cont.)

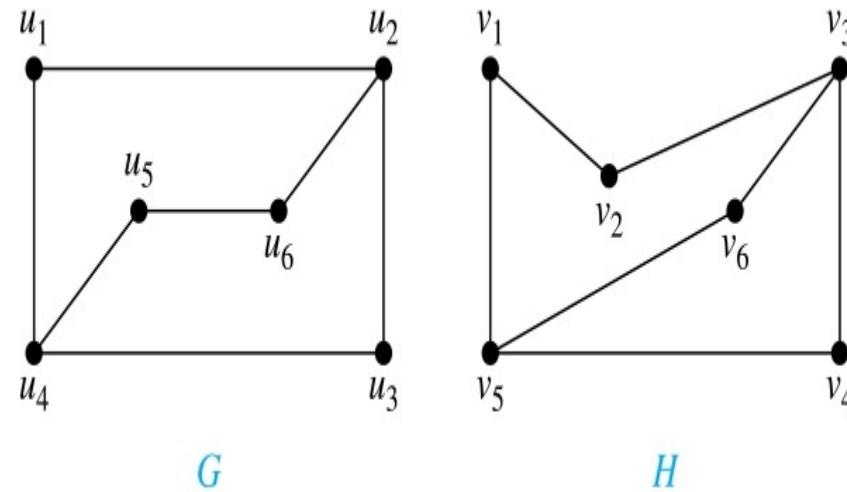
Example: Are these graphs are isomorphic?



Solution: No! Since $\deg(a) = 2$ in G , a must correspond to t , u , x , or y , since these are the vertices of degree 2 in H . But each of these vertices is adjacent to another vertex of degree 2 in H , which is not true for a in G . So, G and H can not be isomorphic.

Isomorphism of Graphs (cont.)

Example: Determine whether these two graphs are isomorphic.



Solution: The function f is defined by: $f(u1) = v6$, $f(u2) = v3$, $f(u3) = v4$, $f(u4) = v5$, $f(u5) = v1$, and $f(u6) = v2$ is a bijection.

Algorithms for Graph Isomorphism

- The best algorithms known for determining whether two graphs are isomorphic have exponential worst-case time complexity (in the number of vertices of the graphs).
- However, there are algorithms with good time complexity in many practical cases.
- See, e.g., a publicly available software called NAUTY for graph isomorphism.

Applications of Graph Isomorphism

The question whether graphs are isomorphic plays an important role in applications of graph theory. For example:

Chemists use molecular graphs to model chemical compounds. Vertices represent atoms and edges represent chemical bonds. When a new compound is synthesized, a database of molecular graphs is checked to determine whether the new compound is isomorphic to the graph of an already known one.

Section Summary

- Paths
- Connectedness in Undirected Graphs
- (strong) Connectedness in Directed Graphs

Paths (in undirected graphs)

Informally, a **path** is a sequence of edges connecting vertices.

Formally:

Definition: For an undirected graph $G = (V, E)$, an integer $n \geq 0$, and vertices $u, v \in V$, a **path (or walk)** of length n from u to v in G is a sequence:

$$x_0, e_1, x_1, e_2, \dots, x_{n-1}, e_n, x_n$$

of interleaved vertices $x_j \in V$ and edges $e_i \in E$, such that $x_0 = u$ and $x_n = v$, and such that $e_i = \{x_{i-1}, x_i\} \in E$ for all $i \in \{1, \dots, n\}$.

Such a path **starts** at u and **ends** at v . A path of length $n \geq 1$ is called a **circuit** (or **cycle**) if $n \geq 1$ and the path starts and ends at the same vertex, i.e., $u = v$.

A path or circuit is called **simple** if it does not contain the same edge more than once.

More on paths

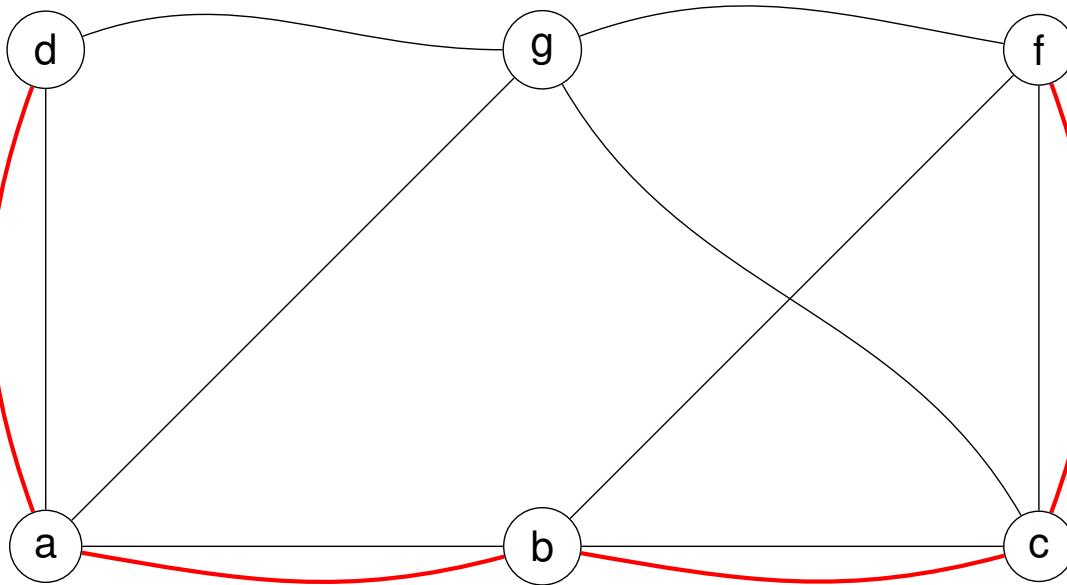
When $G = (V, E)$ is a simple undirected graph a path $x_0, e_1, \dots, e_n, x_n$ is determined uniquely by the sequence of vertices x_0, x_1, \dots, x_n . So, for simple undirected graphs we can denote a path by its sequence of vertices x_0, x_1, \dots, x_n .

Note 1: The word “simple” is overloaded. Don’t confuse a simple undirected graph with a simple path. There can be a simple path in a non-simple graph, and a non-simple path in a simple graph.

Note 2: The terms “path” and “simple path” used in Rosen’s book are not entirely standard. Other books use the terms “walk” and “trail” to denote “path” and “simple path”, respectively. Furthermore, others use “path” itself to mean a walk that doesn’t re-visit any vertex, except possibly the first and last in case it is a circuit. To stick to Rosen’s terminology, we shall use the non-standard term tidy path to refer to such a walk.

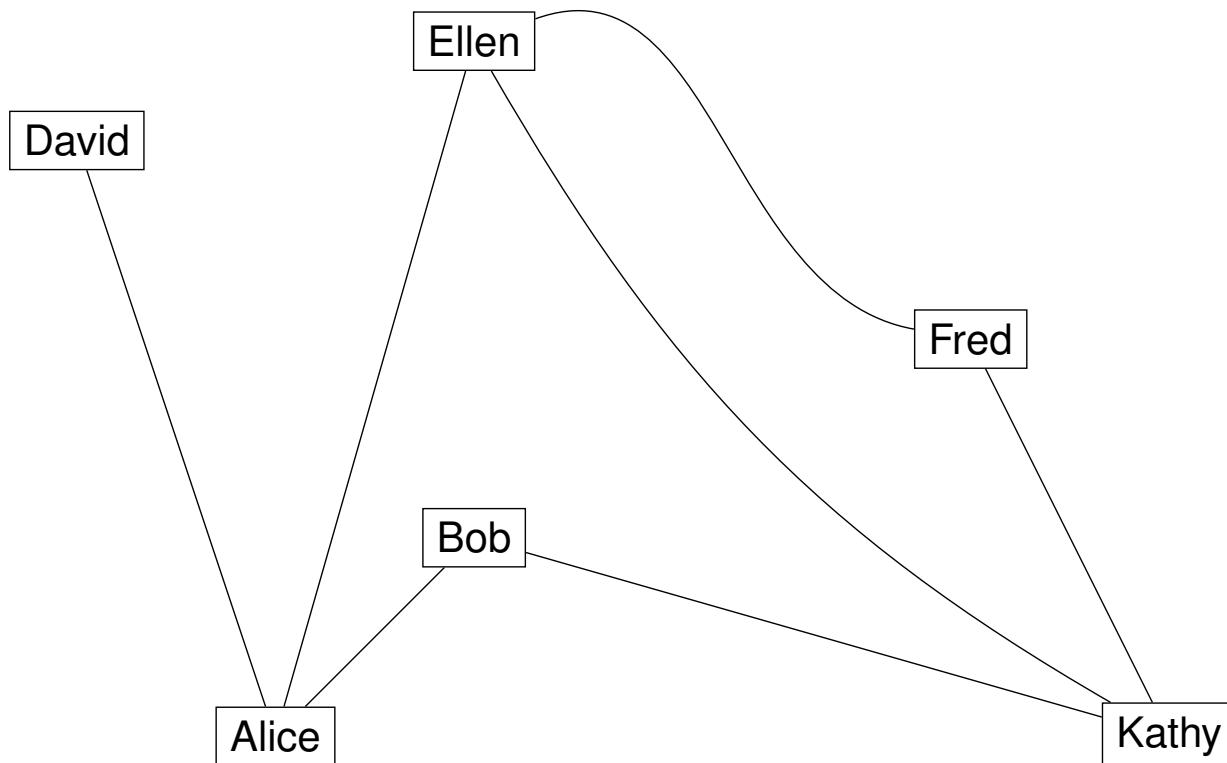
Example

Here is a simple undirected graph:



- d, a, b, c, f is a simple (and tidy) path of length 4.
- d, g, c, b, a, d is a simple (and tidy) circuit of length 5.
- a, b, g, f **is not** a path, because $\{b, g\}$ is not an edge.
- d, a, b, c, f, b, a, g **is** a path, but it **is not** a simple path, because the edge $\{a, b\}$ occurs twice in it.
- c, g, a, d, g, f is a simple path, but it **is not** a tidy path, because vertex g occurs twice in it.

Example: an acquaintance graph



Paths in directed graphs (same definitions)

Definition: For an directed graph $G = (V, E)$, an integer $n \geq 0$, and vertices $u, v \in V$, a **path (or walk) of length n from u to v** in G is a sequence of vertices and edges $x_0, e_1, x_1, e_2, \dots, x_n, e_n$, such that $x_0 = u$ and $x_n = v$, and such that $e_i = (x_{i-1}, x_i) \in E$ for all $i \in \{1, \dots, n\}$.

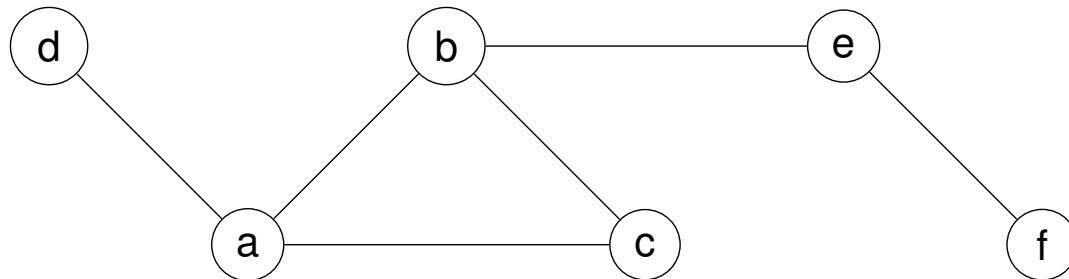
When there are no multi-edges in the directed graph G , the path can be denoted (uniquely) by its vertex sequence x_0, x_1, \dots, x_n .

A path of length $n \geq 1$ is called a **circuit (or cycle)** if the path starts and ends at the same vertex, i.e., $u = v$.

A path or circuit is called **simple** if it does not contain the same edge more than once. (And we call it **tidy** if it does not contain the same vertex more than once, except possibly the first and last in case $u = v$ and the path is a circuit (cycle).)

Connectness in undirected graphs

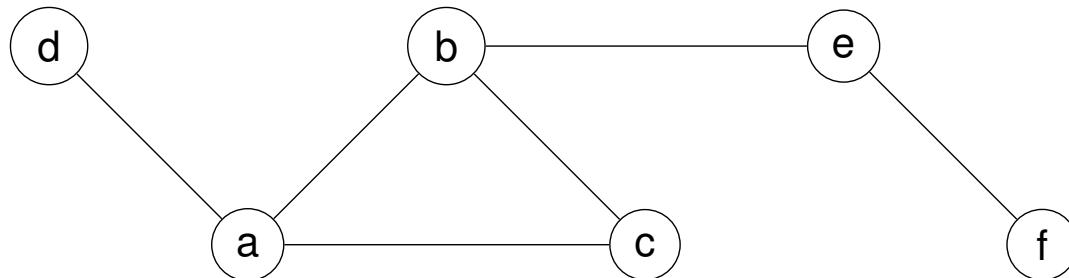
Definition: An undirected graph $G = (V, E)$ is called **connected**, if there is a path between every pair of distinct vertices. It is called **disconnected** otherwise.



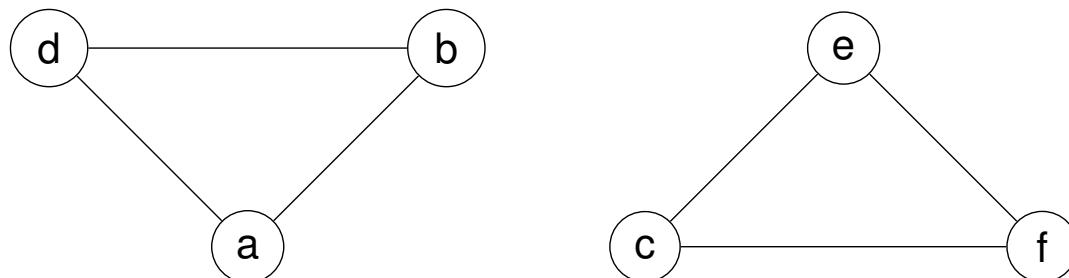
This graph is connected

Connectness in undirected graphs

Definition: An undirected graph $G = (V, E)$ is called **connected**, if there is a path between every pair of distinct vertices. It is called **disconnected** otherwise.



This graph is connected



This graph is **not** connected

Proposition

There is always a simple, and tidy, path between any pair of vertices u, v of a connected undirected graph G .

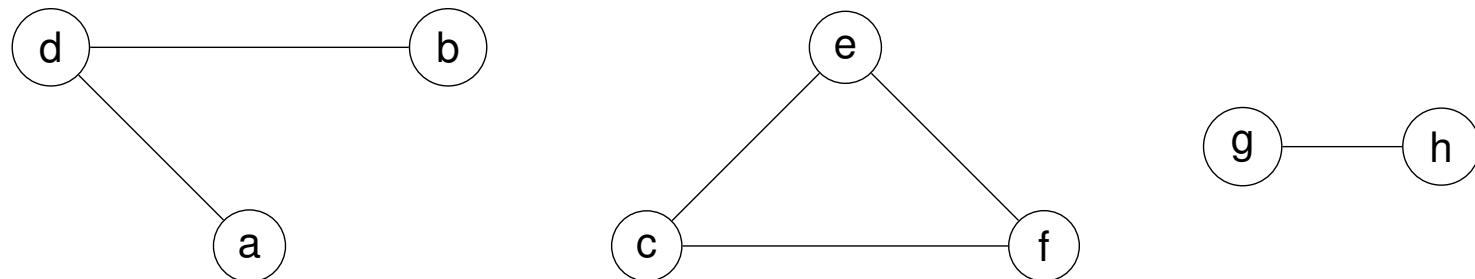
Proof: By definition of connectedness, for every pair of vertices u, v , there must exist a **shortest** path $x_0, e_1, x_1, \dots, e_n, x_n$ in G such that $x_0 = u$ and $x_n = v$.

