

Proof Techniques	<p><b>By Construction (With Example):</b> Prove that there exist irrational numbers p and q such <math>p^q</math> is rational.</p> <p>1. We know from Theorem 4.7.1 (Epp) that <math>\sqrt{2}</math> is irrational</p> <p>2. Consider <math>\sqrt{2}^{\sqrt{2}}</math> : it is either rational or unrational</p> <p>3. <b>Case 1:</b> It is rational</p> <p>3.1 Let <math>p = q = \sqrt{2}</math>, and we are done</p> <p>4. <b>Case 2:</b> It is irrational</p> <p>4.1 Let <math>p = \sqrt{2}^{\sqrt{2}}</math> and <math>q = \sqrt{2}</math>.</p> <p>4.2 p is irrational (by assumption) and so is q by Theorem 4.7.1 (Epp)</p> <p>4.3 Consider <math>p^q = (\sqrt{2}^{\sqrt{2}})^{\sqrt{2}}</math></p> <p><math>= \sqrt{2}^{\sqrt{2} \times \sqrt{2}}</math> (by the power law)</p> <p><math>= \sqrt{2}^2 = 2</math> (by algebra)</p> <p>4.4 Clearly, 2 is rational</p> <p>5. In either case, we have found the required p and q</p>	<p><b>By Contraposition (With Example):</b> The contrapositive of “if P then Q” is “if <math>\sim Q</math> then <math>\sim P</math>”. Both statements are logically equivalent.</p> <p>Prove that if <math>x^2</math> is irrational then x is irrational.</p> <p>1. <b>Contrapositive:</b> if x is rational then <math>x^2</math> is rational</p> <p>2. Since x is rational, there exists a, b <math>\in \mathbb{Z}</math> such that <math>b \neq 0</math> and <math>x = \frac{a}{b}</math></p> <p>3. So <math>x^2 = \frac{a \times a}{b \times b}</math> by basic algebra</p> <p>4. The numerator and denominator are integers, by the closure property of integers.</p> <p>5. Also, by rule T21 of Appendix A (Epp), <math>b^2 \neq 0</math></p> <p>6. Therefore, <math>x^2</math> is a ratio of two integers</p> <p>7. And by definition of rationals, <math>x^2</math> is rational.</p> <p>8. Since the contrapositive of the original statement is true, then the original statement is true.</p>	<p><b>By Contradiction (With Example):</b> Prove that 7 is not colourful (an integer n is said to be colourful if there exists some integer k such that <math>n = 3k</math>).</p> <p>1. Suppose 7 is colourful.</p> <p>2. Then, by definition of colourful, <math>7 = 3k</math> for <math>k \in \mathbb{Z}</math></p> <p>3. <math>1 = 3k - 6</math> (by basic algebra)</p> <p>4. <math>1 = 3(k - 2)</math> (by basic algebra)</p> <p>5. Since <math>(k - 2)</math> is an integer by the closure property, then <math>3 \mid 1</math> by definition of divisibility.</p> <p>6. Then by Theorem 4.3.1 (Epp), this means <math>3 \leq 1</math>.</p> <p>7. Clearly, this is absurd, so the statement is true (by contradiction rule).</p>						
	Logic of Compound Statements	<p><b>Definition 2.1.1 – Statement</b> A statement or proposition is a sentence that is true or false, but not both.</p> <p><b>Definition 2.1.2 – Negation</b> The negation of p is “not p” and is denoted <math>\sim p</math>.</p> <p><b>Definition 2.1.3 – Conjunction</b> The conjunction of p and q is “p and q”, denoted <math>p \wedge q</math>.</p> <p><b>Definition 2.1.4 – Disjunction</b> The disjunction of p and q is “p or q”, denoted as <math>p \vee q</math>.</p> <p><b>Definition 2.1.5 – Statement/Propositional Form</b> A statement/propositional form is an expression made up of statement variables and logical connectives that becomes a statement when actual statements are substituted for the component statement variables.</p> <p><b>Definition 2.1.6 – Logical Equivalence</b> Two statements are logically equivalent only if they have identical truth values for each possible situation of statements for their statement variables. The logical equivalence of p and q is denoted by <math>p \equiv q</math>.</p> <p><b>Definition 2.1.7 – Tautology</b> A tautology is a statement form that is always true regardless of the truth values of the individual statements substituted for its statement variables.</p> <p><b>Definition 2.1.8 – Contradiction</b> A contradiction is a statement form that is always false regardless of the truth values of the individual statements substituted for its statement variables.</p>	<p><b>Definition 2.2.1 – Conditional</b> The conditional of q by p is “if p then q”, denoted <math>p \rightarrow q</math>, where p is the hypothesis/antecedent and q is the conclusion/consequent.</p> <p><b>Definition 2.2.2 – Contrapositive</b> The contrapositive of conditional statement “if p then q” is “if <math>\sim q</math> then <math>\sim p</math>”, denoted <math>\sim q \rightarrow \sim p</math>.</p> <p><b>Definition 2.2.3 – Converse</b> (i.e. converse error) The converse of conditional statement “if p then q” is “if q then p”, denoted <math>q \rightarrow p</math>.</p> <p><b>Definition 2.2.4 – Inverse</b> (i.e. inverse error) The inverse of conditional statement “if p then q” is “if <math>\sim p</math> then <math>\sim q</math>”, denoted <math>\sim p \rightarrow \sim q</math>.</p> <p><b>Definition 2.2.5 – Only If</b> If “p only if q”, it means “if not q then not p” or “if p then q”.</p> <p><b>Definition 2.2.6 – Biconditional</b> The biconditional of p and q is “p if, and only if q”, denoted <math>p \leftrightarrow q</math>.</p> <p><b>Definition 2.2.7 – Necessary and Sufficient Conditions</b> The statement “p is a sufficient condition for q” means “if p then q”. The statement “p is a necessary condition for q” means “if <math>\sim p</math> then <math>\sim q</math>” or “if q then p”.</p>	<p><b>Definition 2.3.1 – Argument</b> An argument is a sequence of statements. All statements in an argument, except the final one, are premises/assumptions/hypotheses. The final statement is the conclusion.</p> <p>An argument is valid if, and only if, whenever statements are substituted that make all the premises true, the conclusion is also true.</p> <p><b>Definition 2.3.2 – Sound and Unsound Arguments</b> An argument is sound if, and only if, it is valid and all its premises are true. Otherwise, it is unsound.</p> <p><b>Order of Operations</b></p> <table><tr><td>Performed first</td><td><math>\sim</math> (not)</td></tr><tr><td></td><td><math>\wedge</math> (and), <math>\vee</math> (or) {coequal in order}</td></tr><tr><td>Performed last</td><td><math>\rightarrow</math> (if-then), <math>\leftrightarrow</math> (if, and only if) {coequal in order}</td></tr></table>	Performed first	$\sim$ (not)		$\wedge$ (and), $\vee$ (or) {coequal in order}	Performed last
Performed first	$\sim$ (not)								
	$\wedge$ (and), $\vee$ (or) {coequal in order}								
Performed last	$\rightarrow$ (if-then), $\leftrightarrow$ (if, and only if) {coequal in order}								

Logic of Compound Statements	Theorem 2.1.1 – Logical Equivalences			
	1	Commutative Laws	$p \wedge q \equiv q \wedge p$	$p \vee q \equiv q \vee p$
	2	Associative Laws	$(p \wedge q) \wedge r \equiv p \wedge (q \wedge r)$	$(p \vee q) \vee r \equiv p \vee (q \vee r)$
	3	Distributive Laws	$p \wedge (q \vee r) \equiv (p \wedge q) \vee (p \wedge r)$	$p \vee (q \wedge r) \equiv (p \vee q) \wedge (p \vee r)$
	4	Identity Laws	$p \wedge \text{true} \equiv p$	$p \vee \text{false} \equiv p$
	5	Negation Laws	$p \wedge \sim p \equiv \text{false}$	$p \vee \sim p \equiv \text{true}$
	6	Double Negative Law	$\sim(\sim p) \equiv p$	
	7	Idempotent Laws	$p \wedge p \equiv p$	$p \vee p \equiv p$
	8	Universal Bound Laws	$p \wedge \text{false} \equiv \text{false}$	$p \vee \text{true} \equiv \text{true}$
	9	De Morgan’s Laws	$\sim(p \wedge q) \equiv \sim p \vee \sim q$	$\sim(p \vee q) \equiv \sim p \wedge \sim q$
	10	Absorption Laws	$p \wedge (p \vee q) \equiv p$	$p \vee (p \wedge q) \equiv p$
	11	Negation of True and False	$\sim \text{true} \equiv \text{false}$	$\sim \text{false} \equiv \text{true}$
	12	Implication Law	$p \rightarrow q \equiv \sim p \vee q$	
	13	Contrapositive	$p \rightarrow q \equiv \sim q \rightarrow \sim p$	
	14	Converse and Inverse	$q \rightarrow p \equiv \sim p \rightarrow \sim q$	

Table 2.3.1 – Rules of Inference			
1	Modus Ponens	$p \rightarrow q$ $p$ $\bullet q$	
2	Modus Tollens	$p \rightarrow q$ $\sim q$ $\bullet \sim p$	
3	Generalisation	$p$ $\Rightarrow p \vee q$	$q$ $\Rightarrow p \vee q$
4	Specialisation	$p \wedge q$ $\Rightarrow p$	$p \wedge q$ $\Rightarrow q$
5	Conjunction	$p$ $q$ $\bullet p \wedge q$	
6	Elimination	$p \vee q$ $\sim q$ $\bullet p$	$p \vee q$ $\sim p$ $\bullet q$
7	Transitivity	$p \rightarrow q$ $q \rightarrow r$ $\bullet p \rightarrow r$	
8	Proof by Division into Cases	$p \vee q$ $p \rightarrow r$ $q \rightarrow r$ $\bullet r$	
9	Contradiction Rule	$\sim p \rightarrow \text{false}$ $\bullet p$	

Logic of Quantified Statements	<p><b>Definition 3.1.1 – Predicate</b> A predicate is a sentence that contains a finite number of variables, and it becomes a statement when specific values are substituted for the variables. The domain of a predicate variable is the set of all values that may be substituted in place of the variable.</p> <p><b>Definition 3.1.2 – Truth Set</b> If <math>P(x)</math> is a predicate and <math>x</math> has the domain <math>D</math>, the truth set is the set of all elements of <math>D</math> that make <math>P(x)</math> true when they are substituted for <math>x</math>. The truth set of <math>P(x)</math> is denoted <math>\{x \in D \mid P(x)\}</math>.</p> <p><b>Definition 3.1.3 – Universal Statement</b> A universal statement is of the form "<math>\forall x \in D, Q(x)</math>". It is true if and only if <math>Q(x)</math> is true for every <math>x</math> in <math>D</math>. It is false if and only if <math>Q(x)</math> is false for at least one <math>x</math> in <math>D</math>. Such a value for <math>x</math> for which <math>Q(x)</math> is false is a counterexample.</p>	<p><b>Definition 3.1.4 – Existential Statement</b> An existential statement is of the form "<math>\exists x \in D</math>, such that <math>Q(x)</math>". It is true if and only if <math>Q(x)</math> is true for at least one <math>x</math> in <math>D</math>. It is false if and only if <math>Q(x)</math> is false for all <math>x</math> in <math>D</math>.</p> <p><b>Notation</b> <b><math>P(x) \Rightarrow Q(x)</math>:</b> Every element in truth set of <math>P(x)</math> is in the truth set of <math>Q(x)</math>, equivalently, "<math>\forall x, P(x) \rightarrow Q(x)</math>".</p> <p><b><math>P(x) \Leftrightarrow Q(x)</math>:</b> <math>P(x)</math> and <math>Q(x)</math> have identical truth sets, equivalently, "<math>\forall x, P(x) \leftrightarrow Q(x)</math>".</p>
--------------------------------	---	---

