MATHEMATICAL FOUNDATION FOR COMPUTER SCIENCE

Prepared by: Er. Ankit Kharel

Nepal college of information technology

LOGIC, INDUCTION AND REASONING

- Proposition and Truth function
- Propositional Logic
- Expressing statements in Logic Propositional Logic
- Rules of Inference
- The predicate Logic
- Validity
- Informal Deduction in Predicate Logic
- Proofs(Informal Proofs & Formal Proofs)
- Elementary Induction
- Complete Induction
- Methods of Tableaux
- Consistency and Completeness of the System

Proposition:

- Declarative statement that is either TRUE or FALSE.

- Symbol ‘T’ for TRUE and ‘F’ for FALSE.

  Examples:

  i) Paris is in France(T).
  ii) Delhi is in Nepal(F).
  iii) $2<4$(T).
  iv) $4=7$(F).

  Example of statement that are not propositions:

  i) What is your name? (This is a Question)
  ii) Do your Homework (This is a command)
  iii) “x” is even number (It depends on the value of x)

- Small alphabets like ‘p’, ’q’ , ’r’ are used to represent prepositions.

  p: Paris is in France.

  q: We live on Earth

Proposition Logic:

- ➢ Deals with proposition also known as Propositional Calculus.

- ➢ First developed by Aristotle.

I) Atomic Proposition:

❖ Which cannot be further broken down.

Example:

"Today is Friday"

II) Compound Proposition:

❖ Which can further be broken down.

❖ Logical operators are used.

Example:

"Ram is intelligent and diligent."

p: "Ram is intelligent"

q: "Ram is diligent"

1.Logical operators/connectives:

- Used to construct compound propositions.
- Some common logical connectives are:

1. NEGATION(NOT) ¬
2. CONJUCTION(AND) ^
3. DISJUNCTION(OR) v
4. EXCLUSIVE OR(XOR) ⊕
5. IMPLICATION(IF-THEN) →

   (Inverse, Converse and Contrapositive)

6. BICONDITIONAL(IF AND ONLY IF) ←→

1.Negation(not):

- If ‘p’ is the proposition , then the negation of ‘p’ is denoted by ‘¬p’.
- ‘¬p’ means “it is not case that p” or simply “not p”.

Examples:

1) p: “Today is Friday”
   ¬p: “It is not the case that today is Friday”
   ¬p: “Today is not Friday”

2) p: “London is in Denmark”
   ¬p: “It is not the case that London is in Denmark”
   ¬p: “London is not in Denmark”

TRUTH TABLE

p	¬p
T	F
F	T

2.conjunction(and):

- If ‘p’ and ‘q’ are two proposition , then the conjunction of ‘p’ and ‘q’ is denoted by ‘p^q’.
- p^q is TRUE only when both ‘p’ and ‘q’ are TRUE, otherwise FALSE.

Examples:

1) p: “Today is Friday”
   q: “It is raining Today”
   p^q: “Today is Friday and it is raining Today”

Truth Table

p	q	p^q
T	T	T
T	F	F
F	T	F
F	F	F

3.disjunction(or):

- If ‘p’ and ‘q’ are two proposition , then the disjunction of ‘p’ and ‘q’ is denoted by ‘pVq’.
- pVq is FALSE when both ‘p’ and ‘q’ are FALSE, otherwise TRUE.

Examples:

1) p: “Today is Friday”
   q: “It is raining Today”
   pVq: “Today is Friday or it is raining Today”

Truth Table

p	q	pVq
T	T	T
T	F	T
F	T	T
F	F	F

4.Exclusive or (xor):

- If ‘p’ and ‘q’ are two proposition , then the Exclusive or of ‘p’ and ‘q’ is denoted by ‘$p \oplus q$’ which means “Either p or q but not both”
- $p \oplus q$ is TRUE when either ‘p’ or ‘q’ is TRUE and FALSE when both are TRUE or both are FALSE.

Examples:

1) p: “Today is Friday”

   q: “It is raining Today”

   $p \oplus q$: “Either today is Friday or it is raining today”

Truth Table

p	q	$p \oplus q$
T	T	F
T	F	T
F	T	T
F	F	F

5.implication (if→then):

- If ‘p’ and ‘q’ are two proposition then the statement “if p then q” is called an implication and denoted by p→q.

- p→q is also called a conditional statement.

- ‘p’ is called ***hypothesis*** or ***antecedent*** or ***premise***.

- ‘q’ is called the ***conclusion*** or ***consequence***.

Some other terminologies used to express p→q are:
✓ If p , then q.
✓ p is sufficient for q
✓ q when p
✓ A necessary condition for p is q
✓ p only if q
✓ q unless ¬p
✓ q follows from p

5.implication (if→then):

Example:

p: “Today is holiday”

q: “The college is closed”

p→q: ”If today is holiday, then the college is closed”

TRUTH TABLE

p	q	p→q
T	T	T
T	F	F
F	T	T
F	F	T

inverse:

$p \rightarrow q$ → $\neg p \rightarrow \neg q$

"if p, then q" *"if not p, then not q"*

p: "Today is holiday" ¬p: "Today is not holiday"

q: "The college is closed" ¬q: "The college is not closed"

$p \rightarrow q$: "If today is holiday, then the college is closed"

$\neg p \rightarrow \neg q$: "If today is not holiday, then the college is not closed"

converse:

$p \rightarrow q$ → $q \rightarrow p$

"if p, then q" *"if q, then p"*

p: "Today is holiday"

q: "The college is closed"

$p \rightarrow q$: "If today is holiday, then the college is closed"

$q \rightarrow p$: "if the college is closed, then today is holiday"

Contra-positive:

$p \rightarrow q$ $\longrightarrow$ $\neg q \rightarrow \neg p$

"if p, then q" *"if not q, then not p"*

p: "Today is holiday"
q: "The college is closed"

¬p: "Today is not holiday"
¬q: "The college is not closed"

$p \rightarrow q$: "If today is holiday, then the college is closed"
$\neg q \rightarrow \neg p$: "If the college is not closed, today is not holiday"

6.Biconditional(if and only if):

- If ‘p’ and ‘q’ are two proposition , then the biconditional statement p←→q is the proposition “p if and only if q”
- (p→q)^(q→p)==p←→q
- These are also called bi-implications.

Examples:

p: “I am breathing”

q: “I am alive”

p←→q : “I am breathing if and only if I am alive”

Truth Table

p	q	p←→q
T	T	T
T	F	F
F	T	F
F	F	T

Operator precedence:

Operator	Precedence (higher the number higher the precedence)
¬	1
∧	2
∨	3
→	4
↔	5

Examples:

1) $\neg p \wedge q$ {Given p =True and q=False}

$=F \wedge F$

$=F$

2) $p \wedge q \vee r$ {Given p=True, q =False, r=True}

$=F \vee T$

$=T$

3) $p \rightarrow q \wedge \neg p$ {Given p=True, q=False}

$=T \rightarrow F \wedge F$

$=T \rightarrow F$

$=F$

4) $(p \wedge q) \rightarrow ((\neg p) \vee q)$ {Given p=True, q=False}

$=(T \wedge F) \rightarrow (F \vee F)$

$=F \rightarrow F$

$=T$

Truth table of compound preposition:

- Construct the truth table of compound proposition $(P \vee \neg Q) \rightarrow (P \wedge Q)$

P	Q	¬Q	PV¬Q	P∧Q	(PV¬Q)→(P∧Q)
T	T	F	T	T	T
T	F	T	T	F	F
F	T	F	F	F	T
F	F	T	T	F	F

Truth table of compound preposition:

- Construct the truth table of compound proposition

1. $(P \vee \neg Q) \rightarrow (P \wedge Q)$

P	Q	¬Q	A=(PV¬Q)	B=(P^Q)	A→B
T	T	F	T	T	T
T	F	T	T	F	F
F	T	F	F	F	T
F	F	T	T	F	F

Truth table of compound preposition:

- Construct the truth table of compound proposition

2. $(P \rightarrow Q) \wedge (Q \rightarrow R)$

P	Q	R	(P→Q)	(Q→R)	(P→Q)^(Q→R)
T	T	T	T	T	T
T	T	F	T	F	F
T	F	T	F	T	F
T	F	F	F	T	F
F	T	T	T	T	T
F	T	F	T	F	F
F	F	T	T	T	T
F	F	F	T	T	T

TRANSLATING ENGLISH SENTENCES:

Examples:

1."You can access the internet from NCIT only if you are a masters student or you are a new student"

Let ,

p: You access the internet from NCIT

q: You are a masters student

r: You are a new student

$$\mathbf{p \rightarrow (q \wedge r)}$$

2."The automated reply can not be sent when file system is full"

Let,

p:The automated reply can be sent

q:File system is full

$$\mathbf{q \rightarrow \neg p}$$

Assignment 1 :

1. What are logical connectives explain each with example and truth table.
2. Construct truth table for
- $\neg(p\wedge q)\vee(r\wedge\neg p)$
- $(p\vee\neg r)\wedge\neg((q\vee r)\vee\neg(r\vee p))$
- $((p\leftrightarrow q) \oplus (\neg p\rightarrow q)) \vee (q\rightarrow \neg r)$
3. Let p, q ,r be:

   p="You have flu"

   q="You miss the final exam"

   r="You pass the course"

   Express each proposition as an English sentence and construct truth table:
   - $p\rightarrow q$
   - $q\rightarrow\neg r$
   - $(p\rightarrow\neg r)\vee(q\rightarrow\neg r)$
4. Translate into mathematical expression
- You can't have voting right if you are mentally unfit and you are not over 18 years.
- Leaders will make correct decision only if you choose a good leader or you raise your voice against incorrect decision

MATHEMATICAL FOUNDATION FOR COMPUTER SCIENCE

Prepared by: Er. Ankit Kharel

Nepal college of information technology

- TAUTOLOGY
- CONTRADICTION
- CONTINGENCY
- PROPOSITIONAL SATISFIABILITY
- LOGICAL EQUIVALENCE

TAUTOLOGY:

- Compound proposition that is always TRUE , not matter what the truth values of the propositional variables that occur in it, is called TAUTOLOGY.

Examples:

a) p V ¬p

p	¬p	p ∧ ¬p
T	F	T
F	T	T

b) (p→q) v (q→p)

p	q	p→q	q→p	(p→q)V(q→p)
T	T	T	T	T
T	F	F	T	T
F	T	T	F	T
F	F	T	T	T

CONTRADICTION:

- Compound proposition that is always FALSE, not matter what the truth values of the propositional variables that occur in it, is called CONTRADICTION.

Examples:

a) p ∧ ¬p

p	¬p	p ∧ ¬p
T	F	F
F	T	F

b) ¬(p^q) ←→ (q^p)

p	q	(p ∧ q)	¬(p^q)	(q^p)	¬(p^q) ←→ (q^p)
T	T	T	F	T	F
T	F	F	T	F	F
F	T	F	T	F	F
F	F	F	T	F	F

CONTINGENGY:

- Compound proposition that is neither a TAUTOLOGY or a CONTRADICTION

Examples:

a) $(p \rightarrow q) \land (q \rightarrow p)$

p	q	p→q	q→p	(p→q) ∧ (q→p)
T	T	T	T	T
T	F	F	T	F
F	T	T	F	F
F	F	T	T	T

SATISFIABILITY:

- Compound proposition is satisfiable is there is at least one true value in its truth table.
- TAUTOLOGY is always satisfiable but satisfiable is not always TAUTOLOGY.