Suppose this path is not tidy, and $n \geq 1$. (If $n = 0$, the Proposition is trivial.) Then $x_j = x_k$ for some $0 \leq j < k \leq n$. But then $x_0, e_1, x_1, \dots, x_j, e_{k+1}, x_{k+1}, \dots, e_n, x_n$ is a shorter path from u to v , contradicting the assumption that the original path was shortest.



connected components of undirected graphs

Definition: A **connected component** $H = (V', E')$ of a graph $G = (V, E)$ is a *maximal* connected subgraph of G , meaning H is connected and $V' \subseteq V$ and $E' \subseteq E$, but H is not a proper subgraph of a larger connected subgraph R of G .



This graph, $G = (V, E)$, has 3 connected components.
(It is thus a disconnected graph.)

One connected component of G is $H_1 = (V'_1, E'_1)$,
where $V'_1 = \{d, a, b\}$ and $E'_1 = \{\{d, a\}, \{d, b\}\}$.

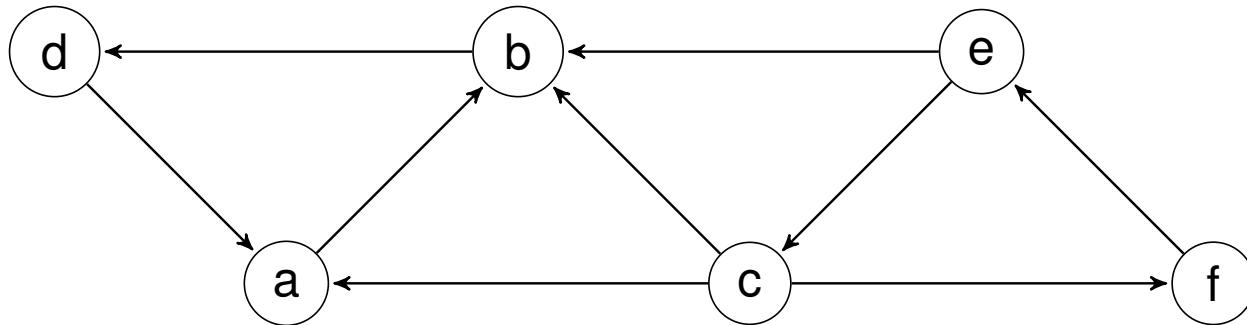
Connectedness in directed graphs

Definition: A directed graph $G = (V, E)$ is called **strongly connected**, if for every pair of vertices u and v in V , there is a (directed) path from u to v , *and* a directed path from v to u .

$(G = (V, E))$ is **weakly connected** if there is a path between every pair of vertices in V in the underlying undirected graph (meaning when we ignore the direction of edges in E .)

A **strongly connected component (SCC)** of a directed graph G , is a maximal strongly connected subgraph H of G which is not contained in a larger strongly connected subgraph of G .

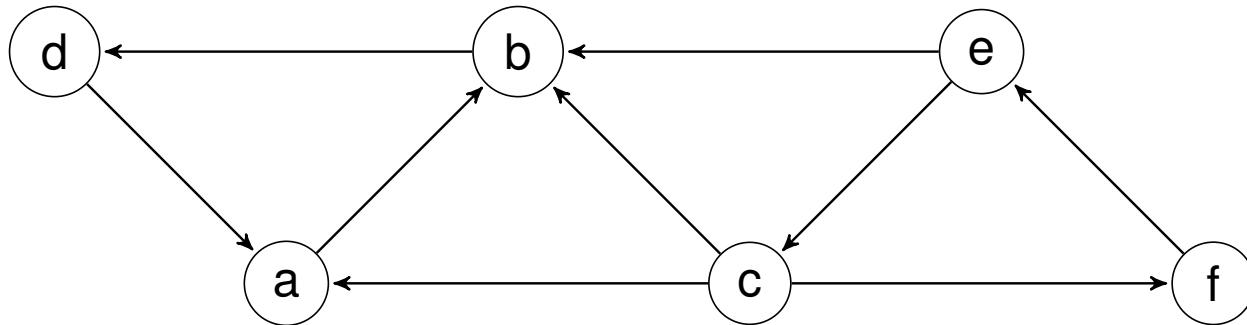
Example



This digraph, G , is **not** strongly connected, because, for example, there is no directed path from b to c .

Question: what are the strongly connected components (SCCs) of G ?

Example



This digraph, G , is **not** strongly connected, because, for example, there is no directed path from b to c .

Question: what are the strongly connected components (SCCs) of G ?

One strongly connected component (SCC) of G is $H_1 = (V'_1, E'_1)$, where $V'_1 = \{d, a, b\}$ and $E'_1 = \{(d, a), (a, b), (b, d)\}$.

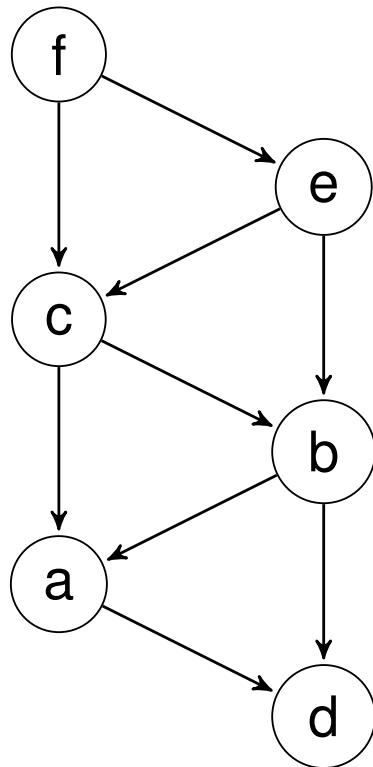
Another SCC of G is $H_2 = (V'_2, E'_2)$, where $V'_2 = \{e, c, f\}$ and $E'_2 = \{(e, c), (c, f), (f, e)\}$.

There are no other SCCs in G .

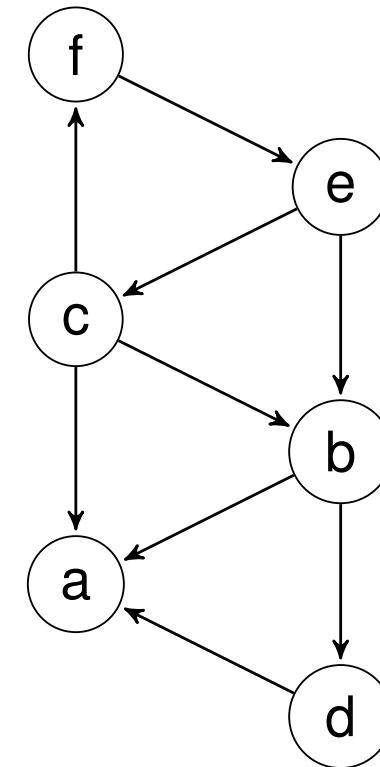
Directed Acyclic Graphs

A **Directed Acyclic Graph (DAG)**, is a directed graph that contains no circuits or loops.

Example:



This is a DAG

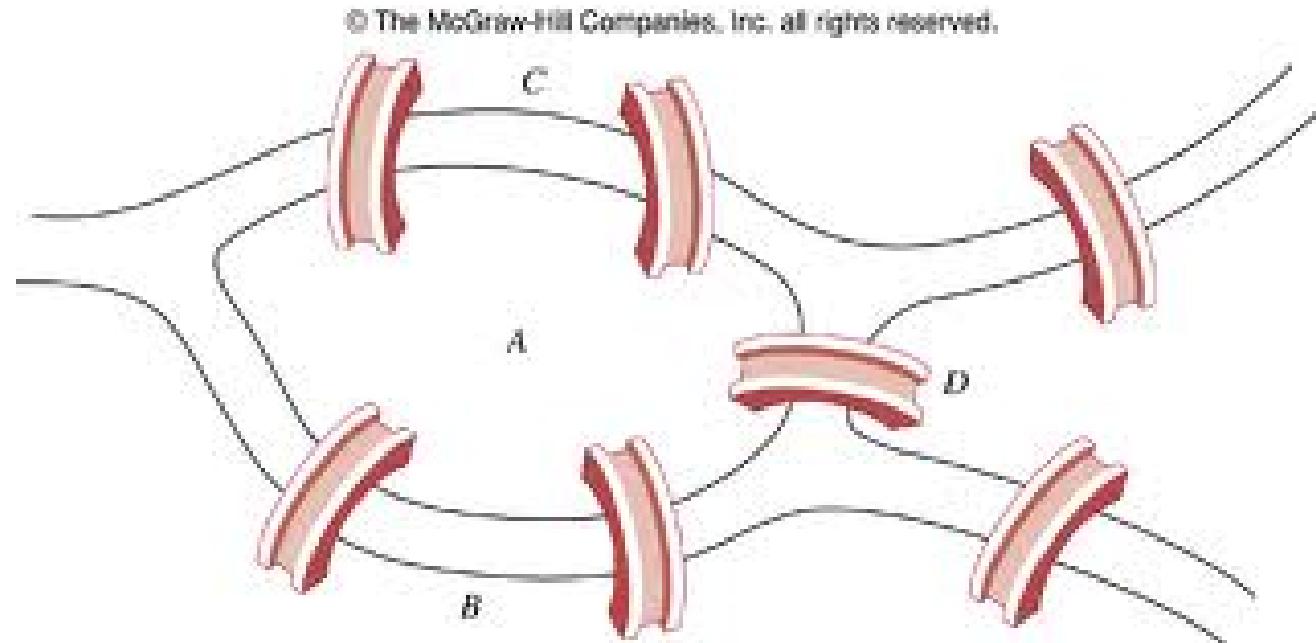


This is NOT a DAG

- Euler Paths and Euler Circuits
- Hamiltonian Paths and Hamiltonian Circuits

The Königsberg Bridge Problem

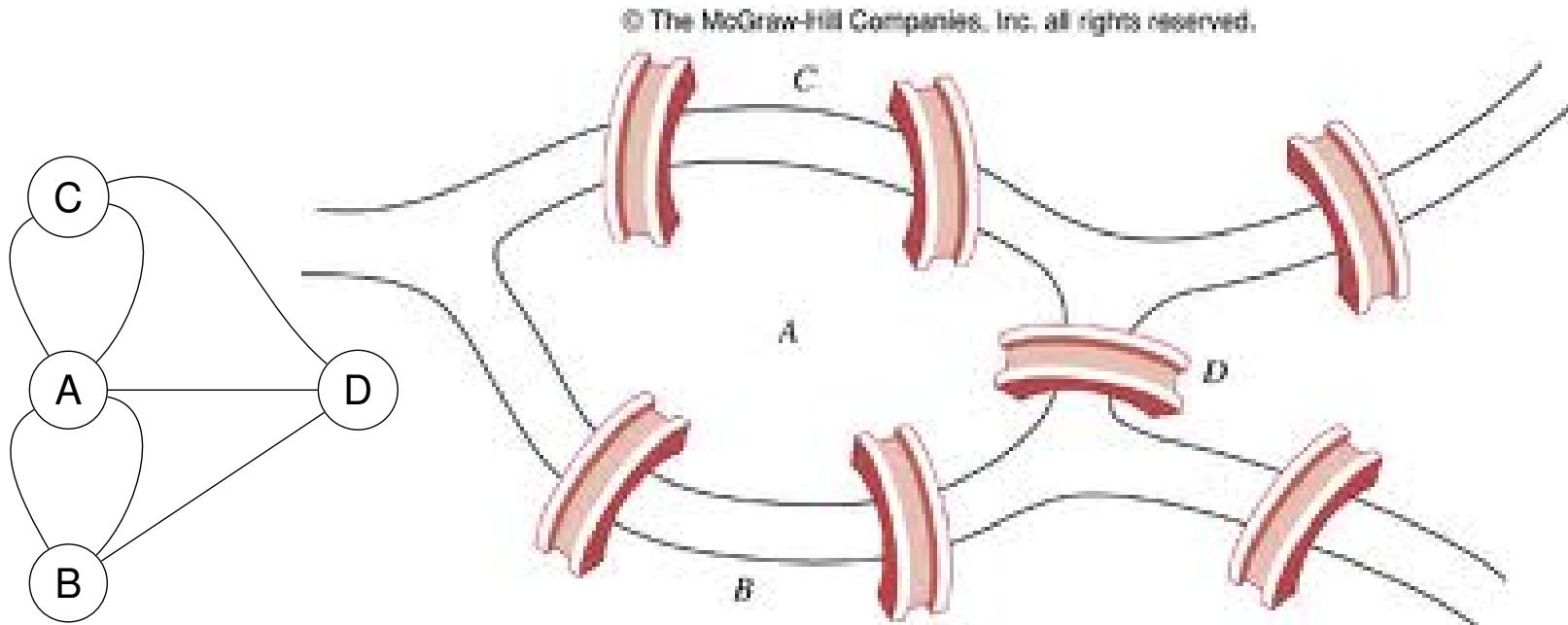
Leonard Euler (1707-1783) was asked to solve the following:



Question: Can you start a walk somewhere in Königsberg, walk across each of the 7 bridges **exactly once**, and end up back where you started from?

The Königsberg Bridge Problem

Leonard Euler (1707-1783) was asked to solve the following:



Question: Can you start a walk somewhere in Königsberg, walk across each of the 7 bridges **exactly once**, and end up back where you started from?

Euler (in 1736) used “graph theory” to answer this question.

Euler paths and Euler Circuits

Recall that an (undirected) **multigraph** does not have any loops, but can have multiple edges between the same pair of vertices.

Definition: An **Euler path** in a multigraph G is a simple path that contains every edge of G .

(So, every edge occurs exactly once in the path.)

An **Euler circuit** in an multigraph G is a simple circuit that contains every edge of G .

(So, every edge occurs exactly once in the circuit.)

Question: Is there a simple criterion for determining whether a multigraph G has an Euler path (an Euler circuit)?

Euler paths and Euler Circuits

Recall that an (undirected) **multigraph** does not have any loops, but can have multiple edges between the same pair of vertices.

Definition: An **Euler path** in a multigraph G is a simple path that contains every edge of G .

(So, every edge occurs exactly once in the path.)

An **Euler circuit** in an multigraph G is a simple circuit that contains every edge of G .

(So, every edge occurs exactly once in the circuit.)

Question: Is there a simple criterion for determining whether a multigraph G has an Euler path (an Euler circuit)?

Answer: Yes.

Euler's Theorem

Euler's Theorem (1736)

A connected undirected multigraph with at least two vertices has an Euler circuit if and only if each of its vertices has even degree.

Proof: “Only if” direction: Suppose a multigraph $G = (V, E)$ has an Euler circuit, $x_0 e_1 x_1 e_2 \dots e_m x_m$, where $x_0 = x_m = u$.

For every vertex $v \in V$, $v \neq u$, each time we enter v via an edge e_i , we must leave v via a different edge e_{i+1} . So, in total, since we see all edges incident to v exactly once, all such vertices v must have even degree.