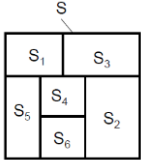
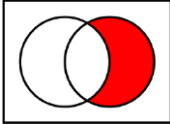
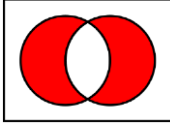
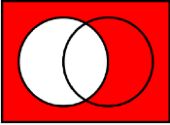
Logic of Quantified Statements	<p><b>Definition 3.2.1 – Contrapositive, Converse, Inverse</b></p> <p>For statement “<math>\forall x \in D</math>, if <math>P(x)</math> then <math>Q(x)</math>”:</p> <p><b>1. Contrapositive:</b> “<math>\forall x \in D</math>, if <math>\sim Q(x)</math> then <math>\sim P(x)</math>”.</p> <p><b>2. Converse:</b> “<math>\forall x \in D</math>, if <math>Q(x)</math> then <math>P(x)</math>”.</p> <p><b>3. Inverse:</b> “<math>\forall x \in D</math>, if <math>\sim P(x)</math> then <math>\sim Q(x)</math>”.</p>	<p><b>Theorem 3.2.1 (Epp) – Negation of a Universal Statement</b></p> <p>The negation of statement “<math>\forall x \in D</math>, <math>P(x)</math>” is logically equivalent to statement “<math>\exists x \in D</math>, such that <math>\sim P(x)</math>”.</p> $\sim(\forall x \in D, P(x)) \equiv \exists x \in D, \text{ such that } \sim P(x)$ <p><b>Theorem 3.2.2 (Epp) – Negation of an Existential Statement</b></p> <p>The negation of statement “<math>\exists x \in D</math>, such that <math>P(x)</math>” is logically equivalent to statement “<math>\forall x \in D</math>, <math>\sim P(x)</math>”.</p> $\sim(\exists x \in D, \text{ such that } P(x)) \equiv \forall x \in D, \sim P(x)$ <p><b>Universal Instantiation</b></p> <p>If some property is true of everything in the set, then it is true for any particular thing in the set.</p> <p><b>Existential Instantiation</b></p> <p>If the existence of a certain kind of object is assumed or has been deduced then it can be given a name, as long as that name is not currently being used to denote something else.</p>	<p><b>Universal Modus Ponens</b></p> <p><math>\forall x</math>, if <math>P(x)</math> then <math>Q(x)</math> <math>P(a)</math> for a particular <math>a</math>. • <math>Q(a)</math></p> <p><b>Universal Modus Tollens</b></p> <p><math>\forall x</math>, if <math>P(x)</math> then <math>Q(x)</math> <math>\sim Q(a)</math> for a particular <math>a</math>. • <math>\sim P(a)</math></p> <p><b>Converse Error (Quantified Form)</b></p> <p><math>\forall x</math>, if <math>P(x)</math> then <math>Q(x)</math> <math>Q(a)</math> for a particular <math>a</math>. • <math>P(a)</math></p> <p><b>Inverse Error (Quantified Form)</b></p> <p><math>\forall x</math>, if <math>P(x)</math> then <math>Q(x)</math> <math>\sim P(a)</math> for a particular <math>a</math>. • <math>\sim Q(a)</math></p>
	<p><b>Regular Induction</b></p> <p>1. Base Step: <math>P(0)</math> (or <math>P(a)</math> where <math>k \in \mathbb{Z}</math>, depending on the question)</p> <p>2. Inductive Step: <math>\forall k \in \mathbb{N}, P(k) \rightarrow P(k + 1)</math></p> <p>3. Conclusion: <math>\forall n \in \mathbb{N}, P(n)</math></p> <p><math>*\mathbb{N} = \{0, 1, 2, 3, 4, 5, \dots\}</math> in CS1231</p> <p><b>Example</b></p> <p>Prove that <math>\forall n \in \mathbb{N}</math>, <math>(4^n - 1)</math> is divisible by 3.</p> <ol style="list-style-type: none"><li>For <math>\forall n \in \mathbb{N}</math>, let <math>P(n) = (3 \mid (4^n - 1))</math></li><li><u>Base Case:</u> let <math>n = 0</math><ol style="list-style-type: none"><li>Clearly, <math>(4^0 - 1) = 0 = 3 \times 0</math></li><li>Thus, <math>P(0)</math> is true.</li></ol></li><li><u>Inductive Step:</u> for any <math>k \in \mathbb{N}</math>:<ol style="list-style-type: none"><li>Assume <math>P(k)</math> is true, i.e. <math>3 \mid (4^k - 1)</math></li><li>Consider the <math>k + 1</math> case:</li><li><math>(4^{k+1} - 1) = (4 \times 4^k - 1) = 4(4^k - 1) + 3</math>, by basic algebra</li><li>By inductive hypothesis, <math>3 \mid (4^k - 1)</math></li><li>Clearly, <math>3 \mid 3</math></li><li>By Theorem 4.1.1, <math>3 \mid (4(4^k - 1) + 3)</math></li><li><math>P(k)</math> is true <math>\rightarrow P(k + 1)</math> is true</li></ol></li><li>So by Mathematical Induction, the statement is true.</li></ol> <p><b>Note:</b></p> <p>- “For <math>\forall n \in \mathbb{N}</math>” is needed to qualify the domain of <math>n</math>, and it is also</p> <p>- Do not define the predicate as “<math>P(n) = (4^n - 1)</math>”! This is not a statement and it cannot be evaluated to be true or false.</p>	<p><b>Strong Induction</b></p> <p>1. Base Step: <math>P(0)</math> (or <math>P(a)</math> where <math>k \in \mathbb{Z}</math>, depending on the question)</p> <p>2. Inductive Step: <math>\forall k \in \mathbb{N}, P(k), P(k - 1), P(k - 2), \dots, P(a) \rightarrow P(k + 1)</math></p> <p>3. Conclusion: <math>\forall n \in \mathbb{N}, P(n)</math></p> <p><b>Example</b></p> <p>Prove that <math>\forall</math> integers <math>n &gt; 1</math>, <math>n</math> has a prime factorisation.</p> <ol style="list-style-type: none"><li><math>\forall</math> integers <math>n &gt; 1</math>, let <math>P(n) = (n \text{ has a prime factorisation})</math>.</li><li><u>Base Case:</u> let <math>n = 2</math><ol style="list-style-type: none"><li>Since 2 is a prime, <math>2 = 2</math> is a trivial prime factorisation.</li><li>Thus, <math>P(2)</math> is true.</li></ol></li><li><u>Inductive Step:</u> for any integer <math>k &gt; 1</math>:<ol style="list-style-type: none"><li>Assume <math>P(i)</math> is true for <math>1 &lt; i \leq k</math> (stronger assumption)</li><li>That is, all integers <math>i</math> within the range <math>1 &lt; i \leq k</math> have prime factorisations.</li><li>Consider the integer <math>k + 1</math>:</li><li>If <math>k + 1</math> is prime, then <math>k + 1 = k + 1</math> is a trivial prime factorisation, and <math>P(k+1)</math> is true.</li><li>If <math>k + 1</math> is composite, then <math>k + 1 = rs</math> for some integers <math>r</math> and <math>s</math>, such that <math>1 &lt; r, s &lt; k + 1</math>, by the definition of composite.</li><li>By the inductive hypothesis, both <math>r</math> and <math>s</math> have prime factorisations.</li><li>Let <math>r = p_1 p_2 \dots p_u</math> and <math>s = q_1 q_2 \dots q_v</math>, where all the factors are prime.</li><li>Then <math>k + 1 = rs = p_1 p_2 \dots p_u q_1 q_2 \dots q_v</math>, by basic algebra.</li><li>Thus <math>k + 1</math> has a prime factorisation and <math>P(k + 1)</math> is true.</li></ol></li><li>So, by Strong Induction, the statement is true.</li></ol> <p><b>Note:</b></p> <p>- Regular induction cannot be applied here because that hypothesis says only <math>P(k)</math> is true, but <math>r</math> and <math>s</math> may not equal to <math>k</math>.</p>	
Mathematical Induction			