(p→q) ^ (q→p)

p	q	p→q	q→p	(p→q) ^ (q→p)
T	T	T	T	T
T	F	F	T	F
F	T	T	F	F
F	F	T	T	T

SATISFIABLE but not TAUTOLOGY

(p→q)V(q→p)

p	q	p→q	q→p	(p→q)V(q→p)
T	T	T	T	T
T	F	F	T	T
F	T	T	F	T
F	F	T	T	T

SATISFIABLE and alsoTAUTOLOGY

UNSATISFIABILITY:

- Compound proposition is unsatisfiable is there is no true value in its truth table.
- CONTRADICTION is always satisfiable.

¬(p^q) ←→ (q^p)

p	q	(p ^ q)	¬(p^q)	(q^p)	¬(p^q) ←→ (q^p)
T	T	T	F	T	F
T	F	F	T	F	F
F	T	F	T	F	F
F	F	F	T	F	F

VALID & INVALID:

VALID: Compound proposition always VALID when it is a TAUTOLOGY.

INVALID: Compound proposition always INVALID when it is either CONTRADICTION or CONTINGENCY.

SUMMARY

TAUTOLOGY
Always TRUE
Satisfiable
VALID

CONTRADICTION
Always FALSE
unsatisfiable
INVALID

CONTINGENCY
Sometimes TRUE or FALSE
Satisfiable
INVALID

LOGICAL EQUIVALENCES:

- Compound proposition ‘p’ and ‘q’ are logically equivalent is they have same Truth Values in all possible cases.
- Notation: $p \equiv q$ or $p \Leftrightarrow q$

Examples:

a) $\neg(p \vee q)$ and $(\neg p \wedge \neg q)$

p	q	(pVq)	¬(pVq)	¬p	¬q	(¬p∧¬q)
T	T	T	F	F	F	F
T	F	T	F	F	T	F
F	T	T	F	T	F	F
F	F	F	T	T	T	T

Hence, $\neg(p \vee q) \Leftrightarrow (\neg p \wedge \neg q)$

Examples:

b) $(p \rightarrow q)$ and $(\neg p \vee q)$

p	q	$(p \rightarrow q)$	$\neg p$	$(\neg p \vee q)$
T	T	T	F	T
T	F	F	F	F
F	T	T	T	T
F	F	T	T	T

Hence, $(p \rightarrow q) \Leftrightarrow (\neg p \vee q)$

IMPORTANT EQUIVALENCE:

Equivalences	Laws
$p \land \mathbf{T} \equiv p$ $p \lor \mathbf{F} \equiv p$	Identity laws
$p \lor \mathbf{T} \equiv \mathbf{T}$ $p \land \mathbf{F} \equiv \mathbf{F}$	Domination laws
$p \lor p \equiv p$ $p \land p \equiv p$	Idempotent laws
$\neg(\neg p) \equiv p$	Double negation law
$p \lor q \equiv q \lor p$ $p \land q \equiv q \land p$	Commutative laws
$(p \lor q) \lor r \equiv p \lor (q \lor r)$ $(p \land q) \land r \equiv p \land (q \land r)$	Associative laws
$p \lor (q \land r) \equiv (p \lor q) \land (p \lor r)$ $p \land (q \lor r) \equiv (p \land q) \lor (p \land r)$	Distributive laws
$\neg(p \land q) \equiv \neg p \lor \neg q$ $\neg(p \lor q) \equiv \neg p \land \neg q$	De Morgan's laws
$p \lor (p \land q) \equiv p$ $p \land (p \lor q) \equiv p$	Absorption laws
$p \lor \neg p \equiv \mathbf{T}$ $p \land \neg p \equiv \mathbf{F}$	Negation laws

p	T	p ^ T
T	T	T
F	T	F

p	F	p V T
T	F	T
F	F	F

1. Identity Laws

p	T	p V T
T	T	T
F	T	T

p	F	p ^ F
T	F	F
F	F	F

2. Domination Laws

EQUIVALENCE INVOLVING CONDITION:

TABLE 7 Logical Equivalences Involving Conditional Statements.

$p \to q \equiv \neg p \vee q$

$p \to q \equiv \neg q \to \neg p$

$p \vee q \equiv \neg p \to q$

$p \wedge q \equiv \neg(p \to \neg q)$

$\neg(p \to q) \equiv p \wedge \neg q$

$(p \to q) \wedge (p \to r) \equiv p \to (q \wedge r)$

$(p \to r) \wedge (q \to r) \equiv (p \vee q) \to r$

$(p \to q) \vee (p \to r) \equiv p \to (q \vee r)$

$(p \to r) \vee (q \to r) \equiv (p \wedge q) \to r$

1. $(p \to q) \Leftrightarrow (\neg p \vee q)$

p	q	(p→q)	¬p	(¬pVq)
T	T	T	F	T
T	F	F	F	F
F	T	T	T	T
F	F	T	T	T

2. $(p \to q) \Leftrightarrow (\neg q \to \neg p)$

p	q	(p→q)	¬p	¬q	(¬q→ ¬p)
T	T	T	F	F	T
T	F	F	F	T	F
F	T	T	T	F	T
F	F	T	T	T	T

EQUIVALENCE INVOLVING BICONDITION:

TABLE 8 Logical Equivalences Involving Biconditional Statements.

$$p \leftrightarrow q \equiv (p \rightarrow q) \wedge (q \rightarrow p)$$

$$p \leftrightarrow q \equiv \neg p \leftrightarrow \neg q$$

$$p \leftrightarrow q \equiv (p \wedge q) \vee (\neg p \wedge \neg q)$$

$$\neg(p \leftrightarrow q) \equiv p \leftrightarrow \neg q$$

1. (p←→q) ⇔ (p→q)^(q→p)

p	q	(p←→q)	(p→q)	(q→p)	(p→q)^(q→p)
T	T	T	T	T	T
T	F	F	F	T	F
F	T	F	T	F	F
F	F	T	T	T	T

Prove the following are logically equivalent by developing a series of logical equivalence.

1. $\neg(p \rightarrow q) \equiv (p \wedge \neg q)$

<u>solution:</u>

Taking LHS,

$= \neg(p \rightarrow q)$

$= \neg(\neg p \vee q)$-----------{$p \rightarrow q \equiv \neg p \vee q$}

$= \neg(\neg p) \wedge (\neg q)$------{De- Morgan's Law}

$= p \wedge \neg q$---------------{Double Negation Law}

Prove the following are logically equivalent by developing a series of logical equivalence.

1. $\neg(p \vee (\neg p \wedge q)) \equiv (\neg p \wedge \neg q)$

<u>solution:</u>

Taking LHS,

$= \neg(p \vee (\neg p \wedge q))$

$= \neg p \wedge \neg(\neg p \wedge q)$ ------------------by the second De Morgan law

$= \neg p \wedge [\neg(\neg p) \vee \neg q]$ --------------------by the first De Morgan law

$= \neg p \wedge (p \vee \neg q)$ -----------------------by the double negation law

$= (\neg p \wedge p) \vee (\neg p \wedge \neg q)$ -----------by the second distributive law

$= F \vee (\neg p \wedge \neg q)$ ----------------------------------because $\neg p \wedge p \equiv F$

$= (\neg p \wedge \neg q) \vee F$--------- by the commutative law for disjunction

$= \neg p \wedge \neg q$ ---------------------------------------by the identity law for F

MATHEMATICAL FOUNDATION FOR COMPUTER SCIENCE

Prepared by: Er. Ankit Kharel

Nepal college of information technology

RULES OF INFERENCE

ARGUMENT:

- An argument is a sequence of proposition written as:

$$\begin{array}{c} P_1 \\ P_2 \\ . \\ . \\ . \\ . \\ . \\ . \\ P_n \\ \hline \therefore q_n \end{array} \qquad (P_1 \wedge P_2 \ldots\ldots\ldots P_n) \rightarrow Q \text{ is TAUTOLOGY}$$

- P_1, P_2 …..are called the Hypothesis or premises and the proposition Q is called Conclusion.
- The argument is valid provided that P_1, P_2 ……and P_n all are TRUE ,then Q also must be TRUE.
- This process of Drawing a conclusion from a sequence of proposition is called Deductive Reasoning.

RULES OF INFERENCE:

- If an argument consists of 10 different proposition variable then $2^{10}=1024$ combination are needed for Truth Table which is a tedious approach.
- Instead we can first establish the validity of some relatively simple arguments forms, called Rules of Inference.
- These rules then can be used to construct more complicated valid arguments.

1. MODUS PONENS:

- It states that if P and P→Q is TRUE then, we can infer Q is true.
- That is, $(P \wedge (P \to Q)) \to Q$ is TAUTOLOGY

$$\begin{array}{l} p \to q \\ p \\ \hline \therefore q \end{array}$$

Proof By Truth Table:

p	q	p→q	(p→q) ^p	((p→q) ^p)→q
T	T	T	T	T
T	F	F	F	T
F	T	T	F	T
F	F	T	F	T

2. MODUS TOLLENS:

- It states that if $P \rightarrow Q$ and $\neg Q$ is TRUE then, we can infer $\neg P$ is true.
- That is, $(\neg q \wedge (P \rightarrow Q)) \rightarrow \neg P$ is TAUTOLOGY

$$\begin{array}{l} p \rightarrow q \\ \neg q \\ \hline \therefore \neg p \end{array}$$

Proof By Truth Table:

p	q	p→q	¬p	¬q	(p→q) ^¬q	(p→q) ^¬q)→¬p
T	T	T	F	F	F	T
T	F	F	F	T	F	T
F	T	T	T	F	F	T
F	F	T	T	T	T	T

3. HYPOTHETICAL SYLLOGISM:

- It states that if P→Q and Q→R is TRUE then, we can infer P→R is true.
- That is, ((P→Q)^(Q→R))→(P→R) is TAUTOLOGY

$$\begin{array}{l} p \rightarrow q \\ q \rightarrow r \\ \hline \therefore p \rightarrow r \end{array}$$

P	q	r	p→q	q→r	p→r	(p→q)^(q→r)	((p→q)^(q→r))→(p→r)
T	T	T	T	T	T	T	T
T	T	F	T	F	F	F	T
T	F	T	F	T	T	F	T
T	F	F	F	T	F	F	T
F	T	T	T	T	T	T	T
F	T	F	T	F	T	F	T
F	F	T	T	T	T	T	T
F	F	F	T	T	T	T	T

4. DISJUNCTIVE SYLLOGISM:

- It states that if PvQ and ¬P is TRUE then, we can infer Q is true.
- That is, (¬P^(PvQ))→Q is TAUTOLOGY

$$\begin{array}{l} p \vee q \\ \neg p \\ \hline \therefore q \end{array}$$

Proof By Truth Table:

p	q	pVq	¬p	(pVq)^¬p	((pVq)^¬p)→q
T	T	T	F	F	T
T	F	T	F	F	T
F	T	T	T	T	T
F	F	F	T	F	T

5. ADDITION:

- It states that if P is TRUE then, PvQ will be TRUE.
- That is, P→(PvQ) is TAUTOLOGY

$$\frac{p}{\therefore\ p \vee q}$$

Proof By Truth Table:

p	q	pVq	P→(pVq)
T	T	T	T
T	F	T	T
F	T	T	T
F	F	F	T

6. SIMPLIFICATION:

- It states that if P^Q is TRUE then, P will be TRUE.
- That is, (P^Q)→P is TAUTOLOGY

$$\frac{p \wedge q}{\therefore p}$$

Proof By Truth Table:

p	q	p^q	(p^q)→p
T	T	T	T
T	F	F	T
F	T	F	T
F	F	F	T

7. CONJUNCTION:

- It states that if P is TRUE and Q is TRUE then, P^Q will be TRUE.