Likewise, the initial (and final) vertex $u = x_0 = x_m$, must also have even degree, because the edges e_1 and e_m pair up in the same way.

Proof of Euler's Theorem (continued)

The (harder) “if” direction: Suppose $G = (V, E)$ is connected and every vertex in V has even degree.

We give a **constructive proof** that, given such a multigraph G , shows how to construct an Euler circuit (efficiently).

Start a “walk” at any vertex v , never re-using an edge, walking for as long as possible until you can not do so any more.

Claim: Such a “walk” (simple path), w_1 , must end at the vertex v where it started (i.e., it must be a circuit).

Reason: For any vertex z other than v , whenever the walk enters z via an edge, there must be an odd number of edges incident to z “remaining”. Note: zero is not an odd number! After leaving z , there must be an even number of edges of z “remaining”.

If the simple circuit w_1 covers every edge of G , we are done.

If not,

Proof of Euler's theorem (final part)

Note that every vertex has even degree “remaining” after the edges of the simple circuit w_1 are removed.

If the simple circuit w_1 does not cover every edge of G , since G is connected, there must be some vertex x' on the circuit w_1 which is incident to an edge not in w_1 . So, $w_1 = w'_1 x w''_1$

We start a new walk at the vertex x' , on the “remaining” graph without the edges of w_1 . This yields a new circuit w_2 that must start and end at x' .

We can then then “splice” w_2 inside w_1 (at the point where x' occurs) in order to get a new longer Euler circuit: $w'_1 w_2 w''_1$.

We can do this same process repeatedly until there are no edges remaining. □

Note: this also yields a reasonably efficient algorithm for computing an Euler circuit in a connected multigraph where every vertex has even degree.

Euler's theorem for paths

Euler's Theorem for paths

A connected undirected multigraph G has an Euler path which is **not** an Euler circuit if and only if G has exactly two vertices of odd degree.

The proof is very similar to the case of Euler circuits: just start the initial walk at one of the vertices of odd degree.

The proof is thus similarly constructive, and yields an efficient algorithm to construct an Euler path, if one exists.

Hamiltonian Paths

Definition: A Hamiltonian path in a (undirected) graph G is a simple path that visits every vertex exactly once. (In other words, it is a tidy path that visits every vertex.)

A Hamiltonian circuit in a (undirected) graph G is a simple circuit that passes through every vertex exactly once (except for the common start and end vertex, which is seen exactly twice).

Question: Is there a simple criterion for determining whether a (simple undirected) graph has a Hamiltonian path, or Hamiltonian circuit?

Hamiltonian Paths

Definition: A Hamiltonian path in a (undirected) graph G is a simple path that visits every vertex exactly once. (In other words, it is a tidy path that visits every vertex.)

A Hamiltonian circuit in a (undirected) graph G is a simple circuit that passes through every vertex exactly once (except for the common start and end vertex, which is seen exactly twice).

Question: Is there a simple criterion for determining whether a (simple undirected) graph has a Hamiltonian path, or Hamiltonian circuit?

Answer: No. Nobody knows any efficient algorithm for determining whether a given (arbitrary) graph G has a Hamiltonian path/circuit. The problem is “**NP-complete**”.

More on Hamiltonian paths/circuits

There are **sufficient** criteria that guarantee existence of a Hamiltonian circuit. For example:

Ore's Theorem

Every simple undirected graph, $G = (V, E)$, with $n \geq 3$ vertices, in which $\deg(u) + \deg(v) \geq n$ for every two non-adjacent vertices u and v in V , has a Hamiltonian circuit.

Corollary (Dirac's Theorem)

Every simple undirected graph, $G = (V, E)$, with $n \geq 3$ vertices, in which $\deg(u) \geq n/2$ for all vertices $u \in V$, has a Hamiltonian circuit.

We will NOT prove these theorems, and we will NOT assume that you know these theorems.

Shortest Paths, and Dijkstra's Algorithm: Overview

- Graphs with lengths/weights/costs on edges.
- Shortest paths in edge-weighted graphs
- Dijkstra's classic algorithm for computing single-source shortest paths.

Graphs with edge “length” (or “weight/cost”)

An **edge-weighted directed graph**, $G = (V, E, w)$, has a **length/weight/cost** function, $w : E \rightarrow \mathbb{N}$, which maps each edge $(u, v) \in E$ to a non-negative integer “length” (or “weight”, or “cost”): $w(u, v) \in \mathbb{N}$.

We can **extend** the “length” function w to a function $w : V \times V \rightarrow \mathbb{N} \cup \{\infty\}$, by letting $w(u, u) = 0$, for all $u \in V$, and letting $w(u, v) = \infty$ for all $(u, v) \notin E$.

Consider a directed path:

$$x_0 e_1 x_1 e_2 \dots e_n x_n$$

from $u = x_0 \in V$ to $v = x_n \in V$, in graph $G = (V, E, w)$. The **length** of this path is defined to be: $\sum_{i=1}^n w(x_{i-1}, x_i)$.

Graphs with edge “length” (or “weight/cost”)

An **edge-weighted directed graph**, $G = (V, E, w)$, has a **length/weight/cost** function, $w : E \rightarrow \mathbb{N}$, which maps each edge $(u, v) \in E$ to a non-negative integer “length” (or “weight”, or “cost”): $w(u, v) \in \mathbb{N}$.

We can **extend** the “length” function w to a function $w : V \times V \rightarrow \mathbb{N} \cup \{\infty\}$, by letting $w(u, u) = 0$, for all $u \in V$, and letting $w(u, v) = \infty$ for all $(u, v) \notin E$.

Consider a directed path:

$$x_0 e_1 x_1 e_2 \dots e_n x_n$$

from $u = x_0 \in V$ to $v = x_n \in V$, in graph $G = (V, E, w)$. The **length** of this path is defined to be: $\sum_{i=1}^n w(x_{i-1}, x_i)$.

Question: Given G and a pair of vertices $u, v \in V$, how do we compute the length of the **shortest path** from u to v ?

Dijkstra's single-source shortest-path algorithm

Input: Edge-weighted graph, $G = (V, E, w)$, with (*extended*) weight function $w : V \times V \rightarrow \mathbb{N}$, and a source vertex $s \in V$.

Output: Function $L : V \rightarrow \mathbb{N} \cup \{\infty\}$, such that for all $v \in V$, $L(v)$ is the length of the shortest path from s to v in G .

Algorithm:

Initialize: $S := \{s\}$; $L(s) := 0$;

Initialize: $L(v) := w(s, v)$, for all $v \in V - \{s\}$;

while ($S \neq V$) **do**

$u := \arg \min_{z \in V - S} \{L(z)\}$

$S := S \cup \{u\}$

for all $v \in V - S$ such that $(u, v) \in E$ **do**

$L(v) := \min\{L(v), L(u) + w(u, v)\}$

end for

end while

Output function $L(\cdot)$.

Why does Dijkstra's algorithm work?

Claim: The While loop of Dijkstra's algorithm maintains the following **invariant** properties of the function L and the set S :

1. $\forall v \in S, L(v)$ is the shortest path length from s to v in G .
2. $\forall v \in V - S, L(v)$ is the length of the shortest path from s to v which uses only vertices in $S \cup \{v\}$.
3. For all $u \in S$ and $v \in V - S, L(u) \leq L(v)$.

Note that the three invariants hold after initialization, just prior to the first iteration of the while loop.

The claim follows once we prove (on board) that **if** the invariants hold just prior to a while loop iteration **then** they hold just after.

Since each iteration adds one vertex to S , it follows that the algorithm halts, at which point $S = V$, and thus, by invariant (1.), the function $L : V \rightarrow \mathbb{N} \cup \{\infty\}$ is the correct answer.

Remarks on Dijkstra's Algorithm

- If Dijkstra's algorithm is implemented naively, it has running time $O(n^2)$, where $n = |V|$.
- With clever data structures (e.g., so called “Fibonacci Heaps”) Dijkstra's algorithm can be implemented much more efficiently: essentially in time $O(m + n \log n)$ where, $n = |V|$ and $m = |E|$.

This increased efficiency can make a **big difference** on huge “sparse” graphs, where m is much smaller than n^2 (e.g., when out-degree is a fixed constant, $m \in O(n)$).

- Dijkstra's algorithm can be augmented to also output a description of a shortest path from the source vertex s to every other vertex v .

We will not describe these extensions, and we will certainly not assume that you know them.

Graph Colouring

Graph Colouring

Suppose we have k distinct colours with which to colour the vertices of a graph. Let $[k] = \{1, \dots, k\}$. For an undirected graph, $G = (V, E)$, an admissible vertex **k -colouring** of G is a function $c : V \rightarrow [k]$, such that for all $u, v \in V$, if $\{u, v\} \in E$ then $c(u) \neq c(v)$.

For an integer $k \geq 1$, we say an undirected graph $G = (V, E)$ is **k -colourable** if there exists a k -colouring of G .

The **chromatic number** of G , denoted $\chi(G)$, is the *smallest positive integer k* , such that G is k -colourable.

Some observations about Graph colouring

- Note that any graph G with n vertices is n -colourable.
- The **n -Clique**, K_n , i.e., the complete graph on n vertices, has chromatic number $\chi(K_n) = n$. All its vertices must get assigned different colours in any admissible colouring.
- The **clique number**, $\omega(G)$, of a graph G is the maximum positive integer $r \geq 1$, such that K_r is a subgraph of G .
- Note that for all graphs G , $\omega(G) \leq \chi(G)$: if G has an r -clique then it is not $(r - 1)$ -colorable.
- However, in general, $\omega(G) \neq \chi(G)$. For instance, The 5-cycle, C_5 , has $\omega(C_5) = 2 < \chi(C_5) = 3$.

More observations about colouring

- As already mentioned, any bipartite graph is 2-colourable. Indeed, that is an equivalent definition of being bipartite.
- More generally, a graph G is k -colourable precisely if it is **k -partite**, meaning its vertices can be partitioned into k disjoint sets such that all edges of the graph are between nodes in different parts.

Algorithms/complexity of colouring graphs

To determine whether a n -vertex graph $G = (V, E)$ is k -colourable by “*brute force*”, we could try all possible colourings of n nodes with k colours.

Difficulty: There are k^n such k -colouring functions $c : V \rightarrow [k]$.

Question: Is there an efficient (polynomial time) algorithm for determining whether a given graph G is k -colourable?

Algorithms/complexity of colouring graphs

To determine whether a n -vertex graph $G = (V, E)$ is k -colourable by “*brute force*”, we could try all possible colourings of n nodes with k colours.

Difficulty: There are k^n such k -colouring functions $c : V \rightarrow [k]$.

Question: Is there an efficient (polynomial time) algorithm for determining whether a given graph G is k -colourable?

Answer: No, no generally efficient (polynomial time) algorithm is known, and even the problem of determining whether a given graph is 3-colourable is **NP-complete**. (Even approximating the chromatic number of a given graph is NP-hard.)

In practice, there are heuristic algorithms that do obtain good colourings for many classes of graphs.

Applications of Graph Colouring (many)

Final Exam Scheduling

- There are n courses, $\{1, \dots, n\}$.
- Some courses have the same students registered for both, so their exams can't be scheduled at the same time.
- Let $G = (\{1, \dots, n\}, E)$ be a graph such that $\{i, j\} \in E$ if and only if $i \neq j$ and courses i and j have a student in common.
- **Question:** What is the minimum number of exam time slots needed to schedule all n exams?
- **Answer:** This is precisely the chromatic number $\chi(G)$ of G .

Furthermore, a k -colouring of G yields an *admissible schedule* of exams into k time slots, allowing all students to attend all their exams, as long as different “colors” are scheduled in disjoint time slots.

Discrete Mathematics & Mathematical Reasoning

Chapter 11: Trees

Kousha Etessami

U. of Edinburgh, UK

A **tree** is a connected simple undirected graph with no simple circuits.

A **forest** is a (not necessarily connected) simple undirected graph with no simple circuits.

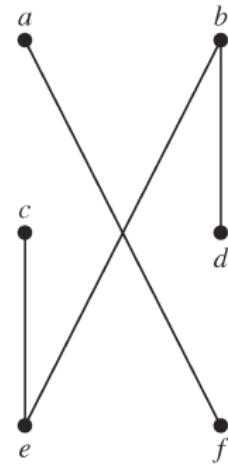
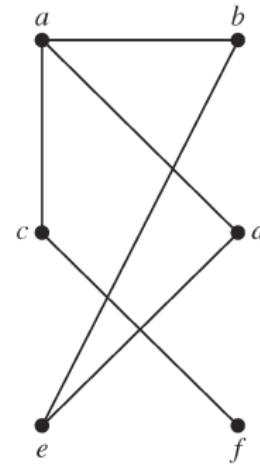
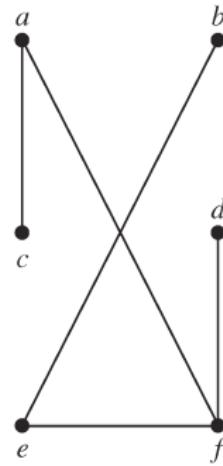
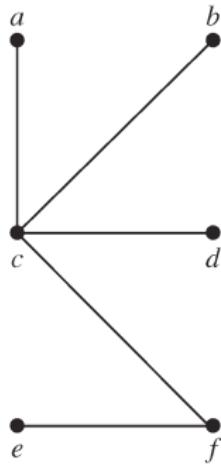


FIGURE 2 Examples of Trees and Graphs That Are Not Trees.

Some important facts about trees

Theorem 1: A graph G is a tree if and only if there is a **unique** simple (and tidy) path between any two vertices of G .

Proof: On the board. (Next slide provides written proof.) □

Theorem 2: Every tree, $T = (V, E)$ with $|V| \geq 2$, has at least two vertices that have degree $= 1$.

Proof: Take any **longest** simple path $x_0 \dots x_m$ in T . Both x_0 and x_m must have degree 1: otherwise there's a longer path in T . □

Theorem 3: Every tree with n vertices has exactly $n - 1$ edges.

Proof: On the board. By induction on n . □

Proof of Theorem 1 about Trees

Suppose there are two distinct simple paths between vertices $u, v \in V$: $x_0x_1x_2 \dots x_n$ and $y_0y_1y_2 \dots y_m$.

Firstly, there must be some $i \geq 1$, such that $\forall 0 \leq k < i$, $x_k = y_k$, but such that $x_i \neq y_i$. (Why is this so?)

Furthermore, there must be a smallest $j \geq i$, such that either x_j appears in y_i, \dots, y_m , or such that y_j appears in $x_i \dots x_n$.

Suppose, without loss of generality, that this holds for some smallest $j \geq i$ and x_j . Then $x_j = y_r$, for some smallest $r \geq i$.

We claim that then the path $x_{i-1}x_i \dots x_jy_{r-1}y_{r-2} \dots y_iy_{i-1}$ must form a simple circuit, which **contradicts** the fact that G is a tree.

Note that by assumption $x_{i-1} = y_{i-1}$, so this is a circuit.

Furthermore, it is simple, because its edges are a disjoint union of edges from the x and y paths, because by construction none of the vertices x_i, \dots, x_j occur in $y_i \dots y_{r-1}$, and $x_i \neq y_i$. □

Rooted Trees

A **rooted tree**, is a pair (T, r) where $T = (V, E)$ is a tree, and $r \in V$ is a chosen **root** vertex of the tree.

Often, the edges of a rooted tree (T, r) are viewed as being directed, such that for every vertex v the unique path from r to v is directed *away from* (or *towards*) r .