Number Theory	<p><b>Definition 1.3.1 – Divisibility</b> If <math>n</math> and <math>d</math> are integers and <math>d \neq 0</math>, then <math>n</math> is divisible by <math>d</math> (i.e. <math>d \mid n</math>) if, and only if, for some integer <math>k</math>, <math>n = dk</math>.</p> <p><b>Theorem 4.1.1</b> For <math>\forall a, b, c \in \mathbb{Z}</math>, if <math>a \mid b</math> and <math>a \mid c</math>, then for <math>\forall x, y \in \mathbb{Z}</math>, <math>a \mid (bx + cy)</math>.</p> <p><b>Definition 4.2.1 – Prime and Composite Numbers</b> Every integer <math>n &gt; 1</math> is either prime or composite.</p> <ol style="list-style-type: none"><li>1. An integer <math>n</math> is a prime if, and only if, <math>n &gt; 1</math> and for all positive integers <math>r</math> and <math>s</math>, if <math>n = rs</math>, then either <math>r</math> or <math>s</math> equals <math>n</math>.</li><li>2. An integer <math>n</math> is composite if, and only if, <math>n &gt; 1</math> and <math>n = rs</math> for some integers <math>r</math> and <math>s</math> with <math>1 &lt; r &lt; n</math> and <math>1 &lt; s &lt; n</math>.</li></ol> <p><b>Proposition 4.2.2</b> For any two primes <math>p</math> and <math>p'</math>, if <math>p \mid p'</math> then <math>p = p'</math>.</p> <p><b>Proposition 4.7.3 (Epp)</b> For any <math>a \in \mathbb{Z}</math> and any prime <math>p</math>, if <math>p \mid a</math> then <math>p \nmid (a + 1)</math>.</p> <p><b>Theorem 4.7.4 (Epp) – Infinitude of Primes</b> The set of primes is infinite.</p> <p><b>Theorem 4.2.3</b> If <math>p</math> is a prime and <math>x_1, x_2, \dots, x_n</math> are any integers such that <math>p \mid x_1 x_2 \dots x_n</math>, then <math>p \mid x_i</math> for some <math>x_i</math> (<math>1 \leq i \leq n</math>).</p> <p><b>Theorem 4.3.5 (Epp) – Unique Prime Factorisation</b> Given any integer <math>n &gt; 1</math>, there exists a positive integer <math>k</math>, distinct prime numbers <math>p_1, p_2, \dots, p_k</math> and positive integers <math>e_1, e_2, \dots, e_k</math> such that <math>n = p_1^{e_1} p_2^{e_2} p_3^{e_3} \dots p_k^{e_k}</math>, and any other expression for <math>n</math> as a product of prime numbers is identical to this except, perhaps, for the order in which the factors are written.</p> <p><b>Theorem 4.3.1 – Lower Bound</b> An integer <math>b</math> is said to be a lower bound for a set <math>X \subseteq \mathbb{Z}</math> if <math>b \leq x</math> for all <math>x \in X</math>.</p> <p><b>Theorem 4.3.2 – Well Ordering Principle</b> If a non-empty set <math>S \subseteq \mathbb{Z}</math> has a lower bound, then <math>S</math> has a least element.</p> <p><b>Proposition 4.3.3 – Uniqueness of Least Element</b> If a set <math>S</math> of integers has a least element, then the least element is unique.</p>	<p><b>Theorem 4.3.2 – Well Ordering 2</b> If a non-empty set <math>S \subseteq \mathbb{Z}</math> has an upper bound, then <math>S</math> has a greatest element.</p> <p><b>Proposition 4.3.4 – Uniqueness of Greatest Element</b> If a set <math>S</math> of integers has a greatest element, then the greatest element is unique.</p> <p><b>Theorem 4.4.1 – Quotient-Remainder Theorem</b> Given any integer <math>a</math> and any positive integer <math>b</math>, there exist unique integers <math>q</math> (quotient) and <math>r</math> (remainder) such that: <math display="block">a = bq + r \text{ and } 0 \leq r &lt; b</math></p> <p><b>Representation of Integers (With Steps and Examples):</b></p> <table><tr><td>Express <math>(109)_{10}</math> in base 2: <math>109 = 2 \times 54 + 1</math> <math>54 = 2 \times 27 + 0</math> <math>27 = 2 \times 13 + 1</math> <math>13 = 2 \times 6 + 1</math> <math>6 = 2 \times 3 + 0</math> <math>3 = 2 \times 1 + 1</math> <math>1 = 2 \times 0 + 1</math> <math>(109)_{10} = (1101101)_2</math></td><td><ol style="list-style-type: none"><li>1. Use the quotient-remainder theorem to find the binary representation of 109.</li><li>2. Read the remainders from bottom up.</li><li>3. Read the binary representation from right to left: <math>(2^0 \cdot 1) + (2^1 \cdot 0) + (2^2 \cdot 1) + (2^3 \cdot 1) + (2^4 \cdot 0) + (2^5 \cdot 1) + (2^6 \cdot 1)</math></li></ol></td></tr><tr><td><math>(10110101)_2</math> to base 10: <math>(2^0 \cdot 1) + (2^1 \cdot 0) + (2^2 \cdot 1) + (2^3 \cdot 0) + (2^4 \cdot 1) + (2^5 \cdot 1) + (2^6 \cdot 0) + (2^7 \cdot 0)</math> <math>= (181)_{10}</math></td><td><math>(10110101)_2</math> to base 8: <math>181 = 8 \times 22 + 5</math> <math>22 = 8 \times 2 + 6</math> <math>2 = 8 \times 0 + 2</math> So <math>(10110101)_2 = (265)_8</math></td></tr></table> <p><b>Binary Notation – Base 2 to Base 16 (Hexadecimal)</b></p> <table><tr><td>0000 = 0</td><td>0001 = 1</td><td>0010 = 2</td><td>0011 = 3</td></tr><tr><td>0100 = 4</td><td>0101 = 5</td><td>0110 = 6</td><td>0111 = 7</td></tr><tr><td>1000 = 8</td><td>1001 = 9</td><td>1010 = A</td><td>1011 = B</td></tr><tr><td>1100 = C</td><td>1101 = D</td><td>1110 = E</td><td>1111 = F</td></tr></table> <p>i.e. <math>(109)_{10} = (0110 \ 1101)_2 = (6D)_{16}</math></p> <p><b>Definition 4.5.1 – Greatest Common Divisor</b> Let <math>a</math> and <math>b</math> be non-zero integers. The greatest common divisor of <math>a</math> and <math>b</math>, denoted <math>\gcd(a, b)</math>, is the integer <math>d</math> satisfying:</p> <ol style="list-style-type: none"><li>a. <math>d \mid a</math> and <math>d \mid b</math></li><li>b. <math>\forall c \in \mathbb{Z}</math>, if <math>c \mid a</math> and <math>c \mid b</math>, then <math>c \leq d</math></li></ol>	Express $(109)_{10}$ in base 2: $109 = 2 \times 54 + 1$ $54 = 2 \times 27 + 0$ $27 = 2 \times 13 + 1$ $13 = 2 \times 6 + 1$ $6 = 2 \times 3 + 0$ $3 = 2 \times 1 + 1$ $1 = 2 \times 0 + 1$ $(109)_{10} = (1101101)_2$	<ol style="list-style-type: none"><li>1. Use the quotient-remainder theorem to find the binary representation of 109.</li><li>2. Read the remainders from bottom up.</li><li>3. Read the binary representation from right to left: <math>(2^0 \cdot 1) + (2^1 \cdot 0) + (2^2 \cdot 1) + (2^3 \cdot 1) + (2^4 \cdot 0) + (2^5 \cdot 1) + (2^6 \cdot 1)</math></li></ol>	$(10110101)_2$ to base 10: $(2^0 \cdot 1) + (2^1 \cdot 0) + (2^2 \cdot 1) + (2^3 \cdot 0) + (2^4 \cdot 1) + (2^5 \cdot 1) + (2^6 \cdot 0) + (2^7 \cdot 0)$ $= (181)_{10}$	$(10110101)_2$ to base 8: $181 = 8 \times 22 + 5$ $22 = 8 \times 2 + 6$ $2 = 8 \times 0 + 2$ So $(10110101)_2 = (265)_8$	0000 = 0	0001 = 1	0010 = 2	0011 = 3	0100 = 4	0101 = 5	0110 = 6	0111 = 7	1000 = 8	1001 = 9	1010 = A	1011 = B	1100 = C	1101 = D	1110 = E	1111 = F	<p><b>Proposition 4.5.2 – Existence of GCD</b> For any non-zero integers <math>a</math> and <math>b</math>, their gcd exists and is unique.</p> <p><b>Euclid's Algorithm (With Steps and Examples):</b> Finding <math>\gcd(a, b)</math> is based on two facts:</p> <ol style="list-style-type: none"><li>a. <math>\gcd(a, 0) = a</math></li><li>b. <math>\gcd(a, b) = \gcd(b, r)</math>, where <math>r = a \% b</math></li></ol> <p>Calculate <math>\gcd(330, 156)</math>:</p> $330 = 156 \times 2 + 18 \leftarrow \gcd(156, 18)$ $156 = 18 \times 8 + 12 \leftarrow \gcd(18, 12)$ $18 = 12 \times 1 + 6 \leftarrow \gcd(12, 6)$ $12 = 6 \times 2 + 0 \leftarrow \gcd(6, 0)$ <p>Thus, the <math>\gcd(330, 156)</math> is 6.</p> <p><b>Theorem 4.5.3 – Bézout's Identity</b> Let <math>a</math> and <math>b</math> be non-zero integers, and let <math>d = \gcd(a, b)</math>. There exist integers <math>x</math> and <math>y</math> such that: <math display="block">ax + by = d</math></p> <p>Using the example of <math>\gcd(330, 156)</math>:</p> $6 = 18 - 12 \times 1$ $6 = 18 + 12 \times (-1)$ $6 = 18 + (156 - 18 \times 8) \times (-1)$ $6 = 156 \times (-1) + 18 \times 9$ $6 = 156 \times (-1) + (330 - 156 \times 2) \times 9$ $6 = 330 \times 9 + 156 \times (-19)$ <p>So <math>x = 9</math> and <math>y = -19</math>.</p> <p>Take note there are multiple solutions to <math>x</math> and <math>y</math> (non-uniqueness of Bézout's Identity). Once a solution pair <math>(x, y)</math> is found, additional pairs may be generated by <math>(x + \frac{kb}{d}, y - \frac{ka}{d})</math>, where <math>k</math> is any integer.</p> <p><b>Definition 4.5.4 – Relatively Prime</b> Integers <math>a</math> and <math>b</math> are relatively prime (or coprime) if, and only if, <math>\gcd(a, b) = 1</math>.</p> <p><b>Proposition 4.5.5</b> For any non-zero integers <math>a</math> and <math>b</math>, if <math>c</math> is a common divisor of <math>a</math> and <math>b</math>, then <math>c \mid \gcd(a, b)</math>.</p>
	Express $(109)_{10}$ in base 2: $109 = 2 \times 54 + 1$ $54 = 2 \times 27 + 0$ $27 = 2 \times 13 + 1$ $13 = 2 \times 6 + 1$ $6 = 2 \times 3 + 0$ $3 = 2 \times 1 + 1$ $1 = 2 \times 0 + 1$ $(109)_{10} = (1101101)_2$	<ol style="list-style-type: none"><li>1. Use the quotient-remainder theorem to find the binary representation of 109.</li><li>2. Read the remainders from bottom up.</li><li>3. Read the binary representation from right to left: <math>(2^0 \cdot 1) + (2^1 \cdot 0) + (2^2 \cdot 1) + (2^3 \cdot 1) + (2^4 \cdot 0) + (2^5 \cdot 1) + (2^6 \cdot 1)</math></li></ol>																					
$(10110101)_2$ to base 10: $(2^0 \cdot 1) + (2^1 \cdot 0) + (2^2 \cdot 1) + (2^3 \cdot 0) + (2^4 \cdot 1) + (2^5 \cdot 1) + (2^6 \cdot 0) + (2^7 \cdot 0)$ $= (181)_{10}$	$(10110101)_2$ to base 8: $181 = 8 \times 22 + 5$ $22 = 8 \times 2 + 6$ $2 = 8 \times 0 + 2$ So $(10110101)_2 = (265)_8$																						
0000 = 0	0001 = 1	0010 = 2	0011 = 3																				
0100 = 4	0101 = 5	0110 = 6	0111 = 7																				
1000 = 8	1001 = 9	1010 = A	1011 = B																				
1100 = C	1101 = D	1110 = E	1111 = F																				

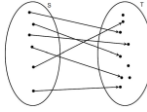
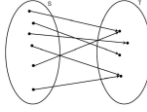
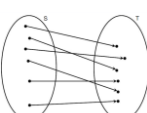
Number Theory	<p><b>Definition 4.6.1 – Least Common Multiple</b> For any non-zero integers <math>a</math> and <math>b</math>, their least common multiple, denoted <math>\text{lcm}(a, b)</math>, is the positive integer <math>m</math> such that:</p> <ol style="list-style-type: none"> <li><math>a \mid m</math> and <math>b \mid m</math></li> <li><math>\forall c \in \mathbb{Z}</math>, if <math>a \mid c</math> and <math>b \mid c</math>, then <math>m \leq c</math></li> </ol> <p>Take note that <math>\text{gcd}(a, b) \times \text{lcm}(a, b) = ab</math>.</p> <p><b>Definition 4.7.1 – Congruence Modulo</b> Let <math>m</math> and <math>n</math> be integers, and let <math>d</math> be a positive integer. We say that <math>m</math> is congruent to <math>n</math> modulo <math>d</math>, denoted as <math>m \equiv n \pmod{d}</math>, if, and only if, <math>d \mid (m - n)</math>.</p> $m \equiv n \pmod{d} \Leftrightarrow d \mid (m - n)$ <p><b>Theorem 8.4.1 (Epp) – Modular Equivalences</b> Let <math>a, b</math> and <math>n</math> be any integers and suppose <math>n &gt; 1</math>. The following statements are all equivalent:</p> <ol style="list-style-type: none"> <li><math>n \mid (a - b)</math></li> <li><math>a \equiv b \pmod{n}</math></li> <li><math>a = b + kn</math>, for some integer <math>k</math></li> <li><math>a</math> and <math>b</math> have the same non-negative remainder when divided by <math>n</math></li> <li><math>a \bmod n = b \bmod n</math></li> </ol> <p><b>Theorem 8.4.3 (Epp) – Modulo Arithmetic</b> Let <math>a, b, c, d</math> and <math>n</math> be integers with <math>n &gt; 1</math> and suppose <math>a \equiv c \pmod{n}</math> and <math>b \equiv d \pmod{n}</math>, then:</p> <ol style="list-style-type: none"> <li><math>(a + b) \equiv (c + d) \pmod{n}</math></li> <li><math>(a - b) \equiv (c - d) \pmod{n}</math></li> <li><math>ab \equiv cd \pmod{n}</math></li> <li><math>a^m \equiv c^m \pmod{n}</math>, for all positive integers <math>m</math></li> </ol>	<p><b>Corollary 8.4.4 (Epp)</b> Let <math>a, b</math> and <math>n</math> be integers with <math>n &gt; 1</math>. Then:</p> <ol style="list-style-type: none"> <li><math>ab \equiv [(a \bmod n)(b \bmod n)] \pmod{n}</math></li> <li><math>a^m \equiv [(a \bmod n)^m] \pmod{n}</math>, for a positive integers <math>m</math></li> </ol> <p><b>Definition 4.7.2 – Multiplicative Inverse of Modulo <math>n</math></b> For any integers <math>a</math> and <math>n</math> with <math>n &gt; 1</math>, if an integer <math>s</math> such that <math>as \equiv 1 \pmod{n}</math>, then <math>s</math> is called the multiplicative inverse of <math>a \bmod n</math>, and the inverse may be written as <math>a^{-1}</math>.</p> <p>By commutative law (which still applies in modulo arithmetic), <math>a^{-1}a \equiv 1 \pmod{n}</math>.</p> <p><b>Theorem 4.7.3 – Existence of Multiplicative Inverses</b> For any integer <math>a</math>, its multiplicative inverse modulo <math>n</math> (where <math>n &gt; 1</math>) <math>a^{-1}</math> exists if, and only if, <math>a</math> and <math>n</math> are coprime.</p> <p><b>Corollary 4.7.4 – Special Case: <math>n</math> is Prime</b> If <math>n = p</math> is a prime number, then all the integers in the range <math>0 &lt; a &lt; p</math> have multiplicative inverses modulo <math>p</math>.</p> <p><b>Theorem 8.4.9 (Epp)</b> For all integers <math>a, b, c</math> and <math>n</math> with <math>n &gt; 1</math> and <math>a</math> and <math>n</math> are coprime, if <math>ab \equiv ac \pmod{n}</math>, then <math>b \equiv c \pmod{n}</math>.</p>
---------------	--	---