- That is, (P)^(Q)→(P^Q) is TAUTOLOGY

$$\begin{array}{l} p \\ q \\ \hline \therefore p \wedge q \end{array}$$

Proof By Truth Table:

p	q	p^q	(p^q)→(p^q)
T	T	T	T
T	F	F	T
F	T	F	T
F	F	F	T

8. RESOLUTION:

- It states that if (PvQ) and (¬PvR) is TRUE then, we can infer (QvR) is true.
- That is,((PvQ)^(¬PvR))→(QvR) is TAUTOLOGY

pVq
¬pVr
∴ qVr

p	q	r	¬p	pVq	¬pVr	qVr	((pVq)^(¬pVr))→(qVr)
T	T	T	F	T	T	T	T
T	T	F	F	T	F	F	T
T	F	T	F	T	T	T	T
T	F	F	F	T	F	F	T
F	T	T	T	T	T	T	T
F	T	F	T	T	T	T	T
F	F	T	T	F	T	T	T
F	F	F	T	F	T	T	T

RULES OF INFERENCE:

TABLE 1 Rules of Inference.

Rule of Inference	*Tautology*	*Name*
p $p \to q$ $\therefore q$	$(p \wedge (p \to q)) \to q$	Modus ponens
$\neg q$ $p \to q$ $\therefore \neg p$	$(\neg q \wedge (p \to q)) \to \neg p$	Modus tollens
$p \to q$ $q \to r$ $\therefore p \to r$	$((p \to q) \wedge (q \to r)) \to (p \to r)$	Hypothetical syllogism
$p \vee q$ $\neg p$ $\therefore q$	$((p \vee q) \wedge \neg p) \to q$	Disjunctive syllogism
p $\therefore p \vee q$	$p \to (p \vee q)$	Addition
$p \wedge q$ $\therefore p$	$(p \wedge q) \to p$	Simplification
p q $\therefore p \wedge q$	$((p) \wedge (q)) \to (p \wedge q)$	Conjunction
$p \vee q$ $\neg p \vee r$ $\therefore q \vee r$	$((p \vee q) \wedge (\neg p \vee r)) \to (q \vee r)$	Resolution

Q.1) State which rule of inference is the basis of the following argument:

"It is below freezing now. Therefore, it is either below freezing or raining now."

<u>Solution:</u>

Let, p :"*It is below freezing now*"

q :"*It is raining now.*"

Then this argument is of the form:

$$\frac{p}{\therefore\ p \lor q}$$

This is an argument that uses the **addition rule.**

Q.2) State which rule of inference is the basis of the following argument:

"It is below freezing and raining now. Therefore, it is below freezing now."

<u>Solution:</u>

let, p:"*It is below freezing now*"

q :"*It is raining now*"

This argument is of the form:

$$\frac{p \land q}{\therefore\ p}$$

This argument uses the **simplification rule.**

Q.3) State which rule of inference is the basis of the following argument:

"If you have a current password, then you can log onto the network. You have a current password. Therefore, You can log onto the network."

Solution:

Let, p :"*you have a current password*"

q :"*you can log onto the network*"

Then this argument is of the form:

$$\begin{array}{l} p \rightarrow q \\ \underline{p \qquad\quad} \\ \therefore q \end{array}$$

This is an argument that uses the **Modus Ponens rule.**

Q.4) State which rule of inference is the basis of the following argument:

"If it rains today, then we will not have a barbecue today. If we do not have a barbecue today, then we will have a barbecue tomorrow. Therefore, if it rains today, then we will have a barbecue tomorrow"

Solution:

let, p:"*it rains today*"

q :"*We will have a barbecue today*"

r :"*we will have a barbecue tomorrow*"

This argument is of the form:

$$\begin{array}{l} p \rightarrow \neg q \\ \underline{\neg q \rightarrow r \qquad} \\ \therefore p \rightarrow r \end{array}$$

This argument uses the **Hypothetical rule.**

Q.5) Show that the premises: *"If I play football then I am tired the next day", "I will take rest if I am tired", "I did not take rest"* will lead to the conclusion *"I did not play football".*

Solution:

Let, p: *"If I play football"*

q: *"I am tired"*

r: *"I will take rest "*

Hypothesis: i) $p \rightarrow q$

ii) $q \rightarrow r$

iii) $\neg r$

Conclusion: $\therefore \neg p$

s.n.	STEPS	REASONS
1.	$p \rightarrow q$	Given Hypothesis
2.	$q \rightarrow r$	Given Hypothesis
3.	$p \rightarrow r$	HYPOTHTICAL SYLLOGISM IN 1 & 2
4.	$\neg r$	Given Hypothesis
5.	$\neg p$	MODUS TOLLENSON 3 & 4

Q.6) Show that the premises *"It is not sunny this afternoon and it is colder than yesterday". "We will go swimming only if it is sunny." "If we do not go swimming, then we will take a canoe trip."* and *"If we take a canoe trip, then we will be home by sunset"* lead to the conclusion *"We will be home by sunset."*

Solution:

Let, p:"*It is sunny this afternoon*"

q:"*It is colder than yesterday*"

r:"*We will go swimming*"

s:"*We will take canoe trip*"

t:"*We will be home by sunset*"

Hypothesis: i) $\neg p \wedge q$

ii) $r \rightarrow p$

iii) $\neg r \rightarrow s$

iv) $s \rightarrow t$

Conclusion: $\therefore t$

s.n.	STEPS	REASONS
1.	$\neg p \wedge q$	Given Hypothesis
2.	$\neg p$	SIMPLIFICATION ON 1
3.	$r \rightarrow p$	Given Hypothesis
4.	$\neg r$	MODUS TOLLENS ON 2 & 3
5.	$\neg r \rightarrow s$	Given Hypothesis
6.	s	MODUS PONENS ON 4 & 5
7.	$s \rightarrow t$	Given Hypothesis
8.	t	MODUS PONENS ON 6 & 7

Q.7) Show that the premises *"If you send me an e-mail message, then I will finish writing the program", "If you do not send me an e-mail message, then I will go to sleep early," "If I go to sleep early, then I will wake up feeling refreshed"* lead to the conclusion *"If I do not finish writing the program, then I will wake up feeling refreshed."*

Solution:

Let, p: "*you send me an e-mail message*"

q: "*I will finish writing the program*"

r: "*I will go to sleep early*"

s: "*I will wake up feeling refreshed*"

Hypothesis: i) $p \to q$

ii) $\neg p \to r$

iii) $r \to s$

Conclusion: $\therefore \neg q \to s$

s.n.	STEPS	REASONS
1.	$p \to q$	Given Hypothesis
2.	$\neg q \to \neg p$	CONTRAPOSITIVE ON 1
3.	$\neg p \to r$	Given Hypothesis
4.	$\neg q \to r$	Hypothetical syllogism using (2) and (3)
5.	$r \to s$	Given Hypothesis
6.	$\neg q \to s$	Hypothetical syllogism using (4) and (5)

Q.8) Show that the premises *"If it does not rain or if it is not foggy, then the sailing race will be held and the lifesaving demonstration will go on," "If the sailing race is held, then the trophy will be awarded,"* and *"The trophy was not awarded"* imply the conclusion *"It rained."*

Solution:

Let, p: *"It rains"*
q: *"It is foggy"*
r: *"The sailing race is held"*
s: *"Life saving demonstration is done"*
t: *"Trophy is awarded"*

Hypothesis: i) $(\neg p \lor \neg q) \rightarrow (r \land s)$
ii) $r \rightarrow t$
iii) $\neg t$

Conclusion: $\therefore p$

s.n.	STEPS	REASONS
1.	$r \rightarrow t$	Given Hypothesis
2.	$\neg t$	Given Hypothesis
3.	$\neg r$	MODUS TOLLENS ON 1 & 2
4.	$(\neg p \lor \neg q) \rightarrow (r \land s)$	Given Hypothesis
5.	$\neg r \lor \neg s$	Addition on 3
6.	$\neg(r \land s)$	DE-MORGAN'S LAW on 5
7	$\neg(\neg p \lor \neg q)$	MODUS TOLLENS ON 4 and 6
8	$p \land q$	DE-MORGAN'S LAW on 7
9	p	SIMPLIFICATION on 8

Q.9) Show that the premises *"If the interest rate drops , the housing market will improve", "The federal discount rate will drop or the housing market will not improve", "Interest rate will drop"* imply the conclusion *"The federal discount rate will drop"*

Solution:

Let, p: *"the interest rate drops "*
q: *"the housing market will improve"*
r: *"The federal discount rate will drop"*

Hypothesis: i) $p \rightarrow q$
ii) $r \vee \neg q$
iii) p

Conclusion: $\therefore r$

STEPS	REASONS
1. $p \rightarrow q$	Given Hypothesis
2. $\neg p \vee q$	Implication on 1
3. $r \vee \neg q$	Given Hypothesis
4. $\neg p \vee r$	Resolution From 2 and 3
5. p	Given hypothesis
6. r	From 4 and 5

Q.10) Show that the premises *"If my cheque book is in office, then I have paid my phone bill", "I was looking for phone bill at breakfast or I was looking for phone bill in my office", "If was looking for phone bill at breakfast then my cheque book is on breakfast table" , "If I was looking for phone bill in my office then my cheque book is in my office", "I have not paid my phone bill"* imply the conclusion *"My cheque book is on my breakfast table"*

Solution:

Let, p: *"my cheque book is in office"*
q: *"I have paid my phone bill"*
r: *"I was looking for phone bill at breakfast"*
s: *"I was looking for phone bill in my office"*
t: *"my cheque book is on breakfast table"*

Hypothesis: i) $p \rightarrow q$
ii) $r \vee s$
iii) $r \rightarrow t$
iv) $s \rightarrow p$
v) $\neg q$

Conclusion: $\therefore t$

Hypothesis: i) $p \to q$
ii) $r \vee s$
iii) $r \to t$
iv) $s \to p$
v) $\neg q$
Conclusion: $\therefore t$

STEPS	REASONS
1. $p \to q$	Given Hypothesis
2. $\neg q$	Given Hypothesis
3. $\neg p$	Modus Tollens on 1 and 2
4. $r \vee s$	Given Hypothesis
5. $r \to t$	Given Hypothesis
6. $\neg r \vee t$	Implication on 5
7. $s \vee t$	Resolution from 4 and 6
8. $s \to p$	Given Hypothesis
9. $\neg s \vee p$	Implication on 8
10. $t \vee p$	Resolution from 7 and 9
11. t	From 3 and 10

PROOF BY RESOLUTION:

- Propositional Resolution works only on expressions in *clausal form*. Before the rule can be applied, the premises and conclusions must be converted to this form.