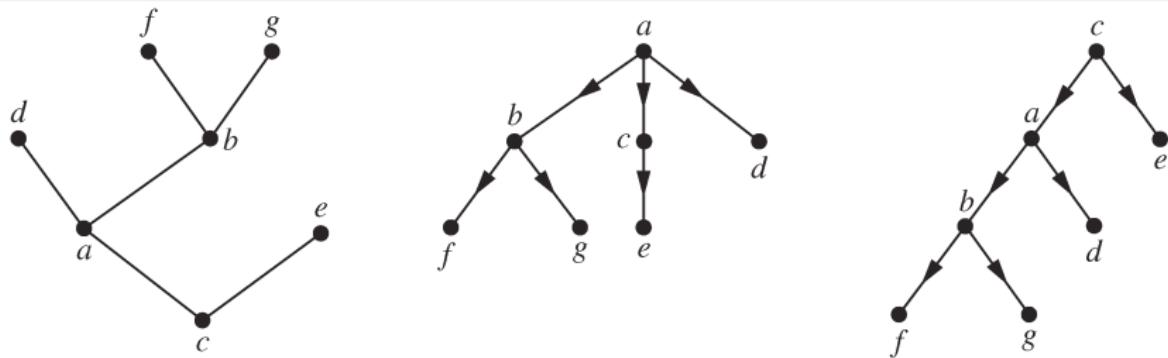


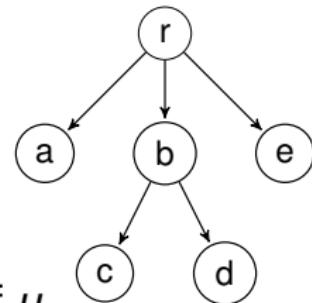
FIGURE 4 A Tree and Rooted Trees Formed by Designating Two Different Roots.

(In CS, rooted trees are typically drawn with **root at the top**.)

Terminology for rooted trees

For a rooted tree (T, r) , with root r ,

- For each node $v \neq r$ the **parent**, is the unique vertex u such that $(u, v) \in E$. v is then called a **child** of u .
Two vertices with the same parent are called **siblings**.
- A **leaf** is a vertex with no children. Non-leaves are called **internal vertices**.
- The **height** of a rooted tree is the length of the longest directed path from the root to any leaf.
- The **ancestors** (**descendants**, respectively) of a vertex v are all vertices $u \neq v$ such that there is a directed path from u to v (from v to u , respectively).
- The **subtree** rooted at v , is the subgraph containing v and all its descendants, and all directed edges between them.



m -ary Trees

For $m \geq 1$, A rooted tree is called a **m -ary tree** if every internal node has at most m children.

It is called a **full m -ary tree** if every internal node has exactly m children.

An m -ary tree with $m = 2$ is called a **binary tree**.

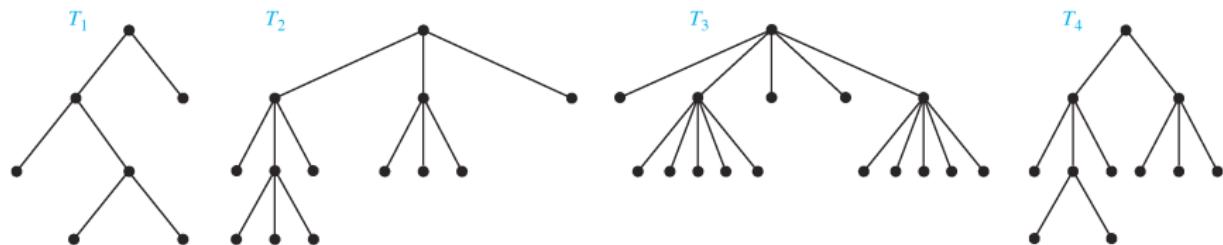


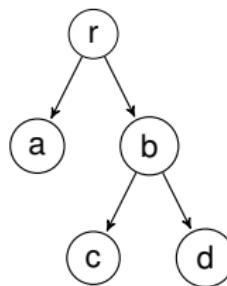
FIGURE 7 Four Rooted Trees.

Which one of these rooted trees is a (full) m -ary tree?

A **rooted ordered tree** is a rooted tree (T, r) where in addition the children of each internal vertex v are linearly ordered according to some ordering \leq_v .

When drawing the tree, we usually write ordered children (from least to greatest) from left to right.

If the rooted ordered tree is a **binary** tree, then the first child is called **left child** and the second child is called **right child**.



Note: rooted ordered trees are **VERY COMMON** in computer science applications: **parse trees**, **XML documents**, **file directories**, “**decision trees**”, “**game trees**”, . . .

Counting vertices in m -ary trees

Theorem C1: For all $m \geq 1$, every full m -ary tree with i internal vertices has exactly $n = m \cdot i + 1$ vertices.

Proof: Every vertex other than the root is a child of an internal vertex. There are thus $m \cdot i$ such children, plus 1 root. □

Theorem C2: For all $m \geq 1$, a full m -ary tree with:

1. n vertices has $i = (n - 1)/m$ internal vertices and $l = [(m - 1)n + 1]/m$ leaves.
2. i internal vertices has $n = m \cdot i + 1$ vertices and $l = (m - 1)i + 1$ leaves.
3. if $m \geq 2$, then if the m -ary tree has l leaves then it has $n = (ml - 1)/(m - 1)$ vertices and $i = (l - 1)/(m - 1)$ internal vertices.

More counting for m -ary trees

Theorem C3: There are at most m^h leaves in an m -ary tree of height h .

Proof: By induction on $h \geq 0$. □

Theorem C4: If an m -ary tree has l leaves, and h is its height, then $h \geq \lceil \log_m l \rceil$.

Proof: Since $l \leq m^h$, we have $\log_m l \leq h$. But h is a non-negative integer, so $\lceil \log_m l \rceil \leq h$. □

Application: bounds for comparison-based sorting

Question: You have to sort a list of distinct unknown numbers: a_1, \dots, a_n , using only the operation of comparing two numbers: $a_i \stackrel{?}{<} a_j$. How many comparisons do you need, in the worst case, in order to sort all the numbers correctly?

Application: bounds for comparison-based sorting

Question: You have to sort a list of distinct unknown numbers: a_1, \dots, a_n , using only the operation of comparing two numbers: $a_i \stackrel{?}{<} a_j$. How many comparisons do you need, in the worst case, in order to sort all the numbers correctly?

Answer: Consider a binary **decision tree**, modeling the comparisons you make. There are $n!$ permutations of a_1, \dots, a_n , so there are $n!$ possible fully sorted orderings. These constitute the leaves of your decision tree.

Since the decision tree is binary (2-ary), by Theorem C4 the height of the tree is $h \geq \log_2 n!$. But note that the height is the worst-case number of comparisons.

By Stirling's formula, $h \geq \log_2 \left(\frac{n}{e}\right)^n = \Omega(n \log_2 n)$.



Spanning Trees of undirected graphs

For a simple undirected graph G , a **spanning tree** of G is a subgraph T of G such that T is a tree and T contains every vertex of G .

Theorem: Every connected graph G has a spanning tree.

Proof: While there is a circuit in G , remove one edge of the circuit. Repeat. Removing one edge of the circuit does not change connectivity, and eventually no circuits can remain (because there are only finitely many edges to be removed). So, the end result is a tree which is a subtree of G . □

Spanning Trees of undirected graphs

For a simple undirected graph G , a **spanning tree** of G is a subgraph T of G such that T is a tree and T contains every vertex of G .

Theorem: Every connected graph G has a spanning tree.

Proof: While there is a circuit in G , remove one edge of the circuit. Repeat. Removing one edge of the circuit does not change connectivity, and eventually no circuits can remain (because there are only finitely many edges to be removed). So, the end result is a tree which is a subtree of G . □

Question: Given a graph G , can we efficiently compute a spanning tree for G ?

Spanning Trees of undirected graphs

For a simple undirected graph G , a **spanning tree** of G is a subgraph T of G such that T is a tree and T contains every vertex of G .

Theorem: Every connected graph G has a spanning tree.

Proof: While there is a circuit in G , remove one edge of the circuit. Repeat. Removing one edge of the circuit does not change connectivity, and eventually no circuits can remain (because there are only finitely many edges to be removed). So, the end result is a tree which is a subtree of G . □

Question: Given a graph G , can we efficiently compute a spanning tree for G ?

Answer: Yes. Even for edge-weighted graphs, we can compute a **minimum-cost spanning tree** efficiently. (The cost of a spanning tree is the sum of its edge costs.)

Prim's algorithm for a minimum spanning tree

Input: Connected, edge-weighted, undirected graph
 $G = (V, E, w)$.

Output: A minimum-cost spanning tree T for G .

Algorithm:

Initialize: $T := \{e\}$, where e is a minimum-weight edge in E .

for $i := 1$ to $n - 2$ **do**

Let $e' :=$ a minimum-weight edge incident to
some vertex in T , and not forming a circuit
if added to T ;

$T := T \cup \{e'\}$;

end for

Output the tree T .

Discrete Mathematics & Mathematical Reasoning

Chapter 7:

Discrete Probability

Kousha Etessami

U. of Edinburgh, UK

Overview of the Chapter

- Sample spaces, events, and probability distributions.
- Independence, conditional probability
- Bayes' Theorem and applications
- Random variables and expectation; linearity of expectation; variance
- Markov's and Chebyshev's inequalities.
- Examples from important probability distributions

Today's Lecture:

- Introduction to Discrete Probability (sections 7.1 and 7.2).

The “sample space” of a probabilistic experiment

Consider the following probabilistic (random) experiment:

“Flip a fair coin 7 times in a row, and see what happens”

Question: What are the **possible outcomes** of this experiment?

The “sample space” of a probabilistic experiment

Consider the following probabilistic (random) experiment:

“Flip a fair coin 7 times in a row, and see what happens”

Question: What are the **possible outcomes** of this experiment?

Answer: The possible outcomes are all the sequences of “Heads” and “Tails”, of length 7. In other words, they are the set of strings $\Omega = \{H, T\}^7$.

The set $\Omega = \{H, T\}^7$ of possible outcomes is called the **sample space** associated with this probabilistic experiment.

Sample Spaces

For any probabilistic experiment or process, the set Ω of all its possible outcomes is called its **sample space**.

In general, sample spaces need not be finite, and **they need not even be countable**. In “**Discrete Probability**”, we focus on finite and countable sample spaces. This simplifies the axiomatic treatment needed to do probability theory. We only consider discrete probability (and mainly finite sample spaces).

Question: What is the sample space, Ω , for the following probabilistic experiment:

“Flip a fair coin repeatedly until it comes up heads.”

Sample Spaces

For any probabilistic experiment or process, the set Ω of all its possible outcomes is called its **sample space**.

In general, sample spaces need not be finite, and **they need not even be countable**. In “**Discrete Probability**”, we focus on finite and countable sample spaces. This simplifies the axiomatic treatment needed to do probability theory. We only consider discrete probability (and mainly finite sample spaces).

Question: What is the sample space, Ω , for the following probabilistic experiment:

“Flip a fair coin repeatedly until it comes up heads.”

Answer: $\Omega = \{H, TH, TTH, TTTH, TTTTH, \dots\} = T^*H$.

Note: This set is **not** finite. So, even for simple random experiments we do have to consider **countable** sample spaces.

Probability distributions

A **probability distribution** over a finite or countable set Ω , is a function:

$$P : \Omega \rightarrow [0, 1]$$

such that $\sum_{s \in \Omega} P(s) = 1$.

In other words, to each outcome $s \in \Omega$, $P(s)$ assigns a probability, such that $0 \leq P(s) \leq 1$, and of course such that the probabilities of all outcomes sum to 1, so $\sum_{s \in \Omega} P(s) = 1$.

Simple examples of probability distributions

Example 1: Suppose a fair coin is tossed 7 times consecutively.

This random experiment defines a probability distribution

$P : \Omega \rightarrow [0, 1]$, on $\Omega = \{H, T\}^7$, where, for all $s \in \Omega$, $P(s) = 1/2^7$.
and $|\Omega| = 2^7$, so $\sum_{s \in \Omega} P(s) = 2^7 \cdot (1/2^7) = 1$.

Example 2: Suppose a fair coin is tossed repeatedly until it lands heads. This random experiment defines a probability distribution $P : \Omega \rightarrow [0, 1]$, on $\Omega = T^*H$, such that, for all $k \geq 0$,

$$P(T^k H) = \frac{1}{2^{k+1}}$$

Note that

$$\sum_{s \in \Omega} P(s) = P(H) + P(TH) + P(TTH) + \dots = \sum_{k=1}^{\infty} \frac{1}{2^k} = 1.$$

Events

For a **countable** sample space Ω , an **event**, E , is simply a subset $E \subseteq \Omega$ of the set of possible outcomes.

Given a probability distribution $P : \Omega \rightarrow [0, 1]$, we define **the probability of the event** $E \subseteq \Omega$ to be $P(E) \doteq \sum_{s \in E} P(s)$.

Example: For $\Omega = \{H, T\}^7$, the following are events:

- “The third coin toss came up heads”.
This is event $E_1 = \{H, T\}^2 H \{H, T\}^4$; $P(E_1) = (1/2)$.
- “the fourth and fifth coin tosses did not both come up tails”.
This is $E_2 = \Omega - \{H, T\}^3 TT \{H, T\}^2$; $P(E_2) = 1 - 1/4 = 3/4$.

Example: For $\Omega = T^* H$, the following are events:

- “The first time the coin comes up heads is after an even number of coin tosses.”
This is $E_3 = \{T^k H \mid k \text{ is odd}\}$; $P(E_3) = \sum_{k=1}^{\infty} (1/2^{2k}) = 1/3$.

Basic facts about probabilities of events

For event $E \subseteq \Omega$, define the **complement event** to be $\bar{E} \doteq \Omega - E$.

Theorem: Suppose E_0, E_1, E_2, \dots are a (finite or countable) sequence of pairwise disjoint events from the sample space Ω . In other words, $E_i \in \Omega$, and $E_i \cap E_j = \emptyset$ for all $i, j \in \mathbb{N}$. Then

$$P\left(\bigcup_i E_i\right) = \sum_i P(E_i)$$

Furthermore, for each event $E \subseteq \Omega$, $P(\bar{E}) = 1 - P(E)$.

Proof: Follows easily from definitions:

for each E_i , $P(E_i) = \sum_{s \in E_i} P(s)$, thus, since the sets E_i are disjoint, $P\left(\bigcup_i E_i\right) = \sum_{s \in \bigcup_i E_i} P(s) = \sum_i \sum_{s \in E_i} P(s) = \sum_i P(E_i)$.

Likewise, since $P(\Omega) = \sum_{s \in \Omega} P(s) = 1$, $P(\bar{E}) = P(\Omega - E) = \sum_{s \in \Omega - E} P(s) = \sum_{s \in \Omega} P(s) - \sum_{s \in E} P(s) = 1 - P(E)$.



Brief comment about non-discrete probability theory

In general (non-discrete) probability theory, with uncountable sample space Ω , the conditions of the prior theorem are actually taken as **axioms** about a “**probability measure**”, P , that maps events to probabilities, and events are not arbitrary subsets of Ω . Rather, the axioms say: Ω is an event; If E_0, E_1, \dots , are events, then so is $\bigcup_i E_i$; and If E is an event, then so is $\overline{E} = \Omega - E$.