Sequences	<p><b>5.3.1 – Arithmetic Sequence</b> Explicit Formula: <math>a_n = a_0 + nd</math> Sum of first <math>n</math> terms: <math>S_n = \frac{n}{2}(2a + (n - 1)d)</math></p> <p><b>5.3.2 – Geometric Sequence</b> Explicit Formula: <math>a_n = a_0 r^n</math> Sum of first <math>n</math> terms: <math>S_n = \frac{a(r^n - 1)}{r - 1}</math> Sum to infinity (where <math> r  &lt; 1</math>): <math>S_\infty = \frac{a}{1 - r}</math></p>	<p><b>5.3.3 – Square Numbers</b> Explicit Formula: <math>\square_n = n^2</math></p> <p><b>5.3.4 – Triangle Numbers</b> Explicit Formula: <math>\triangle_n = \frac{n(n+1)}{2}</math></p> <p><b>5.3.5 – Fibonacci Numbers</b> Explicit Formula: <math>F_n = \frac{(1+\sqrt{5})^n - (1-\sqrt{5})^n}{2^n \sqrt{5}}</math></p>	<p><b>5.3.6 – Binomial Numbers (Pascal's/Yang Hui's Triangle)</b></p> $  \begin{array}{ccccccc}  & & & & 1 & & & & \\  & & & 1 & & 1 & & & \\  & & 1 & & 2 & & 1 & & \\  & 1 & & 3 & & 3 & & 1 & \\  & 1 & 4 & & 6 & & 4 & 1 & \\  1 & 5 & 10 & & 10 & & 5 & 1 &   \end{array}  $ <p>Explicit Formula (entry in <math>n</math>th row and <math>k</math>th column):</p> $\binom{n}{k} = \binom{n-1}{k-1} + \binom{n-1}{k}$	<p><b>5.4.0 – Solving Recurrences</b></p> <ol style="list-style-type: none"> <li>Look it up.</li> <li>Guess and check (aka iteration).</li> <li>Use formula (Theorem 5.8.1/5.8.3 (Epp)).</li> </ol>
-----------	--	--	---	---

Sets	<p><b>Definition 6.3.1 – Empty Set</b> An empty set has no element, and is denoted as <math>\emptyset</math> or <math>\{\}</math>.</p> <p><b>Theorem 6.2.4 (Epp) – An Empty Set is a Subset of all Sets</b> <math>\forall X \forall Z ((\forall Y \sim (Y \in X)) \rightarrow (X \subseteq Z))</math></p> <p><b>Definition 6.3.2 – Set Equality</b> Two sets are equal if and only if they have the same elements. <math>\forall X \forall Y ((\forall Z (Z \in X \Leftrightarrow Z \in Y)) \Leftrightarrow X = Y)</math></p>	<p><b>Proposition 6.3.3</b> For any two sets <math>X</math> and <math>Y</math>, <math>X</math> is a subset of <math>Y</math> and <math>Y</math> is a subset of <math>X</math> if, and only if, <math>X = Y</math>. <math>\forall X \forall Y ((Z \subseteq X \wedge Z \subseteq Y) \Leftrightarrow X = Y)</math></p> <p><b>Corollary 6.2.5 (Epp) – The Empty Set is Unique</b> <math>\forall X_1 \forall X_2 ((\forall Y (\sim (Y \in X_1)) \wedge (\sim (Y \in X_2))) \rightarrow X_1 = X_2)</math></p>	<p><b>Definition 6.3.4 – Power Set</b> Given any set <math>S</math>, the power set of <math>S</math>, denoted by <math>P(S)</math>, or <math>2^S</math>, is the set whose elements are all the subsets of <math>S</math>, nothing less and nothing more. That is, given set <math>S</math>, if <math>T = P(S)</math>, then:</p> $\forall X ((X \in T \Leftrightarrow X \subseteq S))$ <p>If <math>S</math> has <math>n</math> elements, then <math>2^S</math> has <math>2^n</math> elements.</p>
------	---	--	--

Sets	<p><b>Definition 6.4.1 – Union</b> Let <math>S</math> be a set of sets, then we say that <math>T</math> is the union of the sets in <math>S</math>, and write:</p> $T = \bigcup S = \bigcup_{X \in S} X$ <p>If, and only if, each element of <math>T</math> belongs to some set in <math>S</math>, nothing less and nothing more. That is, given <math>S</math>, the set <math>T</math> is such that:</p> $\forall Y ((Y \in T) \leftrightarrow \exists Z ((Z \in S) \wedge (Y \in Z)))$ <p>For two sets <math>A</math> and <math>B</math>, we may simply write <math>T = A \cup B</math>.</p> <p><b>Proposition 6.4.2</b> Let <math>A</math>, <math>B</math> and <math>C</math> be sets. Then:</p> <ul style="list-style-type: none"> <li>• <math>\bigcup \emptyset = \bigcup_{A \in \emptyset} A = \emptyset</math></li> <li>• <math>\bigcup \{A\} = A</math></li> <li>• <math>A \cup \emptyset = A</math></li> <li>• <math>A \cup B = B \cup A</math></li> <li>• <math>A \cup (B \cup C) = (A \cup B) \cup C</math></li> <li>• <math>A \cup A = A</math></li> <li>• <math>A \subseteq B \leftrightarrow A \cup B = B</math></li> </ul>	<p><b>Definition 6.4.3 – Intersection</b> Let <math>S</math> be a non-empty set of sets. The intersection of the sets in <math>S</math> is the set <math>T</math> whose elements belong to all the sets in <math>S</math>, nothing less and nothing more. That is, given <math>S</math>, the set <math>T</math> is such that:</p> $\forall Y ((Y \in T) \leftrightarrow \forall Z ((Z \in S) \rightarrow (Y \in Z)))$ <p>We write:</p> $T = \bigcap S = \bigcap_{X \in S} X$ <p>For two sets <math>A</math> and <math>B</math>, we may simply write <math>T = A \cap B</math>.</p> <p><b>Proposition 6.4.4</b> Let <math>A</math>, <math>B</math> and <math>C</math> be sets. Then:</p> <ul style="list-style-type: none"> <li>• <math>A \cap \emptyset = \emptyset</math></li> <li>• <math>A \cap B = B \cap A</math></li> <li>• <math>A \cap (B \cap C) = (A \cap B) \cap C</math></li> <li>• <math>A \subseteq B \leftrightarrow A \cap B = A</math></li> </ul> <p><u>Distributivity Laws:</u></p> <ul style="list-style-type: none"> <li>• <math>A \cap (B \cup C) = (A \cap B) \cup (A \cap C)</math></li> <li>• <math>A \cup (B \cap C) = (A \cup B) \cap (A \cup C)</math></li> </ul> <p><b>Definition 6.4.5 – Disjoint</b> Let <math>S</math> and <math>T</math> be two sets. <math>S</math> and <math>T</math> are disjoint if, and only if, <math>A \cap B = \emptyset</math>.</p> <p><b>Definition 6.4.6 – Mutually Disjoint</b> Let <math>V</math> be a set of sets. The sets <math>T \subseteq V</math> are mutually disjoint if, and only if, every two distinct sets are disjoint.</p> $\forall X, Y \in V (X \neq Y \rightarrow X \cap Y = \emptyset)$	<p><b>Definition 6.4.7 – Partition</b> Let <math>S</math> be a set, and let <math>V</math> be a set of non-empty subsets of <math>S</math>. Then <math>V</math> is called a partition of <math>S</math> if, and only if, the sets in <math>V</math> are mutually disjoint and the union of the sets in <math>V</math> equals <math>S</math>.</p>  <p><b>Definition 6.4.8 – Non-symmetric Difference</b> Let <math>S</math> and <math>T</math> be two sets. The (non-symmetric) difference (or relative complement) of <math>S</math> and <math>T</math>, denoted <math>S - T^3</math> is the set whose elements belong to <math>S</math> and do not belong to <math>T</math>, nothing less and nothing more.</p> $\forall X ((X \in S - T) \leftrightarrow (X \in S \wedge \neg (X \in T)))$  <p><b>Definition 6.4.9 – Symmetric Difference</b> Let <math>S</math> and <math>T</math> be two sets. The symmetric difference of <math>S</math> and <math>T</math>, denoted <math>S \oplus T^4</math> is the set whose elements belong to <math>S</math> or <math>T</math> but not both, nothing less and nothing more.</p> $\forall X ((X \in S \oplus T) \leftrightarrow (X \in S \oplus X \in T))$  <p><b>Definition 6.4.10 – Set Complement</b> Let <math>U</math> be the Universal set (or the Universe of Discourse) and let <math>A</math> be a subset of <math>U</math>. Then, the complement (or absolute complement) of <math>A</math>, denoted <math>A^C</math>, is <math>U - A</math>.</p> 
Relations	<p><b>Definition 8.1.1 – Ordered Pair</b> Let <math>S</math> be a non-empty set, and let <math>x, y</math> be two elements in <math>S</math>. The ordered pair, denoted <math>(x; y)</math>, is a mathematical object in which the first element of the pair is <math>x</math> and the second element is <math>y</math>. Two ordered pairs <math>(x, y)</math> and <math>(a, b)</math> are equal if, and only if, <math>x = a</math> and <math>y = b</math>.</p> <p><b>Definition 8.1.2 – Ordered n-tuple</b> Let <math>n</math> be a positive integer and let <math>x_1, x_2, \dots, x_n</math> be (not necessarily distinct) elements. The ordered <math>n</math>-tuple, <math>(x_1, x_2, \dots, x_n)</math>, consists of <math>x_1, x_2, \dots, x_n</math> together with the ordering: first <math>x_1</math>, then <math>x_2</math>, and so forth up to <math>x_n</math>. An ordered 2-tuple is called an ordered pair, and an ordered 3-tuple is called an ordered triple.</p>	<p><b>Definition 8.1.3 – Cartesian Product</b> Let <math>S</math> and <math>T</math> be two sets. The Cartesian product (or cross product) of <math>S</math> and <math>T</math>, noted <math>S \times T</math> is the set such that:</p> $\forall X \forall Y ((X, Y) \in S \times T \leftrightarrow (X \in S) \wedge (Y \in T))$ <p><b>Definition 8.1.4 – Generalised Cartesian Product</b> Given sets <math>A_1, A_2, \dots, A_n</math>, the Cartesian product of <math>A_1, A_2, \dots, A_n</math>, denoted <math>A_1 \times A_2 \times \dots \times A_n</math>, is the set of all ordered <math>n</math>-tuples <math>(a_1, a_2, \dots, a_n)</math> where <math>a_1 \in A_1, a_2 \in A_2, \dots, a_n \in A_n</math>.</p> <p><b>Definition 8.2.1 – Binary Relation</b> Let <math>S</math> and <math>T</math> be two sets. A binary relation from <math>S</math> to <math>T</math>, noted <math>R</math>, is a subset of the Cartesian product <math>S \times T</math>.</p> <p><b>Definition 8.2.2 – Domain</b> The domain of <math>R</math> is the set <math>\text{Dom}(R) = \{s \in S \mid \exists t \in T (s R t)\}</math>.</p>	<p><b>Definition 8.2.3 – Image/Range</b> The image (or the range) of <math>R</math> is the set <math>\text{Im}(R) = \{t \in T \mid \exists s \in S (s R t)\}</math>.</p> <p><b>Definition 8.2.4 – Co-domain</b> The co-domain of <math>R</math> is the set <math>\text{coDom}(R) = T</math>.</p> <p><b>Proposition 8.2.5</b> Let <math>R</math> be a binary relation. Then <math>\text{Im}(R) \subseteq \text{coDom}(R)</math>.</p> <p><b>Definition 8.2.6</b> Let <math>S</math> and <math>T</math> be sets. Let <math>R \subseteq S \times T</math> be a binary relation. The inverse of the relation <math>R</math>, denoted <math>R^{-1}</math>, is the relation from <math>T</math> to <math>S</math> such that:</p> $\forall s \in S, \forall t \in T (t R^{-1} s \leftrightarrow s R t)$