A *literal* is either an atomic sentence or a negation of an atomic sentence. For example, if p is a logical constant, the following sentences are both literals.

$$p$$
$$\neg p$$

A *clausal sentence* is either a literal or a disjunction of literals. If p and q are logical constants, then the following are clausal sentences.

$$p$$
$$\neg p$$
$$\neg p \vee q$$

A *clause* is the set of literals in a clausal sentence. For example, the following sets are the clauses corresponding to the clausal sentences above.

$$\{p\}$$
$$\{\neg p\}$$
$$\{\neg p, q\}$$

CONVERTING TO CAUSAL FORM:

- Examples:
  1. $p \rightarrow q \quad = \neg p \vee q$
  2. $p \leftrightarrow q \;= (\neg p \vee q) \wedge (\neg q \vee p)$-------{CNF}
  3. $\neg(p \wedge q) = \neg p \vee \neg q$
  4. $\neg(p \vee q) = \neg p \wedge \neg q$

CNF: CNF (Conjunctive normal form) if it is a ∧(Conjunction) of ∨(Disjunction s) of literals (variables or their negation.)

DNF: DNF (Disjunctive normal form) if it is a ∨(Disjunction s) of ∧(Conjunction) of literals (variables or their negation.)

Example:
1. $\neg p \vee \neg q$
2. $(p \wedge q) \vee (p \wedge r)$

Prove:

i) $P \rightarrow Q$

ii) $\neg P \rightarrow R$

iii) $R \rightarrow S$

$\therefore \neg Q \rightarrow S$

<u>Solution:</u>

The causal form of hypothesis and Conclusion are:

i) $\neg P \vee Q$

ii) $P \vee R$

iii) $\neg R \vee S$

$\therefore Q \vee S$

STEPS	REASONS
1. $\neg P \vee Q$	1. Given Hypothesis
2. $P \vee R$	2. Given Hypothesis
3. $Q \vee R = R \vee Q$	3. Using Resolution
4. $\neg R \vee S$	4. Given Hypothesis
5. $Q \vee S$	4. Using Resolution

Using Graphical Method:

Prove:

i) P→Q

ii) ¬P→R

iii) R→S

∴ ¬Q→S

STEP 1. The causal form of hypothesis and Conclusion are:

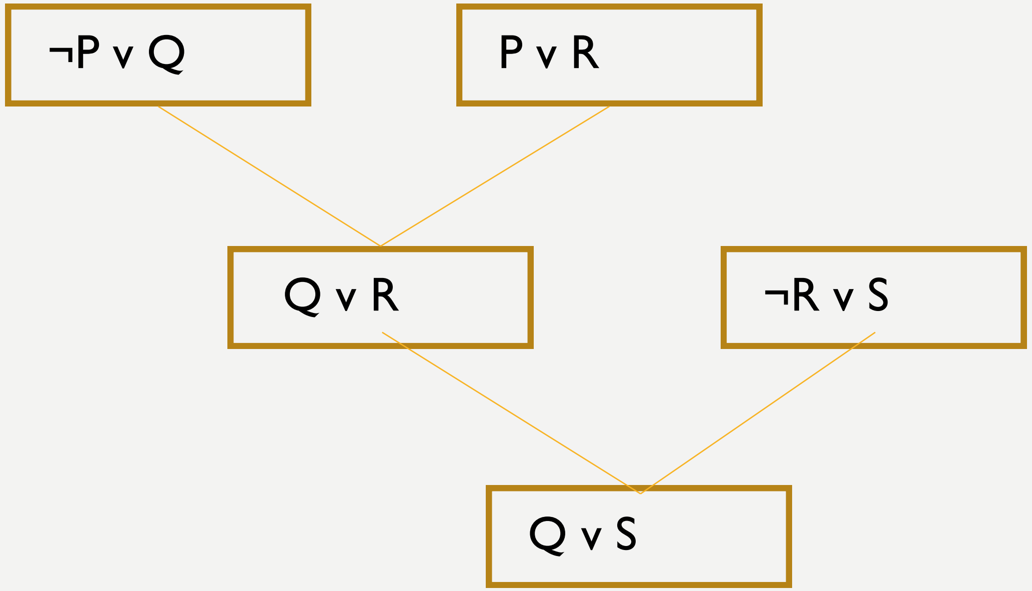
i) ¬P v Q

ii) P v R

iii) ¬R v S

∴ Q v S

STEP 2.



Using Graphical Method:

Prove:

i) P

ii) P→R

iii) R→S

∴ S

STEP 1. The causal form of hypothesis and Conclusion are:

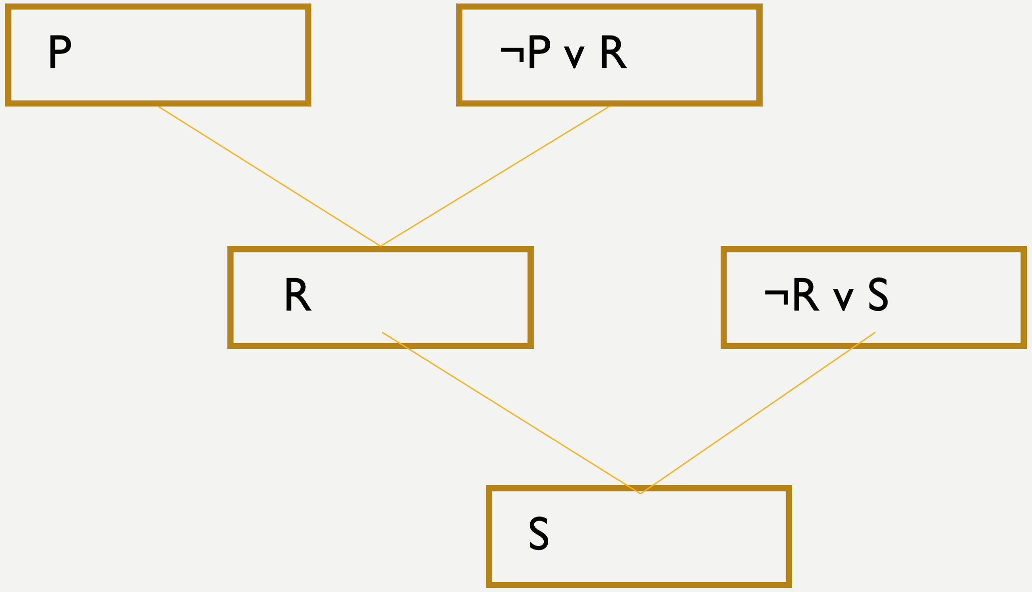
i) P

ii) ¬P v R

iii) ¬R v S

∴ S

STEP 2.



- Using resolution principle, prove that the hypotheses: "If today is Tuesday then I will have a test in Discreet Math or Microprocessor". "If my Microprocessor teacher is sick then I will not have a test in Microprocessor." "Today is Tuesday and my Microprocessor teacher is sick." lead to the conclusion that "I will have a test in Discrete Math"

Solution:

Let, p : "Today is Tuesday"

q : "I will have test in Discrete Math"

r : "I will have test in Microprocessor"

s : "My Microprocessor teacher is sick"

Hypothesis: i) $p \rightarrow (q \vee r)$

ii) $s \rightarrow \neg r$

iii) $p \wedge s$

Conclusion: q

The Causal Forms are:

Hypothesis:

i) $\neg p \vee (q \vee r)$

ii) $\neg s \vee \neg r$

iii) p

iv) s

Conclusion:

$\therefore$ q

The Causal Forms are:

Hypothesis:

i) $\neg p \vee (q \vee r)$

ii) $\neg s \vee \neg r$

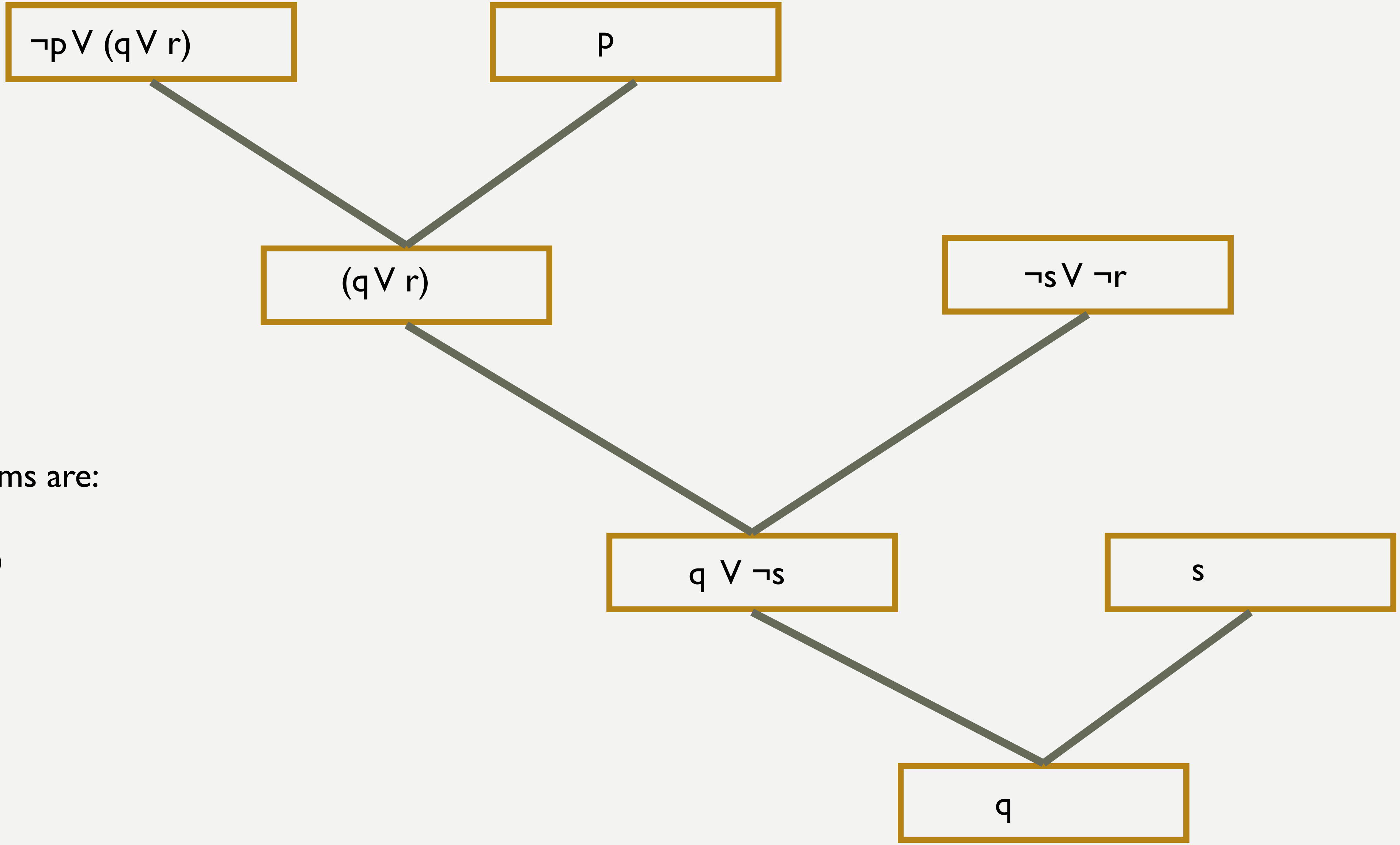
iii) p

iv) s

Conclusion:

$\therefore q$

STEPS	REASONS
1. $\neg p \vee (q \vee r)$	Given Hypothesis
2. p	Given Hypothesis
3. $q \vee r$	From 1 and 2
4. $\neg s \vee \neg r$	Given Hypothesis
5. $q \vee \neg s$	From 3 and 4
6. s	Given Hypothesis
7. q	From 5 and 6



The Causal Forms are:

Hypothesis:

i) ¬p V (q V r)
ii) ¬s V ¬r
iii) p
iv) s

Conclusion:

∴ q

SEMANTIC TABLEAUX:

- The formula is decomposed into its sub-formulas according to certain rules(α and β Rules), resulting a semantic tableau.
- Semantic Tableau is a binary tree constructed using semantic rules.