A set of events $\mathcal{F} \subseteq 2^\Omega$ with these properties is called a **σ -algebra**. General probability theory studies **probability spaces** consisting of a triple (Ω, \mathcal{F}, P) , where Ω is a set, $\mathcal{F} \subseteq 2^\Omega$ is a σ -algebra of events over Ω , and $P : \mathcal{F} \rightarrow [0, 1]$ is a probability measure, defined to have the properties in the prior theorem.

We only discuss discrete probability, and will not assume you know definitions for general (non-discrete) probability.

Conditional probability

Definition: Let $P : \Omega \rightarrow [0, 1]$ be a probability distribution, and let $E, F \subseteq \Omega$ be two events, such that $P(F) > 0$.

The **conditional probability** of E given F , denoted $P(E | F)$, is defined by:

$$P(E | F) = \frac{P(E \cap F)}{P(F)}$$

Example: A fair coin is flipped three times. Suppose we know that the event F = “heads came up exactly once” occurs. what is the probability that of the event E = “the first coin flip came up heads” occurs?

Answer: There are 8 flip sequences $\{H, T\}^3$, all with probability $1/8$. The event that “heads came up exactly once” is $F = \{HTT, THT, TTH\}$. The event $E \cap F = \{HTT\}$.

$$\text{So, } P(E | F) = \frac{P(E \cap F)}{P(F)} = \frac{1/8}{3/8} = \frac{1}{3}.$$

Independence of two events

Intuitively, two events are *independent* if knowing whether one occurred does not alter the probability of the other. Formally:

Definition: Events A and B are called **independent** if
 $P(A \cap B) = P(A)P(B)$.

Note that if $P(B) > 0$ then A and B are independent if and only if

$$P(A | B) = \frac{P(A \cap B)}{P(B)} = P(A)$$

Thus, the probability of A is not altered by knowing B occurs.

Example: A fair coin is flipped three times. Are the events $A =$ “the first coin toss came up heads” and $B =$ “an even number of coin tosses came up head”, independent?

Independence of two events

Intuitively, two events are *independent* if knowing whether one occurred does not alter the probability of the other. Formally:

Definition: Events A and B are called **independent** if
 $P(A \cap B) = P(A)P(B)$.

Note that if $P(B) > 0$ then A and B are independent if and only if

$$P(A | B) = \frac{P(A \cap B)}{P(B)} = P(A)$$

Thus, the probability of A is not altered by knowing B occurs.

Example: A fair coin is flipped three times. Are the events $A =$ “the first coin toss came up heads” and $B =$ “an even number of coin tosses came up head”, independent?

Answer: Yes. $P(A \cap B) = 1/4$, $P(A) = 1/2$, and $P(B) = 1/2$, so $P(A \cap B) = P(A)P(B)$.

Pairwise and mutual independence

What if we have more than two events: E_1, E_2, \dots, E_n .
When should we consider them “independent”?

Pairwise and mutual independence

What if we have more than two events: E_1, E_2, \dots, E_n .
When should we consider them “independent”?

Definition: Events E_1, \dots, E_n are called **pairwise independent**, if for every pair $i, j \in \{1, \dots, n\}$, $i \neq j$, E_i and E_j are independent (i.e., $P(E_i \cap E_j) = P(E_i)P(E_j)$).

Events E_1, \dots, E_n are called **mutually independent**, if for every subset $J \subseteq \{1, \dots, n\}$, $P(\bigcap_{j \in J} E_j) = \prod_{j \in J} P(E_j)$.

Clearly, mutual independence implies pairwise independent.
But... **Warning:** pairwise independence **does not** imply mutual independence. (A tutorial sheet exercise asks you to prove this.)
Typically, when we refer to > 2 events as “independent”, we mean they are “mutually independent”.

Biased coins and Bernoulli trials

In probability theory there are a number of fundamental probability distributions that one should study and understand in detail.

One of these distributions arises from (repeatedly) flipping a biased coin.

A **Bernoulli trial** is a probabilistic experiment that has two outcomes: **success** or **failure** (e.g., heads or tails).

We suppose that p is the probability of success, and $q = (1 - p)$ is the probability of failure.

We can of course have repeated Bernoulli trials. We typically assume the different trials are mutually independent.

Question: A biased coin, which comes up heads with probability $p = 2/3$, is flipped 7 times consecutively. What is the probability that it comes up heads exactly 4 times?

The Binomial Distribution

Theorem: The probability of exactly k successes in n (mutually) independent Bernoulli trials, with probability p of success and $q = (1 - p)$ of failure in each trial, is

$$\binom{n}{k} p^k q^{n-k}$$

Proof: We can associate n Bernoulli trials with outcomes $\Omega = \{H, T\}^n$. Each sequence $s = (s_1, \dots, s_n)$ with exactly k heads and $n - k$ tails occurs with probability $p^k q^{n-k}$. There are $\binom{n}{k}$ such sequences with exactly k heads. □

Definition: The **binomial distribution**, with parameters n and p , denoted $b(k; n, p)$, defines a probability distribution on $k \in \{0, \dots, n\}$, given by

$$b(k; n, p) \doteq \binom{n}{k} \cdot p^k q^{n-k}$$

Random variables

Definition: A **random variable**, is a function $X : \Omega \rightarrow \mathbb{R}$, that assigns a real value to each outcome in a sample space Ω .

Example: Suppose a biased coin is flipped n times. The sample space is $\Omega = \{H, T\}^n$. The function $X : \Omega \rightarrow \mathbb{N}$ that assigns to each outcome $s \in \Omega$ the number $X(s) \in \mathbb{N}$ of coin tosses that came up heads is one random variable.

For a random variable $X : \Omega \rightarrow \mathbb{R}$, we write $P(X = r)$ as shorthand for the probability $P(\{s \in \Omega \mid X(s) = r\})$. The **distribution** of a random variable X is given by the set of pairs $\{(r, P(X = r)) \mid r \text{ is in the range of } X\}$.

Note: These definitions of a random variable and its distribution are only adequate in the context of **discrete** probability distributions. For general probability theory we need more elaborate definitions.

Biased coins and the Geometric Distribution

Question: Suppose a biased coin, comes up heads with probability p , $0 < p < 1$, each time it is tossed. Suppose we repeatedly flip this coin until it comes up heads. What is the probability that we flip the coin k times, for $k \geq 1$?

Biased coins and the Geometric Distribution

Question: Suppose a biased coin, comes up heads with probability p , $0 < p < 1$, each time it is tossed. Suppose we repeatedly flip this coin until it comes up heads. What is the probability that we flip the coin k times, for $k \geq 1$?

Answer: The sample space is $\Omega = \{H, TH, TTH, \dots\}$. Assuming mutual independence of coin flips, the probability of $T^{k-1}H$ is $(1 - p)^{k-1}p$. Note: this does define a probability distribution on $k \geq 1$, because

$$\sum_{k=1}^{\infty} (1 - p)^{k-1}p = p \sum_{k=0}^{\infty} (1 - p)^k = p(1/p) = 1.$$

□

A random variable $X : \Omega \rightarrow \mathbb{N}$, is said to have a **geometric distribution** with parameter p , $0 \leq p \leq 1$, if for all positive integers $k \geq 1$, $P(X = k) = (1 - p)^{k-1}p$.

Discrete Mathematics & Mathematical Reasoning

Chapter 7:

Discrete Probability

Kousha Etessami

U. of Edinburgh, UK

Overview of the Chapter

- Sample spaces, events, and probability distributions.
- Independence, conditional probability
- Bayes' Theorem and applications
- Random variables and expectation; linearity of expectation; variance
- Markov's and Chebyshev's inequalities.
- Examples from important probability distributions

Today's Lecture:

- Introduction to Discrete Probability (sections 7.1 and 7.2).

The “sample space” of a probabilistic experiment

Consider the following probabilistic (random) experiment:

“Flip a fair coin 7 times in a row, and see what happens”

Question: What are the **possible outcomes** of this experiment?

The “sample space” of a probabilistic experiment

Consider the following probabilistic (random) experiment:

“Flip a fair coin 7 times in a row, and see what happens”

Question: What are the **possible outcomes** of this experiment?

Answer: The possible outcomes are all the sequences of “Heads” and “Tails”, of length 7. In other words, they are the set of strings $\Omega = \{H, T\}^7$.

The set $\Omega = \{H, T\}^7$ of possible outcomes is called the **sample space** associated with this probabilistic experiment.

Sample Spaces

For any probabilistic experiment or process, the set Ω of all its possible outcomes is called its **sample space**.

In general, sample spaces need not be finite, and **they need not even be countable**. In “**Discrete Probability**”, we focus on finite and countable sample spaces. This simplifies the axiomatic treatment needed to do probability theory. We only consider discrete probability (and mainly finite sample spaces).

Question: What is the sample space, Ω , for the following probabilistic experiment:

“Flip a fair coin repeatedly until it comes up heads.”

Sample Spaces

For any probabilistic experiment or process, the set Ω of all its possible outcomes is called its **sample space**.

In general, sample spaces need not be finite, and **they need not even be countable**. In “**Discrete Probability**”, we focus on finite and countable sample spaces. This simplifies the axiomatic treatment needed to do probability theory. We only consider discrete probability (and mainly finite sample spaces).

Question: What is the sample space, Ω , for the following probabilistic experiment:

“Flip a fair coin repeatedly until it comes up heads.”

Answer: $\Omega = \{H, TH, TTH, TTTH, TTTTH, \dots\} = T^*H$.

Note: This set is **not** finite. So, even for simple random experiments we do have to consider **countable** sample spaces.

Probability distributions

A **probability distribution** over a finite or countable set Ω , is a function:

$$P : \Omega \rightarrow [0, 1]$$

such that $\sum_{s \in \Omega} P(s) = 1$.

In other words, to each outcome $s \in \Omega$, $P(s)$ assigns a probability, such that $0 \leq P(s) \leq 1$, and of course such that the probabilities of all outcomes sum to 1, so $\sum_{s \in \Omega} P(s) = 1$.

Simple examples of probability distributions

Example 1: Suppose a fair coin is tossed 7 times consecutively.

This random experiment defines a probability distribution

$P : \Omega \rightarrow [0, 1]$, on $\Omega = \{H, T\}^7$, where, for all $s \in \Omega$, $P(s) = 1/2^7$.
and $|\Omega| = 2^7$, so $\sum_{s \in \Omega} P(s) = 2^7 \cdot (1/2^7) = 1$.

Example 2: Suppose a fair coin is tossed repeatedly until it lands heads. This random experiment defines a probability distribution $P : \Omega \rightarrow [0, 1]$, on $\Omega = T^*H$, such that, for all $k \geq 0$,

$$P(T^k H) = \frac{1}{2^{k+1}}$$

Note that

$$\sum_{s \in \Omega} P(s) = P(H) + P(TH) + P(TTH) + \dots = \sum_{k=1}^{\infty} \frac{1}{2^k} = 1.$$

Events

For a **countable** sample space Ω , an **event**, E , is simply a subset $E \subseteq \Omega$ of the set of possible outcomes.

Given a probability distribution $P : \Omega \rightarrow [0, 1]$, we define **the probability of the event** $E \subseteq \Omega$ to be $P(E) \doteq \sum_{s \in E} P(s)$.

Example: For $\Omega = \{H, T\}^7$, the following are events:

- “The third coin toss came up heads”.
This is event $E_1 = \{H, T\}^2 H \{H, T\}^4$; $P(E_1) = (1/2)$.
- “the fourth and fifth coin tosses did not both come up tails”.
This is $E_2 = \Omega - \{H, T\}^3 TT \{H, T\}^2$; $P(E_2) = 1 - 1/4 = 3/4$.

Example: For $\Omega = T^* H$, the following are events:

- “The first time the coin comes up heads is after an even number of coin tosses.”
This is $E_3 = \{T^k H \mid k \text{ is odd}\}$; $P(E_3) = \sum_{k=1}^{\infty} (1/2^{2k}) = 1/3$.

Basic facts about probabilities of events

For event $E \subseteq \Omega$, define the **complement event** to be $\bar{E} \doteq \Omega - E$.

Theorem: Suppose E_0, E_1, E_2, \dots are a (finite or countable) sequence of pairwise disjoint events from the sample space Ω . In other words, $E_i \in \Omega$, and $E_i \cap E_j = \emptyset$ for all $i, j \in \mathbb{N}$. Then

$$P\left(\bigcup_i E_i\right) = \sum_i P(E_i)$$

Furthermore, for each event $E \subseteq \Omega$, $P(\bar{E}) = 1 - P(E)$.

Proof: Follows easily from definitions:

for each E_i , $P(E_i) = \sum_{s \in E_i} P(s)$, thus, since the sets E_i are disjoint, $P\left(\bigcup_i E_i\right) = \sum_{s \in \bigcup_i E_i} P(s) = \sum_i \sum_{s \in E_i} P(s) = \sum_i P(E_i)$.

Likewise, since $P(\Omega) = \sum_{s \in \Omega} P(s) = 1$, $P(\bar{E}) = P(\Omega - E) = \sum_{s \in \Omega - E} P(s) = \sum_{s \in \Omega} P(s) - \sum_{s \in E} P(s) = 1 - P(E)$.



Brief comment about non-discrete probability theory

In general (non-discrete) probability theory, with uncountable sample space Ω , the conditions of the prior theorem are actually taken as **axioms** about a “**probability measure**”, P , that maps events to probabilities, and events are not arbitrary subsets of Ω . Rather, the axioms say: Ω is an event; If E_0, E_1, \dots , are events, then so is $\bigcup_i E_i$; and If E is an event, then so is $\overline{E} = \Omega - E$.

A set of events $\mathcal{F} \subseteq 2^\Omega$ with these properties is called a **σ -algebra**. General probability theory studies **probability spaces** consisting of a triple (Ω, \mathcal{F}, P) , where Ω is a set, $\mathcal{F} \subseteq 2^\Omega$ is a σ -algebra of events over Ω , and $P : \mathcal{F} \rightarrow [0, 1]$ is a probability measure, defined to have the properties in the prior theorem.

We only discuss discrete probability, and will not assume you know definitions for general (non-discrete) probability.

Conditional probability

Definition: Let $P : \Omega \rightarrow [0, 1]$ be a probability distribution, and let $E, F \subseteq \Omega$ be two events, such that $P(F) > 0$.

The **conditional probability** of E given F , denoted $P(E | F)$, is defined by:

$$P(E | F) = \frac{P(E \cap F)}{P(F)}$$

Example: A fair coin is flipped three times. Suppose we know that the event F = “heads came up exactly once” occurs. what is the probability that of the event E = “the first coin flip came up heads” occurs?

Answer: There are 8 flip sequences $\{H, T\}^3$, all with probability $1/8$. The event that “heads came up exactly once” is $F = \{HTT, THT, TTH\}$. The event $E \cap F = \{HTT\}$.

$$\text{So, } P(E | F) = \frac{P(E \cap F)}{P(F)} = \frac{1/8}{3/8} = \frac{1}{3}.$$

Independence of two events

Intuitively, two events are *independent* if knowing whether one occurred does not alter the probability of the other. Formally:

Definition: Events A and B are called **independent** if
 $P(A \cap B) = P(A)P(B)$.

Note that if $P(B) > 0$ then A and B are independent if and only if

$$P(A | B) = \frac{P(A \cap B)}{P(B)} = P(A)$$

Thus, the probability of A is not altered by knowing B occurs.