Relations	<p><b>Definition 8.2.7</b> Let <math>S_i</math>, for <math>i = 1</math> to <math>n</math>, be <math>n</math> sets. An <math>n</math>-ary relation on the sets <math>S_i</math>, denoted <math>R</math>, is a subset of the Cartesian product <math>\prod_{i=1}^n S_i</math>. We call <math>n</math> the arity or degree of the relation.</p> <p><b>Definition 8.2.8</b> Let <math>S</math>, <math>T</math> and <math>U</math> be sets. Let <math>R \subseteq S \times T</math> be a relation. Let <math>R' \subseteq T \times U</math> be a relation. The composition of <math>R</math> with <math>R'</math>, denoted <math>R' \circ R</math>, is the relation from <math>S</math> to <math>U</math> such that:  <math display="block">\forall x \in S, \forall z \in U (x R' \circ R z \leftrightarrow (\exists y \in T (x R y \wedge y R' z))).</math></p> <p><b>Proposition 8.2.9 – Composition is Associative</b> Let <math>S</math>, <math>T</math>, <math>U</math> and <math>V</math> be sets. Let <math>R \subseteq S \times T</math> be a relation. <math>R' \subseteq T \times U</math> be a relation. Let <math>R'' \subseteq U \times V</math> be a relation.  <math display="block">R'' \circ (R' \circ R) = (R'' \circ R') \circ R = R'' \circ R' \circ R</math></p> <p><b>Proposition 8.2.10</b> Let <math>S</math>, <math>T</math>, and <math>U</math> be sets. Let <math>R \subseteq S \times T</math> be a relation. <math>R' \subseteq T \times U</math> be a relation.  <math display="block">(R' \circ R)^{-1} = R^{-1} \circ R'^{-1}</math></p> <p><b>Definition 8.3.1 – Reflexive</b> <math>R</math> is said to be reflexive if, and only if, <math>\forall x \in A (x R x)</math>.</p> <p><b>Definition 8.3.2 – Symmetric</b> <math>R</math> is said to be symmetric if, and only if, <math>\forall x, y \in A (x R y \rightarrow y R x)</math>.</p> <p><b>Definition 8.3.3 – Transitive</b> <math>R</math> is said to be transitive if, and only if, <math>\forall x, y, z \in A ((x R y \wedge y R z) \rightarrow x R z)</math>.</p> <p><b>Definition 8.3.4 – Equivalence Relation</b> <math>R</math> is called an equivalence relation if and only if <math>R</math> is reflexive, symmetric, and transitive.</p> <p><b>Definition 8.3.5 – Equivalence Class</b> Let <math>x \in A</math>. The equivalence class of <math>x</math>, denoted <math>[x]</math>, is the set of all elements <math>y \in A</math> that are in relation with <math>x</math>.  <math display="block">[x] = \{y \in A \mid x R y\}</math></p>	<p><b>Theorem 8.3.4 (Epp) – Partition Induced by an Equivalence Relation</b> Let <math>R</math> be an equivalence relation on a set <math>A</math>. Then the set of distinct equivalence classes form a partition of <math>A</math>.</p> <p><b>Lemma 8.3.2 (Epp)</b> Let <math>R</math> be an equivalence relation on a set <math>A</math> and let <math>a</math> and <math>b</math> be two elements in <math>A</math>. if <math>a R b</math> then <math>[a] = [b]</math>.</p> <p><b>Lemma 8.3.3 (Epp)</b> If <math>R</math> be an equivalence relation on a set <math>A</math> and <math>a</math> and <math>b</math> are elements in <math>A</math>, then either <math>[a] \cap [b] = \emptyset</math> or <math>[a] = [b]</math>.</p> <p><b>Theorem 8.3.1 (Epp) – Equivalence Relation Induced by a Partition</b> Given a partition <math>S_1, S_2, \dots</math> of a set <math>A</math>, there exists an equivalence relation <math>R</math> on <math>A</math> whose equivalence classes make up precisely that partition.</p> <p><b>Definition 8.5.1 – Transitive Closure</b> Let <math>A</math> be a set. Let <math>R</math> be a relation on <math>A</math>. The transitive closure of <math>R</math>, denoted <math>R^t</math>, is a relation that satisfies these three properties:  1. <math>R^t</math> is transitive.  2. <math>R \subseteq R^t</math>.  3. If <math>S</math> is any other transitive relation such that <math>R \subseteq S</math>, then <math>R^t \subseteq S</math>.</p> <p><b>Proposition 8.5.2</b> Let <math>R</math> be a relation on set <math>A</math>. Then,  <math display="block">R^t = \bigcup_{i=1}^{\infty} R^i</math></p> <p><b>Definition 8.6.1 – Anti-symmetric</b> <math>R</math> is said to be anti-symmetric if, and only if,  <math display="block">\forall x \in A, \forall y \in A ((x R y \wedge y R x) \rightarrow x = y)</math></p> <p><b>Definition 8.6.2 – Partial Order</b> <math>R</math> is said to be a partial order if, and only if, it is reflexive, anti-symmetric and transitive.</p>	<p><b>Hasse Diagrams</b></p> <ul style="list-style-type: none"> <li>- Draw the directed graph so that all arrows point upwards.</li> <li>- Eliminate all self-loops.</li> <li>- Eliminate all arrows implied by the transitive property.</li> <li>- Remove the direction of the arrows.</li> </ul> <p><b>Definition 8.6.3 – Comparable</b> Let <math>\leq</math> be a partial order on a set <math>A</math>. Elements <math>a, b</math> of <math>A</math> are said to be comparable if, and only if, either <math>a \leq b</math> or <math>b \leq a</math>. Otherwise, <math>a</math> and <math>b</math> are called noncomparable.</p> <p><b>Definition 8.6.4 – Total Order</b> Let <math>\leq</math> be a partial order on a set <math>A</math>. <math>\leq</math> is said to be a total order if, and only if, <math>\forall x, y \in A (x \leq y \vee y \leq x)</math>. In other words, <math>\leq</math> is a total order if <math>\leq</math> is a partial order and all <math>x</math> and <math>y</math> are comparable.</p> <p><b>Definition 8.6.5 – Maximal</b> An element <math>x</math> is a maximal element if, and only if, <math>\forall y \in A (x \leq y \rightarrow x = y)</math>.</p> <p><b>Definition 8.6.6 – Maximum</b> An element <math>T</math> is the maximum element if, and only if, <math>\forall x \in A (x \leq T)</math>.</p> <p><b>Definition 8.6.7 – Minimal</b> An element <math>x</math> is a minimal element if, and only if, <math>\forall y \in A (y \leq x \rightarrow x = y)</math>.</p> <p><b>Definition 8.6.8 – Minimum</b> An element <math>\perp</math> is the minimum element if, and only if, <math>\forall x \in A (\perp \leq x)</math>.</p> <p><b>Definition 8.6.9 – Well Ordered</b> Let <math>\leq</math> be a total order on a set <math>A</math>. <math>A</math> is well ordered if, and only if, every non-empty subset of <math>A</math> contains a minimum element, formally:  <math display="block">\forall S \in \mathcal{P}(A) (S \neq \emptyset \rightarrow (\exists x \in S \forall y \in S (x \leq y)))</math></p>
-----------	--	---	---

Functions	<p><b>Definition 7.1.1 – Function</b> Let <math>f</math> be a relation such that <math>f \subseteq S \times T</math>. Then <math>f</math> is a function from <math>S</math> to <math>T</math>, denoted <math>f : S \rightarrow T</math> if, and only if,  <math display="block">\forall x \in S, \exists y \in T (x f y \wedge (\forall z \in T (x f z \rightarrow y = z)))</math></p> <p><b>Definition 7.1.2 – Pre-image</b> Let <math>f : S \rightarrow T</math> be a function. Let <math>x \in S</math>. Let <math>y \in T</math> such that <math>f(x) = y</math>. Then <math>x</math> is called the pre-image of <math>y</math>.</p> <p><b>Definition 7.1.3 – Inverse Image (1)</b> Let <math>f : S \rightarrow T</math> be a function. Let <math>y \in T</math>. The inverse image of <math>y</math> is the set of all its pre-images: <math>\{x \in S \mid f(x) = y\}</math>.</p> <p><b>Definition 7.1.4 – Inverse Image (2)</b> Let <math>f : S \rightarrow T</math> be a function. Let <math>U \subseteq T</math>. The inverse image of <math>U</math> is the set of all the pre-images of all elements of <math>U</math>:  <math>\{x \in S \mid \exists y \in U, f(x) = y\}</math>.</p> <p><b>Definition 7.1.5 – Restriction</b> Let <math>f : S \rightarrow T</math> be a function. Let <math>U \subseteq S</math>. The restriction of <math>f</math> to <math>U</math> is the set <math>\{(x, y) \in U \times T \mid f(x) = y\}</math>.</p>	<p><b>Definition 7.2.1 – Injective</b> Let <math>f : S \rightarrow T</math> be a function. <math>f</math> is injective if, and only if,  <math display="block">\forall y \in T, \forall x_1, x_2 \in S ((f(x_1) = y \wedge f(x_2) = y) \rightarrow x_1 = x_2)</math>  We also say that <math>f</math> is an injection/one-to-one.  Every dot in <math>T</math> has at most one incoming arrow.</p> <p><b>Definition 7.2.2 – Surjective</b> Let <math>f : S \rightarrow T</math> be a function. <math>f</math> is surjective if, and only if,  <math display="block">\forall y \in T, \exists x \in S (f(x) = y)</math>  We also say that <math>f</math> is a surjection/onto.  Every dot in <math>T</math> has at least one incoming arrow.</p> <p><b>Definition 7.2.3 – Bijective</b> Let <math>f : S \rightarrow T</math> be a function. <math>f</math> is bijective if, and only if, <math>f</math> is injective and <math>f</math> is surjective. We also say that <math>f</math> is a bijection.  Every dot in <math>T</math> has exactly one incoming arrow.</p> <p><b>Proposition 7.2.4 – Inverse</b> Let <math>f : S \rightarrow T</math> be a function and let <math>f^{-1}</math> be the inverse relation of <math>f</math> from <math>T</math> to <math>S</math>. Then <math>f</math> is bijective if, and only if, <math>f^{-1}</math> is a function.</p>	<p><b>Proposition 7.3.1 – Composition</b> Let <math>f : S \rightarrow T</math> be a function. Let <math>g : T \rightarrow U</math> be a function. The composition of <math>f</math> and <math>g</math>, <math>g \circ f</math>, is a function from <math>S</math> to <math>U</math>.</p> <p><b>Definition 7.3.2 – Identity Function</b> Given a set <math>A</math>, the identity function <math>I_A</math> from <math>A</math> to <math>A</math> is:  <math display="block">\forall x \in A, (I_A(x) = x)</math></p> <p><b>Proposition 7.3.3</b> Let <math>f : A \rightarrow A</math> be an injective function on <math>A</math>. Then <math>f^{-1} \circ f = I_A</math>.</p> <p><b>Definition 7.3.4 – n-ary Operation</b> An <math>n</math>-ary operation on set <math>A</math> is a function <math>f : \prod_1^n A \rightarrow A</math>. <math>n</math> is called the arity or degree of the operation.</p> <p><b>Definition 7.3.5 – Unary Operation</b> An unary operation on a set <math>A</math> is a function <math>f : A \rightarrow A</math>.</p> <p><b>Definition 7.3.6 – Binary Operation</b> A binary operation on a set <math>A</math> is a function <math>f : A \times A \rightarrow A</math>.</p>
			