α Rule

a) $P \vee Q$	b) $P \wedge Q$	c) $P \rightarrow Q$	d) $P \leftrightarrow Q$
P Q	P Q	$\neg P$ Q	P $\neg P$ Q $\neg Q$

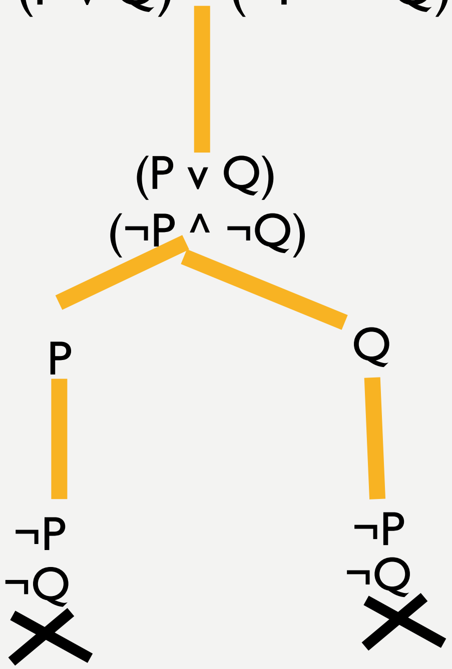
β Rule

a) $\neg(\neg P)$	b) $\neg(P \vee Q)$	c) $\neg(P \wedge Q)$	d) $\neg(P \rightarrow Q)$	e) $\neg(P \leftrightarrow Q)$
P	$\neg P$ $\neg Q$	$\neg P$ $\neg Q$	P $\neg Q$	P $\neg P$ $\neg Q$ Q

SEMANTIC TABLEAUX:

- A finite set of formulas φ is satisfiable iff $T(\varphi)$ is open.
- As a corollary, φ is contradictory (not satisfiable) iff $T(\varphi)$ is closed

$\varphi = (P \vee Q) \wedge (\neg P \wedge \neg Q)$

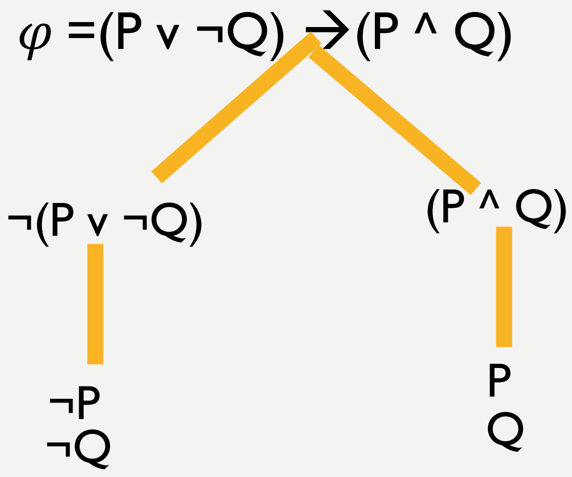


$A = (P \vee Q)$
$B = (\neg P \wedge \neg Q)$

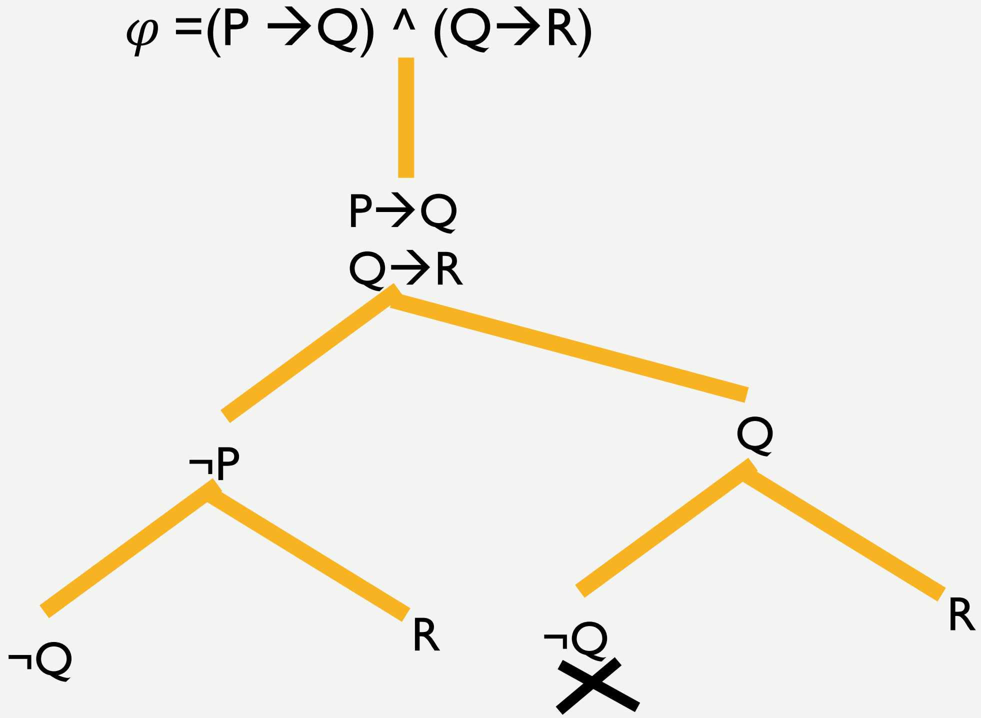
P	Q	¬P	¬Q	A	B	A^B
T	T	F	F	T	F	**F**
T	F	F	T	T	F	**F**
F	T	T	F	T	F	**F**
F	F	T	T	F	T	**F**

- When P and its negation ¬P appear on the same branch, a contradiction has been found and that branch is called closed.

SEMANTIC TABLEAUX:



SATISFIABLE
(There are open branches)

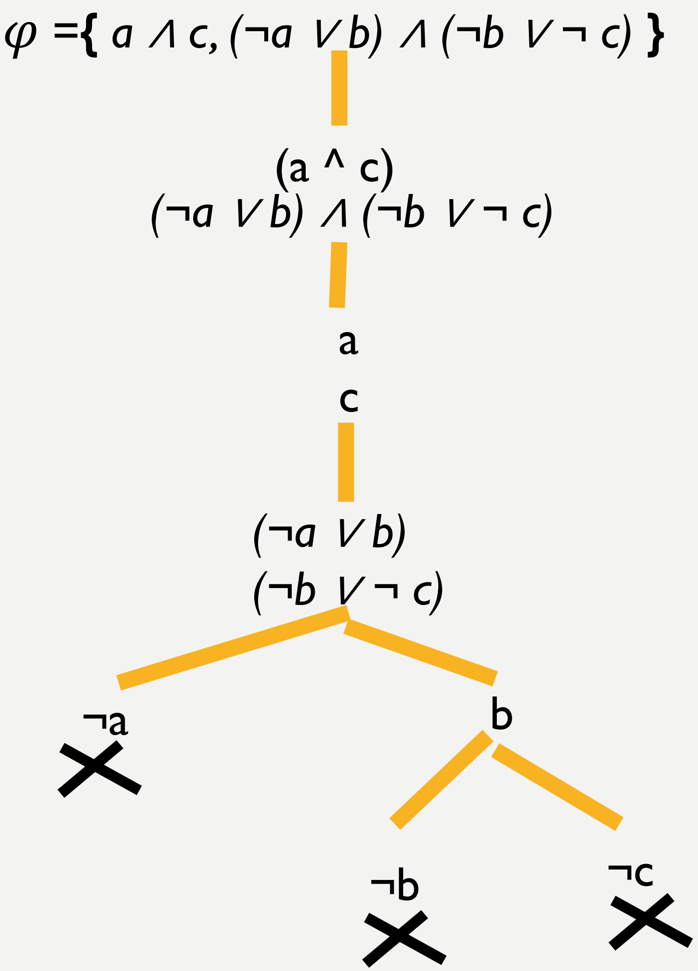


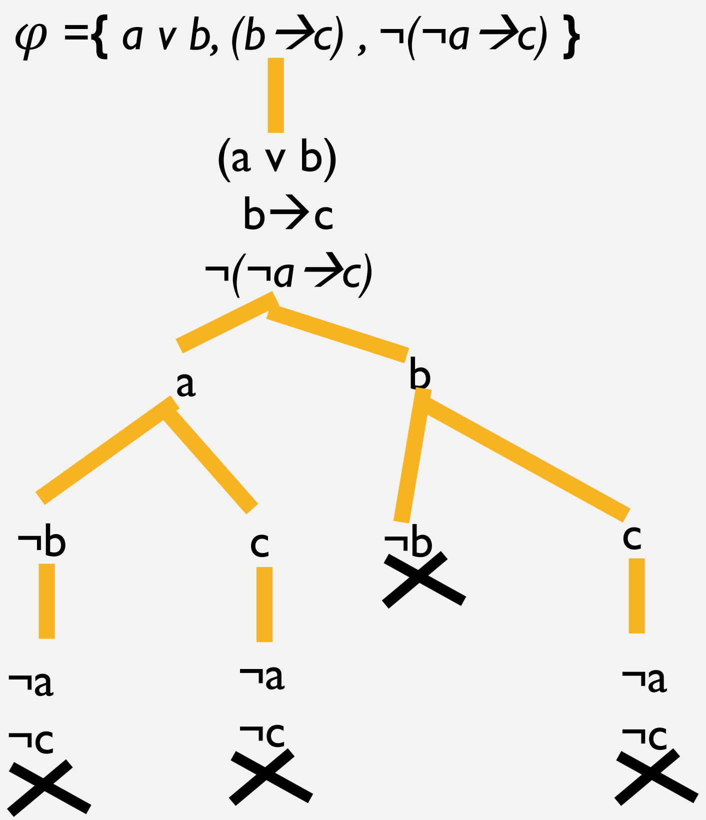
SATISFIABLE
(There are open branches)

P	Q	R	(P→Q)	(Q→R)	(P→Q)^(Q→R)
T	T	T	T	T	T
T	T	F	T	F	F
T	F	T	F	T	F
T	F	F	F	T	F
F	T	T	T	T	T
F	T	F	T	F	F
F	F	T	T	T	T
F	F	F	T	T	T