Example: A fair coin is flipped three times. Are the events $A =$ “the first coin toss came up heads” and $B =$ “an even number of coin tosses came up head”, independent?

Independence of two events

Intuitively, two events are *independent* if knowing whether one occurred does not alter the probability of the other. Formally:

Definition: Events A and B are called **independent** if
 $P(A \cap B) = P(A)P(B)$.

Note that if $P(B) > 0$ then A and B are independent if and only if

$$P(A | B) = \frac{P(A \cap B)}{P(B)} = P(A)$$

Thus, the probability of A is not altered by knowing B occurs.

Example: A fair coin is flipped three times. Are the events $A =$ “the first coin toss came up heads” and $B =$ “an even number of coin tosses came up head”, independent?

Answer: Yes. $P(A \cap B) = 1/4$, $P(A) = 1/2$, and $P(B) = 1/2$, so $P(A \cap B) = P(A)P(B)$.

Pairwise and mutual independence

What if we have more than two events: E_1, E_2, \dots, E_n .
When should we consider them “independent”?

Pairwise and mutual independence

What if we have more than two events: E_1, E_2, \dots, E_n .
When should we consider them “independent”?

Definition: Events E_1, \dots, E_n are called **pairwise independent**, if for every pair $i, j \in \{1, \dots, n\}$, $i \neq j$, E_i and E_j are independent (i.e., $P(E_i \cap E_j) = P(E_i)P(E_j)$).

Events E_1, \dots, E_n are called **mutually independent**, if for every subset $J \subseteq \{1, \dots, n\}$, $P(\bigcap_{j \in J} E_j) = \prod_{j \in J} P(E_j)$.

Clearly, mutual independence implies pairwise independent.
But... **Warning:** pairwise independence **does not** imply mutual independence. (A tutorial sheet exercise asks you to prove this.)
Typically, when we refer to > 2 events as “independent”, we mean they are “mutually independent”.

Biased coins and Bernoulli trials

In probability theory there are a number of fundamental probability distributions that one should study and understand in detail.

One of these distributions arises from (repeatedly) flipping a biased coin.

A **Bernoulli trial** is a probabilistic experiment that has two outcomes: **success** or **failure** (e.g., heads or tails).

We suppose that p is the probability of success, and $q = (1 - p)$ is the probability of failure.

We can of course have repeated Bernoulli trials. We typically assume the different trials are mutually independent.

Question: A biased coin, which comes up heads with probability $p = 2/3$, is flipped 7 times consecutively. What is the probability that it comes up heads exactly 4 times?

The Binomial Distribution

Theorem: The probability of exactly k successes in n (mutually) independent Bernoulli trials, with probability p of success and $q = (1 - p)$ of failure in each trial, is

$$\binom{n}{k} p^k q^{n-k}$$

Proof: We can associate n Bernoulli trials with outcomes $\Omega = \{H, T\}^n$. Each sequence $s = (s_1, \dots, s_n)$ with exactly k heads and $n - k$ tails occurs with probability $p^k q^{n-k}$. There are $\binom{n}{k}$ such sequences with exactly k heads. □

Definition: The **binomial distribution**, with parameters n and p , denoted $b(k; n, p)$, defines a probability distribution on $k \in \{0, \dots, n\}$, given by

$$b(k; n, p) \doteq \binom{n}{k} \cdot p^k q^{n-k}$$

Random variables

Definition: A **random variable**, is a function $X : \Omega \rightarrow \mathbb{R}$, that assigns a real value to each outcome in a sample space Ω .

Example: Suppose a biased coin is flipped n times. The sample space is $\Omega = \{H, T\}^n$. The function $X : \Omega \rightarrow \mathbb{N}$ that assigns to each outcome $s \in \Omega$ the number $X(s) \in \mathbb{N}$ of coin tosses that came up heads is one random variable.

For a random variable $X : \Omega \rightarrow \mathbb{R}$, we write $P(X = r)$ as shorthand for the probability $P(\{s \in \Omega \mid X(s) = r\})$. The **distribution** of a random variable X is given by the set of pairs $\{(r, P(X = r)) \mid r \text{ is in the range of } X\}$.

Note: These definitions of a random variable and its distribution are only adequate in the context of **discrete** probability distributions. For general probability theory we need more elaborate definitions.

Biased coins and the Geometric Distribution

Question: Suppose a biased coin, comes up heads with probability p , $0 < p < 1$, each time it is tossed. Suppose we repeatedly flip this coin until it comes up heads. What is the probability that we flip the coin k times, for $k \geq 1$?

Biased coins and the Geometric Distribution

Question: Suppose a biased coin, comes up heads with probability p , $0 < p < 1$, each time it is tossed. Suppose we repeatedly flip this coin until it comes up heads. What is the probability that we flip the coin k times, for $k \geq 1$?

Answer: The sample space is $\Omega = \{H, TH, TTH, \dots\}$. Assuming mutual independence of coin flips, the probability of $T^{k-1}H$ is $(1 - p)^{k-1}p$. Note: this does define a probability distribution on $k \geq 1$, because

$$\sum_{k=1}^{\infty} (1 - p)^{k-1}p = p \sum_{k=0}^{\infty} (1 - p)^k = p(1/p) = 1.$$

□

A random variable $X : \Omega \rightarrow \mathbb{N}$, is said to have a **geometric distribution** with parameter p , $0 \leq p \leq 1$, if for all positive integers $k \geq 1$, $P(X = k) = (1 - p)^{k-1}p$.

Discrete Mathematics & Mathematical Reasoning

Chapter 7 (section 7.3): Conditional Probability & Bayes' Theorem

Kousha Etessami

U. of Edinburgh, UK



Reverend [Thomas Bayes](#) (1701-1761),
studied [logic](#) and [theology](#) as an undergraduate student
at the University of Edinburgh from 1719-1722.

Bayes' Theorem

Bayes Theorem

Let A and B be two events from a (countable) sample space Ω , and $P : \Omega \rightarrow [0, 1]$ a probability distribution on Ω , such that $0 < P(A) < 1$, and $P(B) > 0$. Then

$$P(A | B) = \frac{P(B | A)P(A)}{P(B | A)P(A) + P(B | \bar{A})P(\bar{A})}$$

This may at first look like an obscure equation, but as we shall see, it is useful....

Proof of Bayes' Theorem:

Let A and B be events such that $0 < P(A) < 1$ and $P(B) > 0$.

By definition, $P(A | B) = \frac{P(A \cap B)}{P(B)}$. So: $P(A \cap B) = P(A | B)P(B)$.

Likewise, $P(B \cap A) = P(B | A)P(A)$.

Likewise, $P(B \cap \bar{A}) = P(B | \bar{A})P(\bar{A})$. (Note that $P(\bar{A}) > 0$.)

Note that $P(A | B)P(B) = P(A \cap B) = P(B | A)P(A)$. So,

$$P(A | B) = \frac{P(B | A)P(A)}{P(B)}$$

Furthermore,

$$\begin{aligned} P(B) &= P((B \cap A) \cup (B \cap \bar{A})) = P(B \cap A) + P(B \cap \bar{A}) \\ &= P(B | A)P(A) + P(B | \bar{A})P(\bar{A}) \end{aligned}$$

So: $P(A | B) = \frac{P(B | A)P(A)}{P(B | A)P(A) + P(B | \bar{A})P(\bar{A})}$. □

Using Bayes' Theorem

Problem: There are two boxes, Box B_1 and Box B_2 .

Box B_1 contains 2 red balls and 8 blue balls.

Box B_2 contains 7 red balls and 3 blue balls.

Suppose Jane first randomly chooses one of two boxes B_1 and B_2 , with equal probability, $1/2$, of choosing each.

Suppose Jane then randomly picks one ball out of the box she has chosen (without telling you which box she had chosen), and shows you the ball she picked.

Suppose you only see that the ball Jane picked is red.

Question: Given this information, what is the probability that Jane chose box B_1 ?

Using Bayes' Theorem, continued

Answer: The underlying sample space, Ω , is:

$$\Omega = \{(a, b) \mid a \in \{1, 2\}, b \in \{\text{red, blue}\}\}$$

Let $F = \{(a, b) \in \Omega \mid a = 1\}$ be the event that box B_1 was chosen. Thus, $\bar{F} = \Omega - F$ is the event that box B_2 was chosen.

Let $E = \{(a, b) \in \Omega \mid b = \text{red}\}$ be the event that a red ball was picked. Thus, \bar{E} is the event that a blue ball was picked.

We are interested in computing the probability $P(F \mid E)$.

We know that $P(E \mid F) = \frac{2}{10}$ and $P(E \mid \bar{F}) = \frac{7}{10}$.

We also know that: $P(F) = 1/2$ and $P(\bar{F}) = 1/2$.

Can we compute $P(F \mid E)$ based on this? Yes, using Bayes'.

Using Bayes' Theorem, continued

Note that, $0 < P(F) < 1$, and $P(E) > 0$.

By Bayes' Theorem:

$$\begin{aligned}P(F | E) &= \frac{P(E | F)P(F)}{P(E | F)P(F) + P(E | \bar{F})P(\bar{F})} \\&= \frac{(2/10) * (1/2)}{(2/10) * (1/2) + (7/10) * (1/2)} \\&= \frac{2/20}{2/20 + 7/20} = \frac{2}{9}. \quad \square\end{aligned}$$

Note that, without the information that a red ball was picked, the probability that Jane chose Box B_1 is $P(F) = 1/2$.
But given the information, E , that a red ball was picked, the probability becomes much less, changing to $P(F | E) = 2/9$.

More on using Bayes' Theorem: Bayesian Spam Filters

Problem: Suppose it has been observed empirically that the word “Congratulations” occurs in 1 out of 10 **spam** emails, but that “Congratulations” only occurs in 1 out of 1000 **non-spam** emails. Suppose it has also been observed empirically that about 4 out of 10 emails are spam.

In Bayesian Spam Filtering, these **empirical probabilities** are interpreted as genuine probabilities in order to help estimate the probability that an incoming email is spam.

Suppose we get a new email that contains “Congratulations”. Let C be the event that a new email contains “Congratulations”. Let S be the event that a new email is spam.

We have observed C . We want to know $P(S | C)$.

Bayesian spam filtering example, continued

Bayesian solution: By Bayes' Theorem:

$$P(S | C) = \frac{P(C | S)P(S)}{P(C | S)P(S) + P(C | \bar{S})P(\bar{S})}$$

From the “empirical probabilities”, we get the estimates:

$$P(C | S) \approx 1/10; \quad P(C | \bar{S}) \approx 1/1000;$$

$$P(S) \approx 4/10; \quad P(\bar{S}) \approx 6/10.$$

So, we estimate that:

$$\begin{aligned} P(S | C) &\approx \frac{(1/10)(4/10)}{(1/10)(4/10) + (1/1000) * (6/10)} \\ &\approx \frac{.04}{.0406} \approx 0.985 \end{aligned}$$

So, with “high probability”, such an email is spam. (However, **much caution is needed** when interpreting such “probabilities”.)

Generalized Bayes' Theorem

Suppose that E, F_1, \dots, F_n are events from sample space Ω , and that $P : \Omega \rightarrow [0, 1]$ is a probability distribution on Ω . Suppose that $\cup_{i=1}^n F_j = \Omega$, and that $F_i \cap F_j = \emptyset$ for all $i \neq j$.

Suppose $P(E) > 0$, and $P(F_j) > 0$ for all j . Then for all j :

$$P(F_j | E) = \frac{P(E | F_j)P(F_j)}{\sum_{i=1}^n P(E | F_i)P(F_i)}$$

Suppose Jane first randomly chooses a box from among n different boxes, B_1, \dots, B_n , and then randomly picks a coloured ball out of the box she chose. (Each Box may have different numbers of balls of each colour.)

We can use the *Generalized Bayes' Theorem* to calculate the probability that Jane chose box B_j (event F_j), given that the colour of the ball that Jane picked is red (event E).

Proof of Generalized Bayes' Theorem: Very similar to the proof of Bayes' Theorem. Observe that:

$$P(F_j | E) = \frac{P(F_j \cap E)}{P(E)} = \frac{P(E | F_j)P(F_j)}{P(E)}$$

So, we only need to show that $P(E) = \sum_{i=1}^n P(E | F_i)P(F_i)$.
But since $\bigcup_i F_i = \Omega$, and since $F_i \cap F_j = \emptyset$ for all $i \neq j$:

$$\begin{aligned} P(E) &= P\left(\bigcup_i (E \cap F_i)\right) \\ &= \sum_{i=1}^n P(E \cap F_i) \quad (\text{because } F_i\text{'s are disjoint}) \\ &= \sum_{i=1}^n P(E | F_i)P(F_i). \quad \square \end{aligned}$$

Discrete Mathematics & Mathematical Reasoning

Chapter 7 (section 7.4): Random Variables, Expectation, and Variance

Kousha Etessami

U. of Edinburgh, UK

Expected Value (Expectation) of a Random Variable

Recall: A **random variable (r.v.)**, is a function $X : \Omega \rightarrow \mathbb{R}$, that assigns a real value to each outcome in a sample space Ω .

The **expected value**, or **expectation**, or **mean**, of a random variable $X : \Omega \rightarrow \mathbb{R}$, denoted by $E(X)$, is defined by:

$$E(X) = \sum_{s \in \Omega} P(s)X(s)$$

Here $P : \Omega \rightarrow [0, 1]$ is the underlying probability distribution on Ω .

Question: Let X be the r.v. outputting the number that comes up when a **fair die** is rolled. What is the expected value, $E(X)$, of X ?

Expected Value (Expectation) of a Random Variable

Recall: A **random variable (r.v.)**, is a function $X : \Omega \rightarrow \mathbb{R}$, that assigns a real value to each outcome in a sample space Ω .

The **expected value**, or **expectation**, or **mean**, of a random variable $X : \Omega \rightarrow \mathbb{R}$, denoted by $E(X)$, is defined by:

$$E(X) = \sum_{s \in \Omega} P(s)X(s)$$

Here $P : \Omega \rightarrow [0, 1]$ is the underlying probability distribution on Ω .

Question: Let X be the r.v. outputting the number that comes up when a **fair die** is rolled. What is the expected value, $E(X)$, of X ?

Answer:

$$E(X) = \sum_{i=1}^6 \frac{1}{6} \cdot i = \frac{21}{6} = \frac{7}{2}. \quad \square$$

A bad way to calculate expectation

The definition of expectation, $E(X) = \sum_{s \in \Omega} P(s)X(s)$, can be used directly to calculate $E(X)$. But sometimes this is **horribly inefficient**.

Example: Suppose that a biased coin, which comes up heads with probability p each time, is flipped 11 times consecutively.

Question: What is the expected # of heads?

A bad way to calculate expectation

The definition of expectation, $E(X) = \sum_{s \in \Omega} P(s)X(s)$, can be used directly to calculate $E(X)$. But sometimes this is **horribly inefficient**.

Example: Suppose that a biased coin, which comes up heads with probability p each time, is flipped 11 times consecutively.

Question: What is the expected # of heads?

Bad way to answer this: Let's try to use the definition of $E(X)$ directly, with $\Omega = \{H, T\}^{11}$. Note that $|\Omega| = 2^{11} = 2048$.

So, the sum $\sum_{s \in \Omega} P(s)X(s)$ has **2048 terms!**

This is **clearly not** a practical way to compute $E(X)$.

Is there a better way? Yes.

Better expression for the expectation

Recall $P(X = r)$ denotes the probability $P(\{s \in \Omega \mid X(s) = r\})$.