			
			

Counting & Probability	<p><b>Definition – Sample Space</b> A sample space is the set of all possible outcomes of a random process or experiment.</p> <p><b>Notation</b> For a finite set <math>A</math>, <math>N(A)</math> denotes the number of elements in <math>A</math>.</p> <p><b>Equally Likely Probability Formula</b> If <math>S</math> is a finite sample space in which all outcomes are equally likely and <math>E</math> is an event in <math>S</math>, then the probability of <math>E</math>, denoted <math>P(E)</math>, is:  <math display="block">P(E) = \frac{\text{Number of outcomes in } E}{\text{Total number of outcomes in } S} = \frac{N(E)}{N(S)}</math></p> <p><b>Theorem 9.1.1 (Epp) – The Number of Elements in a List</b> If <math>m</math> and <math>n</math> are integers and <math>m \leq n</math>, then there are <math>n - m + 1</math> integers from <math>m</math> to <math>n</math> inclusive.</p> <p><b>Theorem 9.2.1 (Epp) – Multiplication Rule</b> If an operation consists of <math>k</math> steps and the first step can be completed in <math>n_1</math> ways, the second step in <math>n_2</math> ways (regardless of how the first step was performed), ..., the <math>k</math>th step in <math>n_k</math> ways (regardless of how the preceding steps were performed), then the entire operation can be performed in <math>n_1 \times n_2 \times \dots \times n_k</math> ways.</p>	<p><b>Theorem 9.2.2 (Epp) – Permutations</b> The number of permutations of a set with <math>n \geq 1</math> elements is <math>n!</math>.</p> <p><b>Theorem 9.2.3 (Epp) – r-permutations from a Set of <math>n</math> Elements</b> If <math>n</math> and <math>r</math> are integers and <math>1 \leq r \leq n</math>, then the number of <math>r</math>-permutations of a set of <math>n</math> elements is given by the formulas  <math display="block">P(n, r) = n(n-1)(n-2)\dots(n-r+1) \text{ or } P(n, r) = \frac{n!}{(n-r)!}</math></p> <p><b>Theorem 9.3.1 (Epp) – The Addition Rule</b> Suppose a finite set <math>A</math> equals the union of <math>k</math> distinct mutually disjoint subsets <math>A_1, A_2, \dots, A_k</math>. Then <math>N(A) = N(A_1) + N(A_2) + \dots + N(A_k)</math>.</p> <p><b>Theorem 9.3.2 (Epp) – The Difference Rule</b> If <math>A</math> is a finite set and <math>B</math> is a subset of <math>A</math>, then <math>N(A - B) = N(A) - N(B)</math>.</p> <p><b>Theorem 9.3.3 (Epp) – The Inclusion/Exclusion Rule for 2/3 Sets</b> If <math>A, B</math> and <math>C</math> are any finite sets, then  <math display="block">N(A \cup B) = N(A) + N(B) - N(A \cap B)</math>  and  <math display="block">N(A \cup B \cup C) = N(A) + N(B) + N(C) - N(A \cap B) - N(A \cap C) - N(B \cap C) + N(A \cap B \cap C)</math></p>	<p><b>Theorem 9.4.1 (Epp)– The Pigeonhole Principle</b> For any function <math>f</math> from a finite set <math>X</math> with <math>n</math> elements to a finite set <math>Y</math> with <math>m</math> elements, if <math>n &gt; m</math>, then <math>f</math> is not one-to-one.</p> <p><u>Generalised Pigeonhole Principle:</u> for any function <math>f</math> from a finite set <math>X</math> with <math>n</math> elements to a finite set <math>Y</math> with <math>m</math> elements and for any positive integer <math>k</math>, if <math>k &lt; n/m</math>, then there is some <math>y \in Y</math> such that <math>y</math> is the image of at least <math>k + 1</math> distinct elements of <math>X</math>.</p> <p><u>Contrapositive Form:</u> If for each <math>y \in Y</math>, <math>f^{-1}(y)</math> has at most <math>k</math> elements, then <math>X</math> has at most <math>km</math> elements (<math>n \leq km</math>).</p> <p><b>Theorem 9.4.2 (Epp) – One-to-One &amp; Onto Finite Sets</b> Let <math>X</math> and <math>Y</math> be finite sets with the same number of elements and suppose <math>f</math> is a function from <math>X</math> to <math>Y</math>. Then <math>f</math> is one-to-one if, and only if, <math>f</math> is onto.</p>



**Theorem 9.5.1 (Epp) – Formula For  $\binom{n}{r}$** 

The number of subsets of size  $r$  (or  $r$ -combinations) that can be chosen from a set of  $n$  elements,  $\binom{n}{r}$ , is given by the formulas

$$\binom{n}{r} = \frac{P(n,r)}{r!} \text{ or } \binom{n}{r} = \frac{n!}{r!(n-r)!}$$

where  $n$  and  $r$  are non-negative integers with  $r \leq n$ .

**Theorem 9.5.2 (Epp) – Permutations with Sets of [Identical] Objects**

Suppose a collection of  $n$  objects of which  $n_1$  are of type 1 and are indistinguishable from each other,  $n_2$  are of type 2 and are indistinguishable from each other, ..., and  $n_k$  are of type  $k$  and are indistinguishable from each other. Suppose  $n_1 + n_2 + \dots + n_k = n$ , then the number of distinguishable permutations of the  $n$  objects is

$$\frac{n!}{n_1! n_2! \dots n_k!}$$
**Definition – Multiset**

An  $r$ -combination with repetition allowed, or multiset of size  $r$ , chosen from a set  $X$  of  $n$  elements is an unordered selection of elements taken from  $X$  with repetition allowed. If  $X = \{x_1, x_2, \dots, x_n\}$ , we write an  $r$ -combination with repetition allowed as  $[x_{i_1}, x_{i_2}, \dots, x_{i_r}]$  where each  $x_{i_j}$  is in  $X$  and some of the  $x_{i_j}$  may equal each other.

**Theorem 9.6.1 (Epp) – Number of  $r$ -combinations (Repetition Allowed)**

The number of  $r$ -combinations with repetitions allowed (multisets of size  $r$ ) that can be selected from a set of  $n$  elements is:

$$\binom{r+n-1}{r}$$

This equals the number of ways  $r$  objects can be selected from  $n$  categories of objects with repetitions allowed.

**Which Formula to Use?**

Repetition	Order Matters	Order Doesn't Matter
Allowed	$n^k$	$\binom{k+n-1}{k}$
Not Allowed	$P(n, k)$	$\binom{n}{k}$

**Theorem 9.7.1 (Epp) – Pascal's Formula**

Let  $n$  and  $r$  be positive integers,  $r \leq n$ . Then

$$\binom{n+1}{r} = \binom{n}{r-1} + \binom{n}{r}$$

**Theorem 9.7.2 (Epp) – Binomial Theorem**

Given any real numbers  $a$  and  $b$  and any non-negative integer  $n$ ,

$$(a+b)^n = \sum_{k=0}^n \binom{n}{k} a^{n-k} b^k = a^n + \binom{n}{1} a^{n-1} b^1 + \dots + b^n$$
**Probability Axioms**

Let  $S$  be a sample space. A probability function  $P$  from the set of all events in  $S$  to the set of real numbers satisfies the following axioms:

For all events  $A$  and  $B$  in  $S$ ,

1.  $0 \leq P(A) \leq 1$
2.  $P(\emptyset) = 0$  and  $P(S) = 1$
3. If  $A$  and  $B$  are disjoint ( $A \cap B = \emptyset$ ), then  $P(A \cup B) = P(A) + P(B)$

**Probability of the Complement of an Event**

If  $A$  is any event in a sample space  $S$ , then  $P(A^C) = 1 - P(A)$ .

**Probability of a General Union of Two Events**

If  $A$  and  $B$  are any events in a sample space  $S$ , then  $P(A \cup B) = P(A) + P(B) - P(A \cap B)$ .

**Definition – Expected Value**

Suppose the possible outcomes of an experiment, or random process, are real numbers  $a_1, a_2, \dots, a_n$  which occur with probabilities  $p_1, p_2, \dots, p_n$ . The expected value of the process is

$$\sum_{k=1}^n a_k p_k = a_1 p_1 + a_2 p_2 + a_3 p_3 + \dots + a_n p_n$$

**Definition – Conditional Probability**

Let  $A$  and  $B$  be events in a sample space  $S$ . If  $P(A) \neq 0$ , then the conditional probability of  $B$  given  $A$ , denoted  $P(B|A)$ , is

$$P(B|A) = \frac{P(A \cap B)}{P(A)}$$

**Theorem 9.9.1 (Epp) – Bayes' Theorem**

Suppose that a sample space  $S$  is a union of mutually disjoint events  $B_1, B_2, \dots, B_n$ . Suppose  $A$  is an event in  $S$ , and suppose  $A$  and all the  $B_i$  have non-zero probabilities. If  $k$  is an integer with  $1 \leq k \leq n$ , then

$$P(B_k|A) = \frac{P(A|B_k) \cdot P(B_k)}{P(A|B_1) \cdot P(B_1) + P(A|B_2) \cdot P(B_2) + \dots + P(A|B_n) \cdot P(B_n)}$$

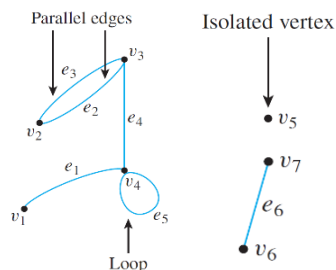
**Definition – Independent Events**

If  $A$  and  $B$  are events in a sample space  $S$ , then  $A$  and  $B$  are independent, if and only if,  $P(A \cap B) = P(A) \cdot P(B)$ .