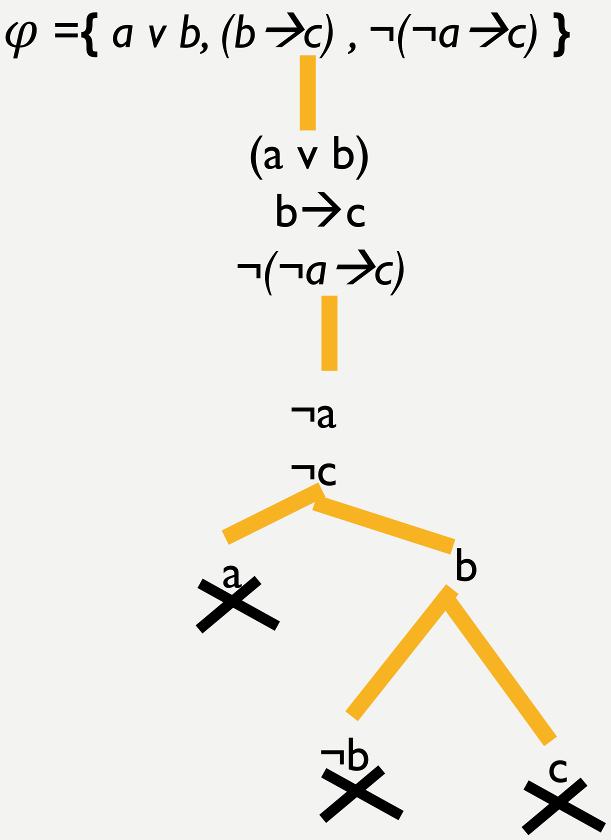
P	Q	¬Q	A=(PV¬Q)	B=(P^Q)	A→B
T	T	F	T	T	T
T	F	T	T	F	F
F	T	F	F	F	T
F	F	T	T	F	F

SEMANTIC TABLEAUX:





[Disjunction First Policy]



[Conjunction First Policy]

SEMANTIC TABLEAUX:

To prove the argument:

$$\begin{array}{c} P_1 \\ P_2 \\ P_3 \\ . \\ . \\ P_n \\ \hline \therefore Q \end{array}$$

$[(P_1 \wedge P_2 , \ldots\ldots. \wedge P_n) \rightarrow Q]$ is Tautology.

- To prove the above argument we show that the set of premises along with negated conclusion is Unstisfiable i.e. we show semantic Tableau is Contradiction.

$$= \ [(P_1 \wedge P_2 , \ldots\ldots. \wedge P_n) \rightarrow Q]$$
$$= \neg[(P_1 \wedge P_2 , \ldots\ldots. \wedge P_n) \rightarrow Q]$$
$$= \ [P_1 \wedge P_2 , \ldots\ldots.. \wedge \ P_n \wedge \neg Q]$$

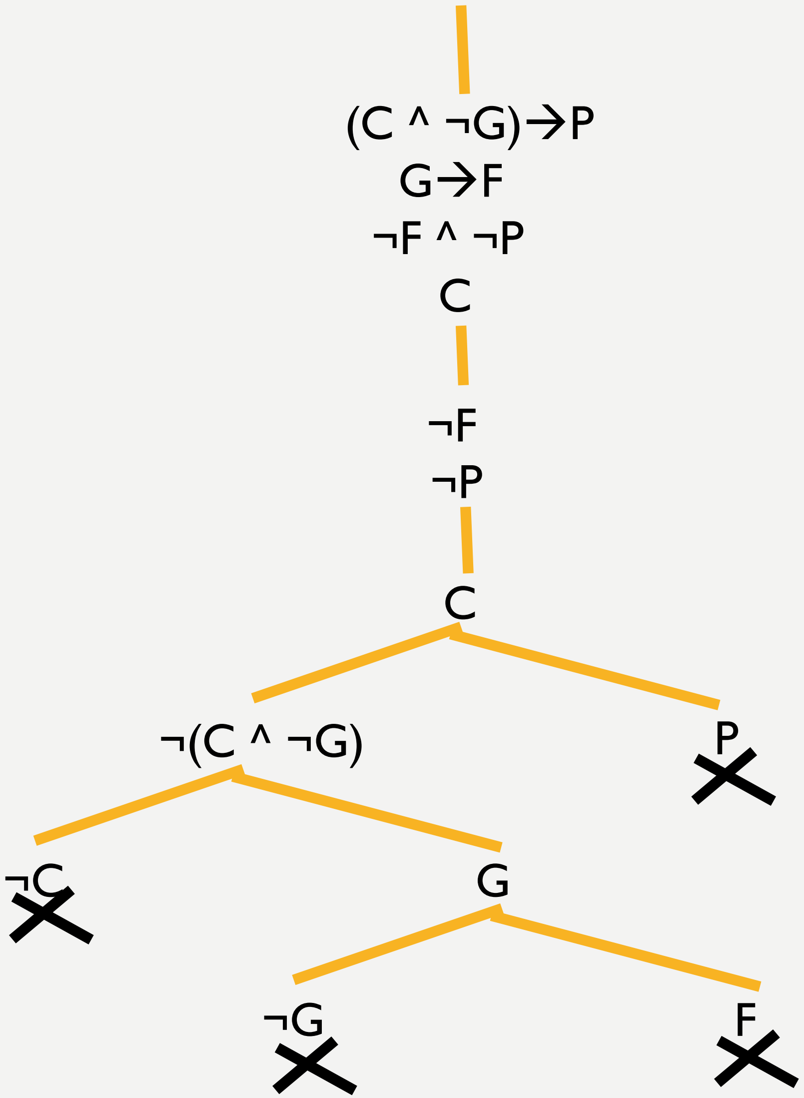
- $\varphi=\{P_1 , P_2 , P_3 , \ldots\ldots \ P_n , \neg Q \}$
- *If* $T(\varphi)$ is closed then the argument is valid else if $T(\varphi)$ is open the argument is invalid

Check if the following argument is valid or not.

$(C \wedge \neg G) \rightarrow P$
$G \rightarrow F$
$\neg F \wedge \neg P$
$\therefore \neg C$

$\varphi = \{(C \wedge \neg G) \rightarrow P\ ,\ G \rightarrow F\ ,\ \neg F \wedge \neg P\ ,\ \neg(\neg C)\ \}$

$\varphi = \{(C \wedge \neg G) \rightarrow P\ ,\ G \rightarrow F\ ,\ \neg F \wedge \neg P\ ,\ \neg(\neg C)\ \}$



The obtained Tableau is contradictory. Hence the argument is valid

i) $p \rightarrow q$
ii) $q \rightarrow r$
iii) $\neg r$
$\therefore \neg p$

Hypothesis: i) $(\neg p \vee \neg q) \rightarrow (r \wedge s)$
ii) $r \rightarrow t$
iii) $\neg t$
Conclusion: $\therefore p$

Hypothesis: i) $\neg p \wedge q$
ii) $r \rightarrow p$
iii) $\neg r \rightarrow s$
iv) $s \rightarrow t$
Conclusion: $\therefore t$

Hypothesis: i) $p \rightarrow q$
ii) $\neg p \rightarrow r$
iii) $r \rightarrow s$
Conclusion: $\therefore \neg q \rightarrow s$

MATHEMATICAL FOUNDATION FOR COMPUTER SCIENCE

Prepared by: Er. Ankit Kharel

Nepal college of information technology

PREDICATE LOGIC

1.QUANTFIERS

2.TYPES OF QUANTIFIERS

3.NEGATING QUANTIFIERS

4.TRANSLATION FROM ENGLISH

LIMITATION OF PROPOSITIONAL LOGIC:

Consider the following :

p: "All men are mortal"

q: "Ram is man"

---

r: ∴ "Ram is mortal"

- No rule of propositional logic will allow us to conclude the truth of 'r'.
- Therefore, We need more powerful type of logic called First order Logic or PREDICATE LOGIC.

- To understand predicate Logic we need to understand:
  a) Subject
  b) Predicates
  c) Quantifiers
  d) Domain(Universe of Discourse0

Consider an statement:

“Jose is Tall”

Subject

Tall(Jose)

Predicate

- Subject: “The **subject** is what (or whom) the sentence is about”

- Predicate: “**Predicate** refers to a property that the subject of a statement can have”

Consider an statement:

"X is greater than 3"   ($x>3$)

Subject: Variable "x"

Predicate: Greater than 3

We can denote "$x>3$" as: $P(x)$

The statement $P(x)$ becomes a proposition once the value has been assigned to the subject.

<u>Example:</u>

$P(5)$: "5 is greater than 3" (TRUE)

$P(2)$: "2 is greater than 3" (FALSE)

Q.1) Let Q(x) denotes the statement : "The word "x" contains the letter 'a' "
What are the Truth value of Q(ankit), Q(Logic), Q(nothing) ?

Solution

Q(x): " The word "x" contains the letter 'a' "

Q(ankit): ""The word "ankit" contains the letter 'a' " (TRUE)
Q(Logic): ""The word "Logic" contains the letter 'a' " (FALSE)
Q(nothing): ""The word "nothing" contains the letter 'a' " (FALSE)

Q.2) Let C(x, y) denotes the statement: "x is the capital of y"
What are the truth value of C(Kathmandu, Nepal), C(Texas, America) ?

Solution

C(x, y): "x is the capital of y"

C(Kathmandu, Nepal) : "Kathmandu is capital of Nepal" (TRUE)
C(Texas, America) : "Texas is capital of America" (FALSE)

Consider The Following:

P(x): "x is greater than 10"

Domain: All positive natural numbers.

Can we say the above statement is true for all values of x?

=No, because for x=1,2,3,4,5,6,7,8,9,10 above statement becomes FALSE.

So, We can say above statement as : **For some x, P(x) is TRUE.**

Consider The following:

Q(x): "x<x+1"

Domain : All positive natural numbers.

Can we say that Q(x) is TRUE For all values of x within our domain?

=Yes

So, we can say above statement as: **For all x, Q(x) is TRUE.**

QUANTIFIERS

In **predicate logic**, **predicates** are used alongside **quantifiers** to express the extent to which a **predicate** is true over a range of elements. Using **quantifiers** to create such propositions is called quantification.

1. UNIVERSAL QUANTIFICATION(∀):

"Every cat drinks milk"

CATS

X_1
X_2
X_3

Domain(Universe of Discourse)

Above statement is Equivalent to:

X_1 Drinks milk.
$\wedge$
X_2 Drinks milk.
$\wedge$
X_3 Drinks milk.