Recall that for a function $X : \Omega \rightarrow \mathbb{R}$,

$$\text{range}(X) = \{r \in \mathbb{R} \mid \exists s \in \Omega \text{ such that } X(s) = r\}$$

Theorem: For a random variable $X : \Omega \rightarrow \mathbb{R}$,

$$E(X) = \sum_{r \in \text{range}(X)} P(X = r) \cdot r$$

Proof: $E(X) = \sum_{s \in \Omega} P(s)X(s)$, but for each $r \in \text{range}(X)$, if we sum all terms $P(s)X(s)$ such that $X(s) = r$, we get $P(X = r) \cdot r$ as their sum. So, summing over all $r \in \text{range}(X)$ we get

$$E(X) = \sum_{r \in \text{range}(X)} P(X = r) \cdot r.$$



So, if $|\text{range}(X)|$ is small, and if we can compute $P(X = r)$, then we need to sum a lot fewer terms to calculate $E(X)$.

Expected # of successes in n Bernoulli trials

Theorem: The expected # of successes in n (independent) Bernoulli trials, with probability p of success in each, is np .

Note: We'll see later that we do not need independence for this.

First, a proof which uses mutual independence: For $\Omega = \{H, T\}^n$, let $X : \Omega \rightarrow \mathbb{N}$ count the number of successes in n Bernoulli trials. Let $q = (1 - p)$. Then...

$$\begin{aligned} E(X) &= \sum_{k=0}^n P(X = k) \cdot k \\ &= \sum_{k=1}^n \binom{n}{k} p^k q^{n-k} \cdot k \end{aligned}$$

The second equality holds because, assuming mutual independence, $P(X = k)$ is the binomial distribution $b(k; n, p)$.

first proof continued

$$\begin{aligned} E(X) &= \sum_{k=0}^n P(X = k) \cdot k = \sum_{k=1}^n \binom{n}{k} p^k q^{n-k} \cdot k = \\ &= \sum_{k=1}^n \frac{n!}{k!(n-k)!} p^k q^{n-k} \cdot k = \sum_{k=1}^n \frac{n!}{(k-1)!(n-k)!} p^k q^{n-k} \\ &= \sum_{k=1}^n n \cdot \frac{(n-1)!}{(k-1)!(n-k)!} p^k q^{n-k} = n \sum_{k=1}^n \binom{n-1}{k-1} p^k q^{n-k} \\ &= np \sum_{k=1}^n \binom{n-1}{k-1} p^{k-1} q^{n-k} = np \sum_{j=0}^{n-1} \binom{n-1}{j} p^j q^{n-1-j} \\ &= np(p+q)^{n-1} \\ &= np . \quad \square \end{aligned}$$

We will soon see this was **an unnecessarily complicated proof.**

Expectation of a geometrically distributed r.v.

Question: A coin comes up heads with probability $p > 0$ each time it is flipped. The coin is flipped repeatedly until it comes up heads. What is the expected number of times it is flipped?

Note: This simply asks: “What is the expected value $E(X)$ of a geometrically distributed random variable with parameter p ?”

Answer: $\Omega = \{H, TH, TTH, \dots\}$, and $P(T^{k-1}H) = (1-p)^{k-1}p$. And clearly $X(T^{k-1}H) = k$. Thus $E(X) = \sum_{s \in \Omega} P(s)X(s) =$

$$E(X) = \sum_{k=1}^{\infty} (1-p)^{k-1}p \cdot k = p \sum_{k=1}^{\infty} k(1-p)^{k-1} = p \cdot \frac{1}{p^2} = \frac{1}{p}.$$

This is because: $\sum_{k=1}^{\infty} k \cdot x^{k-1} = \frac{1}{(1-x)^2}$, for $|x| < 1$. □

Example: If $p = 1/4$, then the expected number of coin tosses before we see Heads for the first time is 4.

Linearity of Expectation (**VERY IMPORTANT**)

Theorem (Linearity of Expectation): For any random variables X, X_1, \dots, X_n on Ω , $E(X_1 + X_2 + \dots + X_n) = E(X_1) + \dots + E(X_n)$.

Furthermore, for any $a, b \in \mathbb{R}$,

$$E(aX + b) = aE(X) + b.$$

(In other words, the expectation function is a **linear function**.)

Proof:

$$E\left(\sum_{i=1}^n X_i\right) = \sum_{s \in \Omega} P(s) \sum_{i=1}^n X_i(s) = \sum_{i=1}^n \sum_{s \in \Omega} P(s) X_i(s) = \sum_{i=1}^n E(X_i).$$

$$\begin{aligned} E(aX + b) &= \sum_{s \in \Omega} P(s)(aX(s) + b) = (a \sum_{s \in \Omega} P(s)X(s)) + b \sum_{s \in \Omega} P(s) \\ &= aE(X) + b. \quad \square \end{aligned}$$

Using linearity of expectation

Theorem: The expected # of successes in n (not necessarily independent) Bernoulli trials, with probability p of success in each trial, is np .

Easy proof, via linearity of expectation: For $\Omega = \{H, T\}^n$, let X be the r.v. counting the expected number of successes, and for each i , let $X_i : \Omega \rightarrow \mathbb{R}$ be the binary r.v. defined by:

$$X_i((s_1, \dots, s_n)) = \begin{cases} 1 & \text{if } s_i = H \\ 0 & \text{if } s_i = T \end{cases}$$

Note that $E(X_i) = p \cdot 1 + (1 - p) \cdot 0 = p$, for all $i \in \{1, \dots, n\}$.

Also, clearly, $X = X_1 + X_2 + \dots + X_n$, so:

$$E(X) = E(X_1 + \dots + X_n) = \sum_{i=1}^n E(X_i) = np. \quad \square$$

Note: this holds even if the n coin tosses are totally correlated.



Using linearity of expectation, continued

Hatcheck problem: At a restaurant, the hat-check person forgets to put claim numbers on hats.

n customers check their hats in, and they each get a **random** hat back when they leave the restuarant.

What is the expected number, $E(X)$, of people who get their correct hat back?

Answer: Let X_i be the r.v. that is 1 if the i 'th customer gets their hat back, and 0 otherwise.

Clearly, $E(X) = E(\sum_i X_i)$.

Furthermore, $E(X_i) = P(i\text{'th person gets its hat back}) = 1/n$.

Thus, $E(X) = n \cdot (1/n) = 1$. □

This would be **much** harder to prove without using the linearity of expectation.

Note: $E(X)$ doesn't even depend on n in this case.

Independence of Random Variables

Definition: Two random variables, X and Y , are called **independent** if for all $r_1, r_2 \in \mathbb{R}$:

$$P(X = r_1 \text{ and } Y = r_2) = P(X = r_1) \cdot P(Y = r_2)$$

Example: Two die are rolled. Let X_1 be the number that comes up on die 1, and let X_2 be the number that comes up on die 2. Then X_1 and X_2 are independent r.v.'s.

Theorem: If X and Y are independent random variables on the same space Ω . Then

$$E(XY) = E(X)E(Y)$$

We will not prove this in class. (The proof is a simple re-arrangement of the sums in the definition of expectation. See Rosen's book for a proof.)

Variance

The “variance” and “standard deviation” of a r.v., X , give us ways to measure (roughly) “on average, how far off the value of the r.v. is from its expectation”.

Variance and Standard Deviation

Definition: For a random variable X on a sample space Ω , the **variance** of X , denoted by $V(X)$, is defined by:

$$V(X) = E((X - E(X))^2) = \sum_{s \in \Omega} (X(s) - E(X))^2 P(s)$$

The **standard deviation** of X , denoted $\sigma(X)$, is defined by

$$\sigma(X) = \sqrt{V(X)}$$

Example, and a useful identity for variance

Example: Consider the r.v., X , such that $P(X = 0) = 1$, and the r.v. Y , such that $P(Y = -10) = P(Y = 10) = 1/2$. Then $E(X) = E(Y) = 0$, but $V(X) = 0 = \sigma(X)$, whereas $V(Y) = 100$ and $\sigma(Y) = 10$. □

Theorem: For any random variable X ,

$$V(X) = E(X^2) - E(X)^2$$

Proof:

$$\begin{aligned} V(X) &= E((X - E(X))^2) \\ &= E(X^2 - 2XE(X) + E(X)^2) \\ &= E(X^2) - 2E(X)E(X) + E(X)^2 \\ &= E(X^2) - E(X)^2. \quad \square \end{aligned}$$

Discrete Mathematics & Mathematical Reasoning

Chapter 7 (continued):

Markov and Chebyshev's Inequalities; and
Examples in probability:
the birthday problem

Kousha Etessami

U. of Edinburgh, UK

Markov's Inequality

Often, for a random variable X that we are interested in, we want to know

“What is the probability that the value of the r.v., X , is ‘far’ from its expectation?”

A generic answer to this, which holds for any non-negative random variable, is given by **Markov's inequality**:

Markov's Inequality

Theorem: For a nonnegative random variable, $X : \Omega \rightarrow \mathbb{R}$, where $X(s) \geq 0$ for all $s \in \Omega$, for any positive real number $a > 0$:

$$P(X \geq a) \leq \frac{E(X)}{a}$$

Proof of Markov's Inequality:

Let the event $A \subseteq \Omega$ be defined by: $A = \{s \in \Omega \mid X(s) \geq a\}$.

We want to prove that $P(A) \leq \frac{E(X)}{a}$. But:

$$\begin{aligned} E(X) &= \sum_{s \in \Omega} P(s)X(s) \\ &= \sum_{s \in A} P(s)X(s) + \sum_{s \notin A} P(s)X(s) \\ &\geq \sum_{s \in A} P(s)X(s) \quad (\text{because } X(s) \geq 0 \text{ for all } s \in \Omega) \\ &\geq \sum_{s \in A} P(s)a \quad (\text{because } X(s) \geq a \text{ for all } s \in A) \\ &= a \sum_{s \in A} P(s) = a \cdot P(A) \end{aligned}$$

Thus, $E(X) \geq a \cdot P(A)$. In other words, $\frac{E(X)}{a} \geq P(A)$, which is what we wanted to prove.

Example

Question: A biased coin, which lands heads with probability $1/10$ each time it is flipped, is flipped 200 times consecutively. Give an upper bound on the probability that it lands heads at least 120 times.

Example

Question: A biased coin, which lands heads with probability $1/10$ each time it is flipped, is flipped 200 times consecutively. Give an upper bound on the probability that it lands heads at least 120 times.

Answer: The number of heads is a binomially distributed r.v., X , with parameters $p = 1/10$ and $n = 200$.

Thus, the expected number of heads is

$$E(X) = np = 200 \cdot (1/10) = 20.$$

By [Markov Inequality](#), the probability of at least 120 heads is

$$P(X \geq 120) \leq \frac{E(X)}{120} = \frac{20}{120} = 1/6. \quad \square$$

Later we will see that one can give **MUCH MUCH BETTER** bounds in this specific case.

Chebyshev's Inequality

Another answer to the question of “what is the probability that the value of X is far from its expectation” is given by **Chebyshev’s Inequality**, which works for **any** random variable (not necessarily a non-negative one).

Chebyshev’s Inequality

Theorem: Let $X : \Omega \rightarrow \mathbb{R}$ be any random variable, and let $r > 0$ be any positive real number. Then:

$$P(|X - E(X)| \geq r) \leq \frac{V(X)}{r^2}$$

First proof of Chebyshev's Inequality:

Let $A \subseteq \Omega$ be defined by: $A = \{s \in \Omega \mid |X(s) - E(X)| \geq r\}$.

We want to prove that $P(A) \leq \frac{V(X)}{r^2}$. But:

$$\begin{aligned} V(X) &= \sum_{s \in \Omega} P(s)(X(s) - E(X))^2 \\ &= \sum_{s \in A} P(s)(X(s) - E(X))^2 + \sum_{s \notin A} P(s)(X(s) - E(X))^2 \\ &\geq \sum_{s \in A} P(s)(X(s) - E(X))^2 \text{ (since } \forall s, (X(s) - E(X))^2 \geq 0) \\ &\geq \sum_{s \in A} P(s)r^2 \text{ (because } |X(s) - E(X)| \geq r \text{ for all } s \in A) \\ &= r^2 \sum_{s \in A} P(s) = r^2 \cdot P(A) \end{aligned}$$

Thus, $V(X) \geq r^2 \cdot P(A)$. In other words, $\frac{V(X)}{r^2} \geq P(A)$, which is what we wanted to prove.

Our first proof of Chebyshev's inequality looked suspiciously like our proof of Markov's Inequality. That is no co-incidence. Chebyshev's inequality can be derived as a special case of Markov's inequality.

Second proof of Chebyshev's Inequality:

Note that

$$A = \{s \in \Omega \mid |X(s) - E(X)| \geq r\} = \{s \in \Omega \mid (X(s) - E(X))^2 \geq r^2\}.$$

Now, consider the random variable, Y , where

$$Y(s) = (X(s) - E(X))^2.$$

Note that Y is a non-negative random variable.

Thus, we can apply Markov's inequality to it, to get:

$$P(A) = P(Y \geq r^2) \leq \frac{E(Y)}{r^2} = \frac{E((X - E(X))^2)}{r^2} = \frac{V(X)}{r^2}. \quad \square$$

Brief look at a more advanced topic: Chernoff bounds

For specific random variables, particularly those that arise as sums of many independent random variables, we can get **much better** bounds on the probability of deviation from expectation.

One very special case of **Chernoff Bounds**

Theorem: Suppose we conduct a sequence of n mutually independent Bernoulli trials, with probability p of “success” (heads) in each trial. Let $X : \Omega \rightarrow \mathbb{N}$ be the binomially distributed r.v. that counts the total number of successes (recall that $E(X) = np$). Then, for all $C \geq 6E(X)$:

$$P(X \geq C) \leq 2^{-C}$$

We will not prove this theorem, and we will not assume you know it (it is not in the book).

An application of Chernoff bounds

Question: A biased coin is flipped 200 times consecutively, and comes up heads with probability $1/10$ each time it is flipped. Give an upper bound the probability that it will come up heads at least 120 times.

An application of Chernoff bounds

Question: A biased coin is flipped 200 times consecutively, and comes up heads with probability $1/10$ each time it is flipped. Give an upper bound the probability that it will come up heads at least 120 times.

Solution: Let X be the r.v. that counts the number of heads. Recall: $E(X) = 200 * (1/10) = 20$. By Chernoff bounds,

$$P(X \geq 120) = P(X \geq 6E(X)) \leq 2^{-6E(X)} = 2^{-(6 \cdot 20)} = 2^{-120}. \quad \square$$

Note: By using Markov's inequality, we were only able to determine that $P(X \geq 120) \leq (1/6)$.

But by using Chernoff bounds, which are specifically geared for large deviation bounds for binomial and related distributions, we get that $P(X \geq 120) \leq 2^{-120}$.

That is a vastly better upper bound!

The Birthday Problem

There are many illuminating and surprising examples in probability theory. We will see some of them in the next couple of lectures, in order to build our intuition about probability.

One well-known example is called the [Birthday problem](#).

Birthday problem

There are 30 people in a room. I am willing to bet you that “[at least two people in the room have the same birthday](#)”.

Should you take my bet? (I offer even odds.)

The Birthday Problem

There are many illuminating and surprising examples in probability theory. We will see some of them in the next couple of lectures, in order to build our intuition about probability.

One well-known example is called the [Birthday problem](#).