**Definition – Pairwise and Mutually Independent**

Let  $A, B$  and  $C$  be events in a sample space  $S$ .  $A, B$  and  $C$  are pairwise independent, if and only if, they satisfy conditions 1 – 3 below. They are mutually independent if, and only if, they satisfy all four conditions below.

1.  $P(A \cap B) = P(A) \cdot P(B)$
2.  $P(A \cap C) = P(A) \cdot P(C)$
3.  $P(B \cap C) = P(B) \cdot P(C)$
4.  $P(A \cap B \cap C) = P(A) \cdot P(B) \cdot P(C)$

**10.1 – Definitions and Basic Properties**

A graph  $G$  consists of

- a set of vertices  $V(G)$ , and
- a set of edges  $E(G)$ .

We also write  $G = \{V, E\}$ .

Edges incident on  $v_4$ :  $e_1, e_4$  and  $e_5$

Vertices adjacent to  $v_4$ :  $v_1, v_3$  and  $v_4$

Edges adjacent to  $e_2$ :  $e_3$  and  $e_4$

Where  $V = \{v_1, v_2, v_3, v_4, v_5, v_6, v_7\}$  and  $E = \{e_1, e_2, e_3, e_4, e_5, e_6\}$ ,

$e_1 = \{v_1, v_4\}$ ,  $e_2 = \{v_2, v_3\}$ ,  $e_4 = \{v_3, v_4\}$ ,  $e_5 = \{v_4, v_4\}$  and  $e_6 = \{v_6, v_7\}$ .

Graphs & Trees

Definition – Directed Graph

A directed graph, or digraph,  $G$ , consists of 2 finite sets: a nonempty set  $V(G)$  of vertices and a set  $D(G)$  of directed edges, where each edge is associated with an ordered pair of vertices called its endpoints. If edge  $e$  is associated with the pair  $(v, w)$  of vertices, then  $e$  is said to be the (directed) edge from  $v$  to  $w$ . We write  $e = (v, w)$ .

Definition – Simple Graph

A simple graph is an undirected graph that does not have any loops or parallel edges.

Definition – Complete Graph

A complete graph on  $n$  vertices,  $n > 0$ , denoted  $K_n$ , is a simple graph with  $n$  vertices and exactly one edge connecting each pair of distinct vertices.

Definition – Complete Bipartite Graph

A complete bipartite graph on  $(m, n)$  vertices, where  $m, n > 0$ , denoted  $K_{m,n}$ , is a simple graph with distinct vertices  $v_1, v_2, \dots, v_m$ , and  $w_1, w_2, \dots, w_n$  that satisfies the following properties:  
For all  $i, k = 1, 2, \dots, m$  and for all  $j, l = 1, 2, \dots, n$ ,

- There is an edge from each vertex  $v_i$  to each vertex  $w_j$ .
- There is no edge from any vertex  $v_i$  to any other vertex  $v_k$ .
- There is no edge from any vertex  $w_j$  to any other vertex  $w_l$ .

Definition – Subgraph of a Graph

A graph  $H$  is said to be a subgraph of graph  $G$  if, and only if, every vertex in  $H$  is also a vertex in  $G$ , every edge in  $H$  is also an edge in  $G$ , and every edge in  $H$  has the same endpoints as it has in  $G$ .

Definition – Degree of a Vertex and Total Degree of a Graph

Let  $G$  be a graph and  $v$  a vertex of  $G$ . The degree of  $v$ , denoted  $\deg(v)$ , equals the number of edges that are incident on  $v$ , with an edge that is a loop counted twice. The total degree of  $G$  is the sum of the degrees of all the vertices of  $G$ .

Theorem 10.1.1 (Epp) – The Handshake Theorem

If the vertices of  $G$  are  $v_1, v_2, \dots, v_n$ , where  $n \geq 0$ , then the total degree of  $G = \deg(v_1) + \deg(v_2) + \dots + \deg(v_n) = 2 \times$  (the number of edges of  $G$ ).

Corollary 10.1.2 (Epp)

The total degree of a graph is even.

Proposition 10.1.3 (Epp)

In any graph, there are an even number of vertices of odd degree.

10.2 – Trails, Paths and Circuits

Let  $G$  be a graph, and let  $v$  and  $w$  be vertices of  $G$ . A walk from  $v$  to  $w$  is a finite alternating sequence of adjacent vertices and edges of  $G$ . Thus a walk has the form  $v_0 e_1 v_1 e_2 \dots v_{n-1} e_n v_n$ , where the  $v$ 's represent vertices, the  $e$ 's represent edges,  $v_0 = v$ ,  $v_n = w$ , and for all  $i \in \{1, 2, \dots, n\}$ ,  $v_{i-1}$  and  $v_i$  are the endpoints of  $e_i$ .

- The trivial walk from  $v$  to  $v$  consists of the single vertex  $v$ .
- A trail from  $v$  to  $w$  is a walk from  $v$  to  $w$  that does not contain a repeated edge.
- A path from  $v$  to  $w$  is a trail that does not contain a repeated vertex.
- A closed walk is a walk that starts and ends at the same vertex.
- A circuit (or cycle) is a closed walk that contains at least one edge and does not contain a repeated edge.
- A simple circuit (or simple cycle) is a circuit that does not have any other repeated vertex except the first and last.

	Repeated Edge	Repeated Vertex	Starts And Ends At Same Point	Must Contain At Least One Edge
Walk	Allowed ✓	Allowed ✓	Allowed ✓	No ✗
Trail	No ✗	Allowed ✓	Allowed ✓	No ✗
Path	No ✗	No ✗	No ✗	No ✗
Closed Walk	Allowed ✓	Allowed ✓	Yes ✓	No ✗
Circuit	No ✗	Allowed ✓	Yes ✓	Yes ✓
Simple Circuit	No ✗	First and Last only	Yes ✓	Yes ✓

Definition – Connectedness

Two vertices  $v$  and  $w$  of a graph  $G$  are connected if, and only if, there is a walk from  $v$  to  $w$ . The graph  $G$  is connected if, and only if, given any two vertices  $v$  and  $w$  in  $G$ , there is a walk from  $v$  to  $w$ . Symbolically,  $G$  is connected if, and only if,  $\forall$  vertices  $v, w \in V(G)$ ,  $\exists$  a walk from  $v$  to  $w$ .

Definition – Connected Component

A graph  $H$  is a connected component of a graph  $G$  if, and only if,

- The graph  $H$  is a subgraph of  $G$ ;
- The graph  $H$  is connected; and
- No connected subgraph of  $G$  has  $H$  as a subgraph and contains vertices or edges that are not in  $H$ .

Definition – Euler Circuit

Let  $G$  be a graph. An Euler circuit for  $G$  is a circuit that contains every vertex and every edge of  $G$ . That is, an Euler circuit for  $G$  is a sequence of adjacent vertices and edges in  $G$  that has at least one edge, starts and ends at the same vertex, uses every vertex of  $G$  at least once, and uses every edge of  $G$  exactly once.

Theorem 10.2.2 (Epp)

If a graph has an Euler circuit, then every vertex of the graph has positive even degree.

Contrapositive:

If some vertex of a graph has odd degree, then the graph does not have an Euler circuit.

Theorem 10.2.3 (Epp)

If a graph  $G$  is connected and the degree of every vertex of  $G$  is a positive even integer, then  $G$  has an Euler circuit.

Definition – Euler Trail

Let  $G$  be a graph, and let  $v$  and  $w$  be two distinct vertices of  $G$ . An Euler trail/path from  $v$  to  $w$  is a sequence of adjacent edges and vertices that starts at  $v$ , ends at  $w$ , passes through every vertex of  $G$  at least once, and traverses every edge of  $G$  exactly once.

Theorem 10.2.4 (Epp)

A graph  $G$  has an Euler circuit if, and only if,  $G$  is connected and every vertex of  $G$  has positive even degree.

Corollary 10.2.5 (Epp)

Let  $G$  be a graph, and let  $v$  and  $w$  be two distinct vertices of  $G$ . There is an Euler trail from  $v$  to  $w$  if, and only if,  $G$  is connected,  $v$  and  $w$  have odd degree, and all other vertices of  $G$  have positive even degree.

Definition – Hamiltonian Circuit

Given a graph  $G$ , a Hamiltonian circuit for  $G$  is a simple circuit that includes every vertex of  $G$ . That is, a Hamiltonian circuit for  $G$  is a sequence of adjacent vertices and distinct edges in which every vertex of  $G$  appears exactly once, except for the first and the last, which are the same.

Proposition 10.2.6 (Epp)

If a graph  $G$  has a Hamiltonian circuit, then  $G$  has a subgraph  $H$  with the following properties:

- $H$  contains every vertex of  $G$ .
- $H$  is connected.
- $H$  has the same number of edges as vertices.
- Every vertex of  $H$  has degree 2.

### 10.3 – Matrix Representation of Graphs

#### Definition – Adjacency Matrix of a Directed Graph

Let  $G$  be a directed graph with ordered vertices  $v_1, v_2, \dots, v_n$ . The adjacency matrix of  $G$  is the  $n \times n$  matrix  $A = (a_{ij})$  over the set of non-negative integers such that  $a_{ij}$  = the number of arrows from  $v_i$  to  $v_j$  for all  $i, j = 1, 2, \dots, n$ .

#### Definition – Adjacency Matrix of an Undirected Graph

Let  $G$  be an undirected graph with ordered vertices  $v_1, v_2, \dots, v_n$ . The adjacency matrix of  $G$  is the  $n \times n$  matrix  $A = (a_{ij})$  over the set of non-negative integers such that  $a_{ij}$  = the number of edges connecting  $v_i$  and  $v_j$  for all  $i, j = 1, 2, \dots, n$ .

#### Definition – Symmetric Matrix

An  $n \times n$  square matrix  $A = (a_{ij})$  is called symmetric if, and only if, for all  $i, j = 1, 2, \dots, n$ ,  $a_{ij} = a_{ji}$ .

#### Theorem 10.3.1 (Epp)

Let  $G$  be a graph with connected components  $G_1, G_2, \dots, G_k$ . If there are  $n_i$  vertices in each connected component  $G_i$  and these vertices are numbered consecutively, then the adjacency matrix of  $G$  has the form:

$$\begin{bmatrix} A_1 & 0 & 0 & \dots & 0 & 0 \\ 0 & A_2 & 0 & \dots & 0 & 0 \\ 0 & 0 & A_3 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & 0 & A_k \end{bmatrix}$$

where each  $A_i$  is  $n_i \times n_i$  adjacency matrix of  $G_i$ , for all  $i = 1, 2, \dots, k$ , and the  $0$ 's represent matrices whose entries are all  $0$ s.

#### Definition – nth Power of a Matrix

For any  $n \times n$  matrix  $A$ , the powers of  $A$  are defined as follows:

1.  $A^0 = I$  where  $I$  is the  $n \times n$  identity matrix
2.  $A^n = A A^{n-1}$  for all integers  $n \geq 1$

#### Theorem 10.3.2 (Epp)

If  $G$  is a graph with vertices  $v_1, v_2, \dots, v_m$  and  $A$  is the adjacency matrix of  $G$ , then for each positive integer  $n$  and for all integers  $i, j = 1, 2, \dots, m$ , the  $ij$ -th entry of  $A^n$  = the number of walks of length  $n$  from  $v_i$  to  $v_j$ .

### 10.4 – Isomorphisms of Graphs

#### Definition – Isomorphic Graph

Let  $G$  and  $G'$  be graphs with vertex sets  $V(G)$  and  $V(G')$  and edge sets  $E(G)$  and  $E(G')$  respectively.  $G$  is isomorphic to  $G'$  if, and only if, there exist one-to-one correspondences  $g: V(G) \rightarrow V(G')$  and  $h: E(G) \rightarrow E(G')$  that preserve the edge-endpoint functions of  $G$  and  $G'$  in the sense that for all  $v \in V(G)$  and  $e \in E(G)$ ,  $v$  is an endpoint of  $e \Leftrightarrow g(v)$  is an endpoint of  $h(e)$ .