$\mathrm{Milk}(X_1)$
$\wedge$
$\mathrm{Milk}(X_2)$
$\wedge$
$\mathrm{Milk}(X_3)$

$$\forall_x \mathrm{Milk}(X)$$

=For all X, Milk(X)

or

=For ever X, Milk(X)

1. UNIVERSAL QUANTIFICATION(∀):

The Universal Quantification of P(x) is:

"P(x) for all value of x in the Domain"

$$= \forall_x P(x)$$

We can also read $\forall_x P(x)$ as:

"For all P(x) " or "for every x, P(x)"

<u>Example:</u>

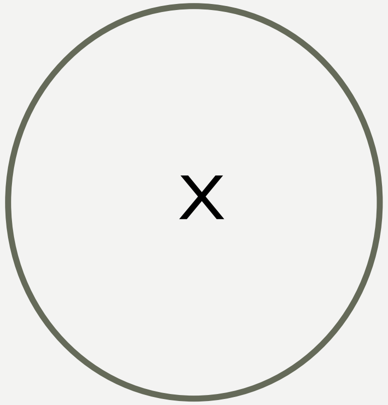
Q.1) ***"All student of BEIT takes course on Discrete Mathematics"***

let,

D(x):"x takes course on Discrete Mathematics"

$$= \forall_x D(x)$$

<u>BEIT</u>



<u>Domain(Universe of Discourse)</u>

1. UNIVERSAL QUANTIFICATION(∀):

Example:

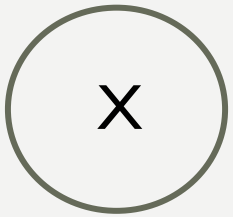
Q.2) ***"Every men are Mortal"***

let,

$M(x)$:"x is mortal"

$=\forall_x M(x)$

MEN



Domain(Universe of Discourse)

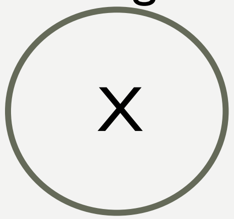
Q.3) "x+1>x"

let,

$P(x)$:"x+1>x"

$=\forall_x P(x)$

+ve integers



Domain(Universe of Discourse)

1. UNIVERSAL QUANTIFICATION($\forall$):

Q.4) Let Q(x) be the statement "x<5". What is the truth value of Quantification, $\forall_x Q(X)$, where domain of discourse is all real numbers.

Solution

Q(x) is not True for every real number.

for instance,

Q(6)="6<5" is FALSE.

Thus , $\forall_x Q(x)$ is FALSE.

COUNTER EXAMPLE: An Element for which P(x) is False is called Counter Example of $\forall_x P(x)$.

Q.5) What is the truth value of $\forall_x P(x)$, where P(x) is the statement "$X^2 < 10$" and the domain consists of the positive integers not exceeding 4.

Solution: The statement $\forall_x P(x)$ is the same as the conjunction

$P(1) \land P(2) \land P(3) \land P(4)$, because the domain consists of the integers 1, 2, 3, and 4.

Because P (4), which is the statement "42 < 10," is false, it follows that $\forall_x P(x)$ is false.

2. EXISTENTIAN QUANTIFICATION(∃):

"some lion drinks milk"

LION

X_1
X_2
X_3

Domain(Universe of Discourse)

Above statement is Equivalent to:

X_1 Drinks milk.
V
X_2 Drinks milk.
V
X_3 Drinks milk.

Milk(X_1)
V
Milk(X_2)
V
Milk(X_3)

$$\exists_x \text{Milk}(X)$$

=There exist an x in the domain such that Milk(X)

=There is at least one x such that Milk(x)

=for some x, Milk(x)

2. EXISTENTIAN QUANTIFICATION(∃):

Example:

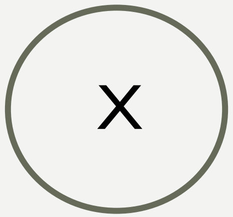
Q.1) *"Some people are Bad"*

let,

B(x):"x is Bad"

$=\exists_x B(x)$

People



Domain(Universe of Discourse)

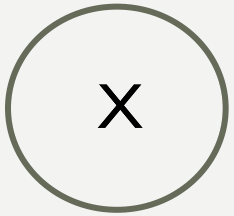
Q.2) *"Some dogs are big"*

let,

D(x):"x is Big"

$=\exists_x D(x)$

DOGS



Domain(Universe of Discourse)

2. EXISTENTIAN QUANTIFICATION(∃):

Q.1) Let P (x) denote the statement "$x > 3$." What is the truth value of the quantification $\exists_x P(x)$, where the domain consists of all real numbers.

Solution:

Because "$x > 3$" is sometimes true—for instance, when $x = 4$—the existential quantification of P (x), which is $\exists_x P(x)$, is true.

Q.2) What is the truth value of $\exists_x P(x)$, where P(x) is the statement "$x^2 > 10$" and the universe of discourse consists of the positive integers not exceeding 4?

Solution:

Because the domain is {1, 2, 3, 4}, the proposition $\exists_x P(x)$ is the same as the disjunction $P(1) \vee P(2) \vee P(3) \vee P(4)$. Because P (4), which is the statement "42 > 10," is true, it follows that $\exists_x P(x)$ is true.

Statement	When True?	When False?
$\forall_x P(x)$	P(x) is true for every x.	There is an x for which P(x) is false
$\exists_x P(x)$	There is an x for which P(x) is true	P(x) is false for every x.

FREE & BOUND VARIABLES:

- When the variable is assigned a value or it is quantified it is called bound variable. If the variable is not bounded then it is called free variable.
- An occurrence of a variable that is not bound by a quantifier or set equal to a particular value is said to be free.
- Example:
  1. P(x, y) has two free variables x and y.
  2. P(2, y) has one bound variable 2 and one free variable y.
  3. ∀x P(x) has a bound variable x.
  4. ∀x P(x, y) has one bound variable x and one free variable y.
- Expression with no free variable is a proposition.
- Expression with at least one free variable is a predicate only.

NEGATING QUANTIFICATIONS:

1. Negating Universal Quantification:

   $\neg[\forall_x P(x)]$

   a) Negate the Proposition Function$[\neg P(x)]$

   b) Change to Existential Quantification

$$\neg[\forall_x P(x)] = \exists_x \neg[P(x)]$$

2. Negating Existential Quantification:

   $\neg[\exists_x P(x)]$

   a) Negate the Proposition Function$[\neg P(x)]$

   b) Change to Universal Quantification

$$\neg[\exists_x P(x)] = \forall_x[\neg P(x)]$$

De-Morgan's Law For Quantifiers

Negate The Following :

1. ***"Every student in BEIT has Taken Data mining***" [Domain: All BEIT student]

   Solution

   let, p(x): "x has taken Data Mining"

   $=\forall_x P(x)$

   Negation:

   $= \boldsymbol{\neg[\forall_x P(x)]}$

   $\boldsymbol{=\exists_x \neg[P(x)]}$

   "There is a student in BEIT who has not taken Data Mining"

2. "***There is a student in class who has long hair"*** [Domain: All BEIT student]

   Solution

   let, p(x): "x has long hair"

   $= \exists_{\mathbf{x}} P(x)$

   Negation:

   $= \boldsymbol{\neg[\exists_x} P(x)\boldsymbol{]}$

   $\boldsymbol{=\forall_x \neg[P(x)]}$

   "All student in the class do not have long hair"

3. What are the negations of the statements:

a) $\forall_x(x^2 > x)$

Solution:

The negation of $\forall_x(x^2 > x)$ is,

$\neg\forall_x(x^2 > x)$, which is equivalent to

$\exists_x\neg(x^2 > x)$.

This can be rewritten as $\exists_x(x^2 \leq x)$.

b) $\exists_x(x^2 = 2)$

Solution:

The negation of $\exists_x(x^2 = 2)$is,

$\neg\exists_x(x^2 = 2)$, which is equivalent to

$\forall_x\neg(x^2 = 2)$.

This can be rewritten as $\forall_x(x^2 != 2)$.

TRANSLATING FROM ENGLISH:

1. Express the statement *"Every student in BEIT class has studied calculus"* using predicates and quantifiers.

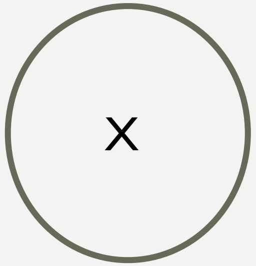
Solution:

First, we introduce a variable x so that our statement becomes *"For every student x in BEIT, x has studied calculus."*

Now, let C(x): *"x has studied calculus."*

Domain: BEIT

$$=\forall_x C(x)$$



DOMAIN: BEIT students

Domain: All people

Our statement becomes:

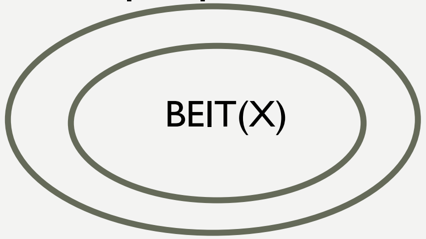
"For every person x, if person x is a student in BEIT, then x has studied calculus."

Now, let S(x) : *"x is a student in BEIT."*

C(x) : *"x has studied calculus"*

$$=\forall_x [S(x) \rightarrow C(x)]$$

all people



TRANSLATING FROM ENGLISH:

Domain: All people

Our statement becomes:

"For every person x, if person x is a student in BEIT, then x has studied calculus."

Now, let S(x) :*"x is a student in BEIT."*

C(x) :*"x has studied calculus"*

$= \forall_x [S(x) \rightarrow C(x)]$

Q(x, Calculus) :*"Student x has studied Calculus"*

$= \forall_x [S(x) \rightarrow Q(x, Calculus)]$

[Caution! Our statement cannot be expressed as $\forall_x [S(x) \wedge C(x)]$ because this statement says that all people are students in this class and have studied calculus!]

TRANSLATING FROM ENGLISH:

2. Express the statement *"Some student in this class has visited Jhapa"* using predicates and quantifiers.

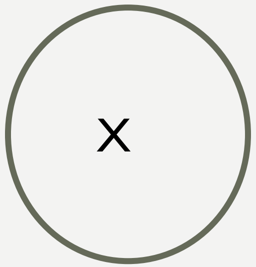
Solution:

First, we introduce a variable x so that our statement becomes *"There is a student x in this class that has visited Jhapa"*

Now, let p(x): *"x has visited Jhapa"*

Domain: BEIT

$= \exists_x p(x)$



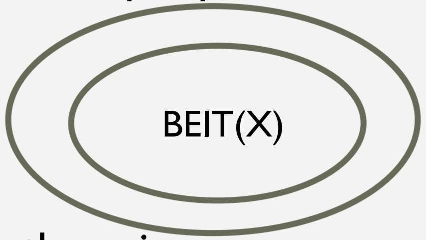
DOMAIN: BEIT students

Domain: All people

Our statement becomes:

"There is a person x, if person x is in this class, then x has studied calculus."

all people



Now, let S(x) : *"x is a student in BEIT."*

p(x) : *"x has visited Jhapa"*

$= \exists_x [S(x) \wedge p(x)]$

Caution! Our statement cannot be expressed as $\exists_x(S(x) \rightarrow M(x))$, which is true when there is someone not in the class because, in that case, for such a person x, $S(x) \rightarrow M(x)$ becomes either $F \rightarrow T$ or $F \rightarrow F$, both of which are true.

TRANSLATING FROM ENGLISH:

3. Express the statement *"Every Student in this class has visited Jhapa or Kathmandu"* using predicates and quantifiers.

<u>Solution</u>:

Now, let k(x): "*x has visited Kathmandu*"

J(x): "*x has visited Jhapa*"

Domain: BEIT

$= \forall_x [k(x) \vee j(x)]$

Domain: All people

Our statement becomes:

"for all person x, if person x is in this class, then x has visited Jhapa or Kathmandu."