Birthday problem

There are 30 people in a room. I am willing to bet you that “[at least two people in the room have the same birthday](#)”.

Should you take my bet? (I offer even odds.)

In other words, you have to calculate:

is there at least $1/2$ probability that no two people will have the same birthday in a room with 30 people?

([We are implicitly assuming that these people's birthdays are independent and uniformly distributed throughout the 365\(+1\) days of the year, taking into account leap years.](#))

Toward a solution to the Birthday problem:

Question: What is the probability, p_m , that m people in a room all have different birthdays?

Toward a solution to the Birthday problem:

Question: What is the probability, p_m , that m people in a room all have different birthdays?

We can equate the birthdays of m people to a list (b_1, \dots, b_m) , with each $b_i \in \{1, \dots, 366\}$.

We are assuming each list in $B = \{1, \dots, 366\}^m$ is equally likely.

Note that $|B| = 366^m$. What is the size of

$$A = \{(b_1, \dots, b_m) \in B \mid b_i \neq b_j \text{ for all } i \neq j, i, j \in \{1, \dots, m\}\} ?$$

This is simply the # of ***m-permutations*** from a set of size 366.

Thus $|A| = 366 \cdot (366 - 1) \dots (366 - (m - 1))$.

$$\text{Thus, } p_m = \frac{|A|}{|B|} = \prod_{i=1}^m \frac{366-i+1}{366} = \prod_{i=1}^m \left(1 - \frac{i-1}{366}\right).$$

By brute-force calculation, $p_{30} = 0.2947$. Thus, the probability that at least two people **do** have the same birthday in a room with 30 people is $1 - p_{30} = 0.7053$.

So, you shouldn't have taken my bet! Not even for 23 people in a room, because $1 - p_{23} = 0.5063$. But $1 - p_{22} = 0.4745$.



A general result underlying the birthday paradox

Theorem: Suppose that each of $m \geq 1$ pigeons independently and uniformly at random enter one of $n \geq 1$ pigeon-holes. If

$$m \geq (1.1775 \cdot \sqrt{n}) + 1$$

then the probability that two pigeons go into the same pigeon-hole is greater than $1/2$.

Proof: Basic Fact: $1 + x \leq e^x$, for all real numbers x .

The probability that m random pigeons all go in different pigeonholes, when there are n pigeonholes, is:

$$\prod_{i=1}^{m-1} \left(1 + \left(-\frac{i}{n}\right)\right) \leq \prod_{i=1}^{m-1} e^{-i/n} = e^{-\frac{1}{n} \sum_{i=1}^{m-1} i} = e^{-\frac{m(m-1)}{2n}}$$

So we want m to be big enough so that $e^{-\frac{m(m-1)}{2n}} < 1/2$.
Taking logs, and negating, this is equivalent to

$$\frac{m(m-1)}{2n} > \ln 2 \iff m(m-1) > (2 \cdot \ln 2) \cdot n$$

Thus, since $m(m-1) > (m-1)^2$, it suffices if

$$(m-1)^2 \geq (2 \cdot \ln 2) \cdot n \iff (m-1) \geq \sqrt{(2 \cdot \ln 2)} \cdot \sqrt{n}$$

Thus, since $\sqrt{(2 \ln 2)} = 1.177410\dots \leq 1.1775$, it suffices if:

$$m \geq (1.1775 \cdot \sqrt{n}) + 1. \quad \square$$

We will not assume you know this proof.

Discrete Mathematics & Mathematical Reasoning

Chapter 7 (continued): Examples in probability: Ramsey numbers

and the probabilistic method

Kousha Etessami

U. of Edinburgh, UK



Frank Ramsey (1903-1930)

A brilliant logician/mathematician.

He studied and lectured at Cambridge University.

He died tragically young, at age 26.

Despite his early death,
he did hugely influential work in several fields:
logic, combinatorics, and economics.

Friends and Enemies

Theorem: Suppose that in a group of 6 people every pair are either friends or enemies.

Then, there are either 3 mutual friends or 3 mutual enemies.

Proof: Let $\{A, B, C, D, E, F\}$ be the 6 people.

Consider A 's friends & enemies. A has 5 relationships, so A must either have 3 friends or 3 enemies.

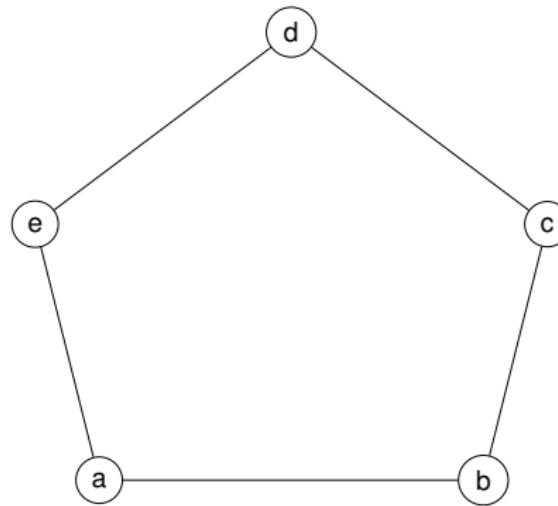
Suppose, for example, that $\{B, C, D\}$ are all friends of A .

If some pair in $\{B, C, D\}$ are friends, for example $\{B, C\}$, then $\{A, B, C\}$ are 3 mutual friends. Otherwise, $\{B, C, D\}$ are 3 mutual enemies.

The same argument clearly works if A had 3 enemies instead of 3 friends.

Remarks on “Friends and Enemies”: 6 is the smallest number possible for finding 3 friends or 3 enemies

Note that it is possible to have 5 people, where every pair of them are either friends or enemies, such that there does not exist 3 of them who are all mutual friends or all mutual enemies:



Graphs and Ramsey's Theorem

Ramsey's Theorem (a special case, for graphs)

Theorem: For any positive integer, k , there is a positive integer, n , such that in any undirected graph with n or more vertices:
either there are k vertices that are all mutually adjacent,
meaning they form a k -clique,
or, there are k vertices that are all mutually non-adjacent,
meaning they form a k -independent-set.

For each integer $k \geq 1$, let $R(k)$ be the **smallest** integer $n \geq 1$ such that every undirected graph with n or more vertices has either a k -clique or a k -independent-set as an induced subgraph.

The numbers $R(k)$ are called **diagonal Ramsey numbers**.

Proof of Ramsey's Theorem: Consider any integer $k \geq 1$, and any graph, $G_1 = (V_1, E_1)$ with at least 2^{2k} vertices.

Initialize: $S_{Friends} := \{\}$; $S_{Enemies} := \{\}$;

for $i := 1$ to $2k - 1$ **do**

Pick any vertex $v_i \in V_i$;

if (v_i has at least 2^{2k-i} friends in G_i) **then**

$S_{Friends} := S_{Friends} \cup \{v_i\}$; $V_{i+1} := \{\text{friends of } v_i\}$;

else (* in this case v_i has at least 2^{2k-i} enemies in G_i *)

$S_{Enemies} := S_{Enemies} \cup \{v_i\}$; $V_{i+1} := \{\text{enemies of } v_i\}$;

end if

Let $G_{i+1} = (V_{i+1}, E_{i+1})$ be the subgraph of G_i induced by V_{i+1} ;

end for

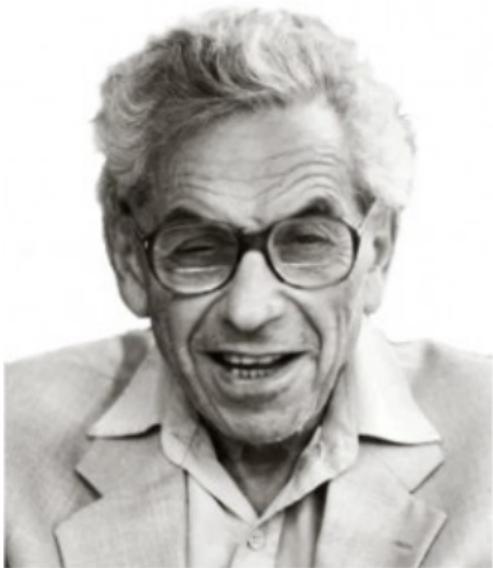
At the end, all vertices in $S_{Friends}$ are mutual friends, and all vertices in $S_{Enemies}$ are mutual enemies. Since

$|S_{Friends} \cup S_{Enemies}| = 2k - 1$, either $|S_{Friends}| \geq k$ or $|S_{Enemies}| \geq k$.

Done.

Remarks on the proof, and on Ramsey numbers

- The proof establishes that $R(k) \leq 2^{2k} = 4^k$.
(A more careful look at this proof shows that $R(k) \leq 2^{2k-1}$.)
- **Question:** Can we give a better upper bound on $R(k)$?
- **Question:** Can we give a good **lower bound** on $R(k)$?



Paul Erdős (1913-1996)

Immensely prolific mathematician,
eccentric nomad,
father of the probabilistic method in combinatorics.

Lower bounds on Ramsey numbers, and the Probabilistic Method

Theorem (Erdős, 1947)

For all $k \geq 3$,

$$R(k) > 2^{k/2}$$

The proof uses the probabilistic method:

General idea of “the probabilistic method”: To show the **existence** of a hard-to-find object with a desired property, Q , try to construct a probability distribution over a sample space Ω of objects, and show that **with positive probability** a randomly chosen object in Ω has the property Q .

Proof that $R(k) > 2^{k/2}$ using the probabilistic method:

Let Ω be the set of all graphs on the vertex set $V = \{v_1, \dots, v_n\}$.
(We will later determine that $n \leq 2^{k/2}$ suffices.)

There are $2^{\binom{n}{2}}$ such graphs. Let $P : \Omega \rightarrow [0, 1]$, be the uniform probability distribution on such graphs.

So, every graph on V is equally likely. This implies that:

$$\text{For all } i \neq j \quad P(\{v_i, v_j\} \text{ is an edge of the graph}) = 1/2. \quad (1)$$

We could also define the distribution P by saying it satisfies (1).

There are $\binom{n}{k}$ subsets of V of size k .

Let $S_1, S_2, \dots, S_{\binom{n}{k}}$ be an enumeration of these subsets of V .

For $i = 1, \dots, \binom{n}{k}$, let E_i be the event that S_i forms either a k -clique or a k -independent-set in the graph. Note that:

$$P(E_i) = 2 \cdot 2^{-\binom{k}{2}} = 2^{-\binom{k}{2}+1}$$

Proof of $R(k) > 2^{k/2}$ (continued):

Note that $E = \bigcup_{i=1}^{\binom{n}{k}} E_i$ is the event that there **exists** either a k -clique or a k -independent-set in the graph. But:

$$P(E) = P\left(\bigcup_{i=1}^{\binom{n}{k}} E_i\right) \leq \sum_{i=1}^{\binom{n}{k}} P(E_i) = \binom{n}{k} \cdot 2^{-\binom{k}{2}+1}$$

Question: How small must n be so that $\binom{n}{k} \cdot 2^{-\binom{k}{2}+1} < 1$?

For $k \geq 2$: $\binom{n}{k} = \frac{n(n-1)\dots(n-k+1)}{k(k-1)\dots 1} < \frac{n^k}{2^{k-1}}$

Thus, if $n \leq 2^{k/2}$, then

$$\binom{n}{k} \cdot 2^{-\binom{k}{2}+1} < \frac{(2^{k/2})^k}{2^{k-1}} \cdot 2^{-\binom{k}{2}+1} = \frac{2^{k^2/2}}{2^{k-1}} \cdot 2^{-k(k-1)/2+1} = 2^{2-\frac{k}{2}}$$

Completion of the proof that $R(k) > 2^{k/2}$:

For $k \geq 4$, $2^{2-(k/2)} \leq 1$.

So, for $k \geq 4$, $P(E) < 1$, and thus $P(\Omega - E) = 1 - P(E) > 0$.

But note that $P(\Omega - E)$ is the probability that in a random graph of size $n \leq 2^{k/2}$, there is no k -clique and no k -independent-set.

Thus, since $P(\Omega - E) > 0$, such a graph **must exist** for any $n \leq 2^{k/2}$.

Note that we earlier argued that $R(3) = 6$, and clearly $6 > 2^{3/2} = 2.828 \dots$

Thus, we have established that for all $k \geq 3$,

$$R(k) > 2^{k/2}. \quad \square$$

A Remark

In the proof, we used the following trivial but often useful fact:

Union bound

Theorem: For any (finite or countable) sequence of events E_1, E_2, E_3, \dots

$$P\left(\bigcup_i E_i\right) \leq \sum_i P(E_i)$$

Proof (trivial):

$$P\left(\bigcup_i E_i\right) = \sum_{s \in \bigcup_i E_i} P(s) \leq \sum_i \sum_{s \in E_i} P(s) = \sum_i P(E_i). \quad \square$$

Remarks on Ramsey numbers

- We have shown that

$$2^{k/2} = (\sqrt{2})^k < R(k) \leq 4^k = 2^{2k}$$

¹See [Conlon,2009] for state-of-the-art upper bounds.

Remarks on Ramsey numbers

- We have shown that

$$2^{k/2} = (\sqrt{2})^k < R(k) \leq 4^k = 2^{2k}$$

- Despite decades of research by many combinatorists,
nothing significantly better is known!¹ In particular:

no constant $c > \sqrt{2}$ is known such that $c^k \leq R(k)$, and
no constant $c' < 4$ is known such that $R(k) \leq (c')^k$.

- For specific small k , more is known:

$$R(1) = 1 ; R(2) = 2 ; R(3) = 6 ; R(4) = 18$$

$$43 \leq R(5) \leq 49$$

$$102 \leq R(6) \leq 165$$

...

¹See [Conlon,2009] for state-of-the-art upper bounds.

Why can't we just compute $R(k)$ exactly, for small k ?

For each k , we know that $2^{k/2} < R(k) < 2^{2k}$,

So, we could try to check, exhaustively, for each r such that $2^{k/2} < r < 2^{2k}$, whether there is a graph G with r vertices such that G has no k -clique and no k -independent set.

Question: How many graphs on r vertices are there?

There are $2^{\binom{r}{2}} = 2^{r(r-1)/2}$ (labeled) graphs on r vertices.

So, for $r = 2^k$, we would have to check $2^{2^k(2^k-1)/2}$ graphs!!

So for $k = 5$, just for $r = 2^5$, we have to check 2^{496} graphs !!

Quote attributed to Paul Erdős:

Suppose an alien force, vastly more powerful than us, landed on Earth demanding to know the value of $R(5)$, or else they would destroy our planet.

Quote attributed to Paul Erdős:

Suppose an alien force, vastly more powerful than us, landed on Earth demanding to know the value of $R(5)$, or else they would destroy our planet.

In that case, I believe we should marshal all our computers, and all our mathematicians, in an attempt to find the value.

Quote attributed to Paul Erdős:

Suppose an alien force, vastly more powerful than us, landed on Earth demanding to know the value of $R(5)$, or else they would destroy our planet.

In that case, I believe we should marshal all our computers, and all our mathematicians, in an attempt to find the value.

But suppose instead they asked us for $R(6)$.

Quote attributed to Paul Erdős:

Suppose an alien force, vastly more powerful than us, landed on Earth demanding to know the value of $R(5)$, or else they would destroy our planet.

In that case, I believe we should marshal all our computers, and all our mathematicians, in an attempt to find the value.

But suppose instead they asked us for $R(6)$.

In that case, I believe we should attempt to destroy the aliens.