#### Theorem 10.4.1 (Epp) – Graph Isomorphism is an Equivalence Relation

Let  $S$  be a set of graphs and let  $R$  be the relation of graph isomorphism on  $S$ . Then  $R$  is an equivalence relation on  $S$ .

#### Definition – Invariant for Graph Isomorphism

A property  $P$  is called an invariant for graph isomorphism iff given any graphs  $G$  and  $G'$ , if  $G$  has property  $P$  and  $G'$  is isomorphic to  $G$ , then  $G'$  has property  $P$ .

#### Theorem 10.4.2 (Epp) – Invariants for Graph Isomorphism

Each of the following properties is an invariant for graph isomorphism, where  $n, m$ , and  $k$  are all non-negative integers:

- |                                     |  |
|-------------------------------------|--|
| 1. has $n$ vertices;                | 6. has a simple circuit of length $k$ ;    |
| 2. has $m$ edges;                   | 7. has $m$ simple circuits of length $k$ ; |
| 3. has a vertex of degree $k$ ;     | 8. is connected;                           |
| 4. has $m$ vertices of degree $k$ ; | 9. has an Euler circuit;                   |
| 5. has a circuit of length $k$ ;    | 10. has a Hamiltonian circuit.             |

#### Definition – Isomorphic

If  $G$  and  $G'$  are simple graphs, then  $G$  is isomorphic to  $G'$  iff there exists a one-to-one correspondence  $g$  from the vertex set  $V(G)$  of  $G$  to the vertex set  $V(G')$  of  $G'$  that preserves the edge-endpoint functions of  $G$  and  $G'$  in the sense that for all vertices  $u$  and  $v$  of  $G$ ,  $\{u, v\}$  is an edge in  $G \Leftrightarrow \{g(u), g(v)\}$  is an edge in  $G'$ .

### 10.5 – Trees

#### Definitions – Circuit-Free, Tree, Trivial Tree & Forest

1. A graph is said to be circuit-free if, and only if, it has no circuits.
2. A graph is called a tree if, and only if, it is circuit-free and connected.
3. A trivial tree is a graph that consists of a single vertex.
4. A graph is called a forest if, and only if, it is circuit-free and not connected.

#### Definitions – Terminal Vertex (Leaf) & Internal Vertex

Let  $T$  be a tree. If  $T$  has only one or two vertices, then each is called a terminal vertex (or leaf). If  $T$  has at least three vertices, then a vertex of degree 1 in  $T$  is called a terminal vertex (or leaf), and a vertex of degree greater than 1 in  $T$  is called an internal vertex.

#### Lemma 10.5.1 (Epp)

Any non-trivial tree has at least one vertex of degree 1.

#### Theorem 10.5.2 (Epp)

Any tree with  $n$  vertices ( $n > 0$ ) has  $n - 1$  edges.

#### Lemma 10.5.3 (Epp)

If  $G$  is any connected graph,  $C$  is any circuit in  $G$ , and one of the edges of  $C$  is removed from  $G$ , then the graph that remains is still connected.

#### Theorem 10.5.4 (Epp)

If  $G$  is a connected graph with  $n$  vertices and  $n - 1$  edges, then  $G$  is a tree.

### 10.6 – Rooted Trees

#### Definitions – Rooted Tree, Level & Height

1. A rooted tree is a tree in which there is one vertex that is distinguished from the others and is called the root.
2. The level of a vertex is the number of edges along the unique path between it and the root.
3. The height of a rooted tree is the maximum level of any vertex of the tree.

#### Definitions – Child, Parent, Sibling, Ancestor & Descendent

1. Given the root or any internal vertex  $v$  of a rooted tree, the children of  $v$  are all those vertices that are adjacent to  $v$  and are one level farther away from the root than  $v$ .
2. If  $w$  is a child of  $v$ , then  $v$  is called the parent of  $w$ , and two distinct vertices that are both children of the same parent are called siblings.
3. Given two distinct vertices  $v$  and  $w$ , if  $v$  lies on the unique path between  $w$  and the root, then  $v$  is an ancestor of  $w$ , and  $w$  is a descendant of  $v$ .

#### Definitions – Binary Tree & Full Binary Tree

1. A binary tree is a rooted tree in which every parent has at most two children. Each child is designated either a left child or a right child (but not both), and every parent has at most one left child and one right child.
2. A full binary tree is a binary tree in which each parent has exactly two children.

### Definition – Left/Right Subtree

Given any parent  $v$  in a binary tree  $T$ , if  $v$  has a left child, then the left subtree of  $v$  is the binary tree whose root is the left child of  $v$ , whose vertices consist of the left child of  $v$  and all its descendants, and whose edges consist of all those edges of  $T$  that connect the vertices of the left subtree. The right subtree of  $v$  is defined analogously.

### Theorem 10.6.1 (Epp) – Full Binary Tree Theorem

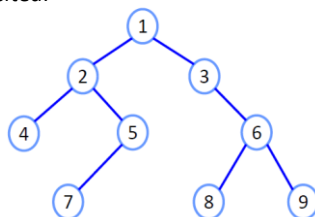
If  $T$  is a full binary tree with  $k$  internal vertices, then  $T$  has a total of  $2k + 1$  vertices and has  $k + 1$  terminal vertices (leaves).

### Theorem 10.6.2 (Epp)

For non-negative integers  $h$ , if  $T$  is any binary tree with height  $h$  and  $t$  terminal vertices (leaves), then  $t \leq 2^h$ . Equivalently,  $\log_2 t \leq h$ .

### Binary Tree Traversal

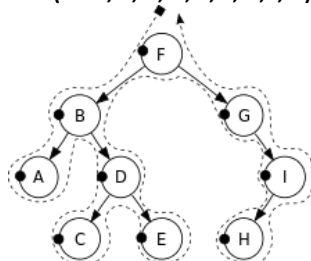
**Breadth-First Search:** In breadth-first search (by E.F. Moore), it starts at the root and visits its adjacent vertices, and then moves to the next level. The figure below shows the order of the vertices visited:



### Depth-First Search:

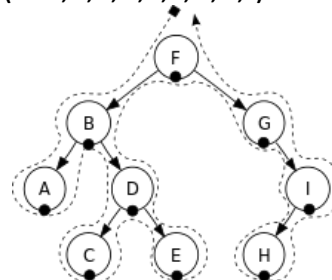
There are three types of depth-first traversal:

#### 1. Pre-order (i.e. F, B, A, D, C, E, G, I, H)



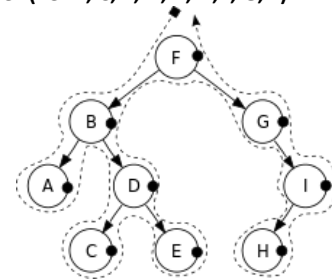
- Print the data of the root (or current vertex)
- Traverse the left subtree by recursively calling the pre-order function
- Traverse the right subtree by recursively calling the pre-order function

#### 2. In-order (i.e. A, B, C, D, E, F, G, H, I)



- Traverse the left subtree by recursively calling the in-order function
- Print the data of the root (or current vertex)
- Traverse the right subtree by recursively calling the in-order function

#### 3. Post-order (i.e. A, C, E, D, B, H, I, G, F)



- Traverse the left subtree by recursively calling the post-order function
- Traverse the right subtree by recursively calling the post-order function
- Print the data of the root (or current vertex)

## 10.7 – Spanning Trees & Shortest Paths

### Definition – Spanning Tree

A spanning tree for a graph  $G$  is a subgraph of  $G$  that contains every vertex of  $G$  and is a tree.

### Proposition 10.7.1

Every connected graph has a spanning tree. Also, any two spanning trees for a graph have the same number of edges.

### Definitions – Weighted Graph & Minimum Spanning Tree

- A weighted graph is a graph for which each edge has an associated positive real number weight. The sum of the weights of all the edges is the total weight of the graph.
- A minimum spanning tree for a connected weighted graph is a spanning tree that has the least possible total weight compared to all other spanning trees for the graph.
- If  $G$  is a weighted graph and  $e$  is an edge of  $G$ , then  $w(e)$  denotes the weight of  $e$  and  $w(G)$  denotes the total weight of  $G$ .

### Algorithm 10.7.1 – Kruskal

**Input:**  $G$  (a connected weighted graph with  $n$  vertices)

#### Algorithm:

- Initialize  $T$  to have all the vertices of  $G$  and no edges.
- Let  $E$  be the set of all edges of  $G$ , and let  $m = 0$ .
- While ( $m < n - 1$ )
  - Find an edge  $e$  in  $E$  of least weight.
  - Delete  $e$  from  $E$ .
  - If addition of  $e$  to the edge set of  $T$  does not produce a circuit, then add  $e$  to the edge set of  $T$  and set  $m = m + 1$
  - End while

**Output:**  $T$  ( $T$  is a minimum spanning tree for  $G$ )

### Algorithm 10.7.2 – Prim

**Input:**  $G$  (a connected weighted graph with  $n$  vertices)

#### Algorithm:

- Pick a vertex  $v$  of  $G$  and let  $T$  be the graph with this vertex only.
- Let  $V$  be the set of all vertices of  $G$  except  $v$ .
- For  $i = 1$  to  $n - 1$ 
  - Find an edge  $e$  of  $G$  such that (1)  $e$  connects  $T$  to one of the vertices in  $V$ , and (2)  $e$  has the least weight of all edges connecting  $T$  to a vertex in  $V$ . Let  $w$  be the endpoint of  $e$  that is in  $V$ .
  - Add  $e$  and  $w$  to the edge and vertex sets of  $T$ , and delete  $w$  from  $V$ .

**Output:**  $T$  ( $T$  is a minimum spanning tree for  $G$ )

### Algorithm 10.7.3 – Dijkstra

Input:  $G$  (a connected simple graph with positive weight for every edge),  $\infty$  (a number greater than the sum of the weights of all the edges in  $G$ ),  $w(u, v)$  (the weight of edge  $\{u, v\}$ ),  $a$  (the source vertex),  $z$  (the destination vertex).

Algorithm:

1. Initialize  $T$  to be the graph with vertex  $a$  and no edges.  
Let  $V(T)$  be the set of vertices of  $T$ , and let  $E(T)$  be the set of edges of  $T$ .
2. Let  $L(a) = 0$ , and for all vertices in  $G$  except  $a$ , let  $L(u) = \infty$ .  
(The number  $L(x)$  is called the label of  $x$ .)
3. Initialize  $v$  to equal  $a$  and  $F$  to be  $\{a\}$ . (The symbol  $v$  is used to denote the vertex most recently added to  $T$ )
4. Let  $\text{Adj}(x)$  denote the set of vertices adjacent to vertex  $x$ . While  $\{z \notin V(T)\}$ 
  - a.  $F \leftarrow (F - \{v\}) \cup \{\text{vertices} \in \text{Adj}(v) \wedge \notin V(T)\}$   
(The set  $F$  is the set of fringe vertices)
  - b. For each vertex  $u \in \text{Adj}(v) \wedge \notin V(T)$ ,  
if  $L(v) + w(v, u) < L(u)$  then  
     $L(u) \leftarrow L(v) + w(v, u)$   
     $D(u) \leftarrow v$
  - c. Find a vertex  $x$  in  $F$  with the smallest label.  
Add vertex  $x$  to  $V(T)$ , and add edge  $\{D(x), x\}$  to  $E(T)$ .  
 $v \leftarrow x$

Output:  $L(z)$  (the length of the shortest path from  $a$  to  $z$ )