Now, let S(x) : "*x is a student in BEIT.*"

k(x): "*x has visited Kathmandu*"

J(x): "*x has visited Jhapa*"

$= \forall_x [S(x) \rightarrow (k(x) \vee j(x))]$

TRANSLATING FROM ENGLISH:

Consider these statement:

- *No professor are ignorant*
- *All ignorant people are vain*
- *Some professor are ignorant*

Let , P(x): *x is a Professor* , I(x): *x is ignorant* ,V(x): *x is vain*

Express above statement using quantifiers where domain consist of all people

a) No professor are ignorant

$$\forall_x[p(x) \rightarrow \neg q(x)]$$

b) All ignorant people are vain

$$\forall_x[q(x) \rightarrow r(x)]$$

c) Some professor are ignorant

$$\exists_x[p(x) \wedge q(x)]$$

TRANSLATING FROM ENGLISH:

Q. Let P (x) be the statement "x can speak Russian"

Q(x) be the statement "x knows the computer language C++."

Express each of these sentences in terms of P (x), Q(x), quantifiers, and logical connectives. The domain for quantifiers consists of all students at your school.

a) There is a student at your school who can speak Russian and who knows C++.
   $=\exists_x (P(x) \land Q(x))$.

a) There is a student at your school who can speak Russian but who doesn't know C++.
   $= \exists_x (P(x) \land \neg Q(x))$

a) Every student at your school either can speak Russian or knows C++.
   $= \forall_x (P(x) \lor Q(x))$

a) No student at your school can speak Russian or knows C++.
   $= \forall_x$ **$[\neg P(x) \land \neg Q(x)]$**

TRANSLATING FROM ENGLISH:

1. No one is sleeping.

   Negation of above: There is some who is sleeping

   $=\exists_x[\text{Person}(x) \wedge \text{sleeping}(x)]$

   Now , negate the predicate:

   $=\neg\exists_x[\text{Person}(x) \wedge \text{sleeping}(x)]$

2. Not everyone is sleeping.

   Negation of above: Everyone is sleeping.

   $= \forall_x [\text{Person}(x) \rightarrow \text{Sleeping}(x)]$

   Now , negate the predicate:

   $=\neg\forall_x [\text{Person}(x) \rightarrow \text{Sleeping}(x)]$

3. No one in this class is wearing glass and a cap.

   Negation of above statement: There is some one in this class who is wearing glass and a cap.

   $=\exists_x[\text{Glass}(x) \wedge \text{Cap}(x)]$

   Now, negate the predicate:

   $=\neg\exists_x[\text{Glass}(x) \wedge \text{Cap}(x)]$

MATHEMATICAL FOUNDATION FOR COMPUTER SCIENCE

Prepared by: Er. Ankit Kharel

Nepal college of information technology

PREDICATE LOGIC

2.NESTED QUANTIFIERS

3.TRANSLATION FROM ENGLISH

1. NESTED QUANTIFIERS:

1. Consider the following :

"The sum of any two positive real number is positive"

This assertion can be restated as:

"for ever x and for every y , If x>0 and y>0, then x+y>0"

Let,

p(x, y) :"$(x>0)\wedge(y>0) \rightarrow (x+y)>0$

The given statement says that the sum of any two positive real number is positive, so we need two Universal quantifiers.

The Quantification is:

$$\boxed{\forall_x \forall_y [p(x , y)]}$$

1. NESTED QUANTIFIERS:

3. Consider the following :

"Some student in your class has taken some computer training course"

Restating above statement as:

"For some student x ,there exist a computer training course y such that x has taken y"

let, Q(x, y) :"Student x has taken training y"

The Quantification is:

$$\boxed{\exists_x \exists_y [Q(x, y)]}$$

1. NESTED QUANTIFIERS:

4. Consider the following :

"If a person is female and is a parent , then this person is someone's mother"

Restating above statement as:

"For every person x, if x is a female and person x is a parent , then there exist a person y such that person x is the mother of y.

let, $F(x)$:"x is female"

$P(x)$:" x is parent"

$M(x, y)$:"x is the mother of y"

The Quantification is:

$$= \forall_x[(F(x) \wedge P(x)) \rightarrow \exists_y M(x, y)]$$
$$= \forall_x \exists_y [(F(x) \wedge P(x)) \rightarrow M(x, y)]$$

1. NESTED QUANTIFIERS:

5. Consider the following :

"There is a man that has taken a flight on every airline in the world"

Restating above statement as:

"There is a man x, for all airlines a ,there exist a flight f such that x has taken flight f"

let, P(x, f) :"x has taken flight f"

Q(f, a) :" f is a flight on airline a"

The Quantification is:

$$\boxed{= \exists_x \forall_a \exists_f [(P(x, f) \wedge Q(f, a)]}$$

1. NESTED QUANTIFIERS:

Let,
L(x, y): "x loves y"

a) "Everyone Loves Somebody"
=For every person x, there exist person y such that x loves y
$=\forall_x \exists_y L(x, y)$

b) "Someone Loves Somebody"
=There exist some person x and some person y such that x loves y
$=\exists_x \exists_y L(x, y)$

c) "Someone is loved by everyone"
=There exist some person y for all x such that x loves y.
$=\exists_y \forall_x L(x, y)$

d) "Everybody Loves Everybody"
$= \forall_x \forall_y L(x, y)$

Q. Let $Q(x, y)$ denote "$x + y = 0$."
What are the truth values of the quantifications
a) $\exists_y\forall_x Q(x, y)$ and b) $\forall_x\exists_y Q(x, y)$, where the domain for all variables consists of all real numbers?

Solution:

a) The quantification $\exists_y\forall_x Q(x, y)$ denotes the proposition "There is a real number y such that for every real number x, $Q(x, y)$." No matter what value of y is chosen, there is only one value of x for which $x + y = 0$. Because there is no real number y such that $x + y = 0$ for all real numbers x, the statement $\exists_y\forall_x Q(x, y)$ is false.

b) The quantification $\forall_x\exists_y Q(x, y)$ denotes the proposition "For every real number x there is a real number y such that $Q(x, y)$." Given a real number x, there is a real number y such that $x + y = 0$; namely, $y = -x$. Hence, the statement $\forall_x\exists_y Q(x, y)$ is true.

Q. Let $P(x, y)$ be the statement "$x + y = y + x$."
What are the truth values of the quantifications a) $\forall_x\forall_y P(x, y)$ and b) $\forall_y\forall_x P(x, y)$ where the domain for all variables consists of all real numbers?

Solution:

a) The quantification $\forall_x\forall_y P(x, y)$ denotes the proposition "For all real numbers x, for all real numbers y, $x + y = y + x$." Because $P(x, y)$ is true for all real numbers x and y the proposition $\forall_x\forall_y P(x, y)$ is true.

b) The quantification $\forall_y\forall_x P(x, y)$ says "For all real numbers y, for all real numbers x, $x + y = y + x$." Because $P(x, y)$ is true for all real numbers x and y the proposition $\forall_y\forall_x P(x, y)$ is true.

Q. Let $Q(x, y)$ denote "$x + y = xy$."
What are the truth values of the quantifications
a) $\exists_x\exists_y Q(x, y)$ and b) $\exists_y\exists_x Q(x, y)$, domain for all variables consists of all positive real numbers?

Solution:

a) The quantification $\exists_x\exists_y Q(x, y)$ denotes the proposition "There is exist a number x such that for some number y, $Q(x, y)$." $Q(x, y)$ is true for $x=(0,2)$ and $y= (0, 2)$. Hence, $\exists_x\exists_y Q(x, y)$ is TRUE

b) The quantification $\exists_y\exists_x Q(x, y)$ denotes the proposition "There is exist a number y such that for some number x, $Q(x, y)$." $Q(x, y)$ is true for $y=(0,2)$ and $x= (0, 2)$. Hence, $\exists_y\exists_x Q(x, y)$ is TRUE

$\exists_y\forall_x Q(x, y) \;!\!= \forall_x\exists_y Q(x, y)$
$\forall_x\forall_y P(x, y) = \forall_y\forall_x P(x, y)$
$\exists_x\exists_y P(x, y) = \exists_y\exists_x P(x, y)$

2. NEGATING NESTED QUANTIFIERS:

- Statements involving nested quantifiers can be negated by successively applying the De-Morgan's rules for negating statements involving a single quantifier.

a) Express the negation of the statement $\forall_x \exists_y (xy = 1)$.

$=\neg[\forall_x \exists_y (xy = 1)]$

$=\exists_x \neg[\exists y(xy = 1)]$

$=\exists_x \forall_y \neg(xy = 1)$

$= \exists_x \forall_y (xy != 1)$

b) $\exists_x \exists_y P(x, y) \wedge \forall_x \forall_y Q(x, y)$

$= \neg[\exists_x \exists_y P(x, y) \wedge \forall_x \forall_y Q(x, y)]$

$=\neg[\exists_x \exists_y P(x, y)] \vee \neg[\forall_x \forall_y Q(x, y)]$

$= \forall_x \neg\exists_y P(x, y) \vee \exists_x \neg\forall_y Q(x, y)$

$= \forall_x \forall_y \neg P(x, y) \vee \exists_x \exists_y \neg Q(x, y)$

Statement	When True?	When False?
$\forall x\, \forall y\, P(x,y)$ $\forall y\, \forall x\, P(x,y)$	$P(x,y)$ is true for every pair x,y	There is a pair x,y for which $P(x,y)$ is false
$\forall x\, \exists y\, P(x,y)$	For every x there is a y for which $P(x,y)$ is true	There is an x such that $P(x,y)$ is false for every y
$\exists x\, \forall y\, P(x,y)$	There is an x for which $P(x,y)$ is true for every y	For every x there is a y for which $P(x,y)$ is false
$\exists x\, \exists y\, P(x,y)$ $\exists y\, \exists x\, P(x,y)$	There is a pair x,y for which $P(x,y)$ is true	$P(x,y)$ is false for every pair x,y

$\forall x\, \exists y\, P(x, y)$

$= \neg[\forall x\, \exists y\, P(x, y)]$

$= \exists_x \neg \exists y\, P(x, y)$

$= \boldsymbol{\exists_x \forall_y \neg P(x, y)}$

$\exists_x \forall_y P(x, y)$

$= \neg[\exists_x \forall_y P(x, y)]$

$= \forall_x \neg \forall_y P(x, y)$

$\boldsymbol{= \forall_x \exists_y \neg P(x, y)}$

3. TRANSLATING FROM NESTED QUANTIFIERS INTO ENGLISH:

Q. Translate the statement:

$\forall_x[C(x) \lor \exists_y(C(y) \land F(x, y))]$ into English, where

$C(x)$ is "x has a computer,"

$F(x, y)$ is "x and y are friends," and the domain for both x and y consists of all students in your school.

<u>Solution:</u>

The statement says that "for every student x in your school, x has a computer or there is a student y such that y has a computer and x and y are friends."

In other words, "*Every student in your school has a computer or has a friend who has a computer*"

3. TRANSLATING FROM NESTED QUANTIFIERS INTO ENGLISH:

Q. Translate the statement:
$\forall_x$**[**$S(x) \lor \exists_y(S(y) \land F(x, y))$**]** into English, where
$S(x)$ is "x uses snapchat"
$F(x, y)$ is "x and y are friends," and the domain for both x and y consists of all students in your school.

<u>Solution:</u>

The statement says that "for every student x in your school, x either uses snapchat or there is a student y such that y uses snapchat and y and x are friends."

In other words, "*Every student in your school either uses snapchat or are friends with a student who uses snapchat*"

3. TRANSLATING FROM NESTED QUANTIFIERS INTO ENGLISH:

Q. Translate the statement

$\exists_x \forall_y \forall_z((F(x, y) \land F(x, z) \land (y != z)) \rightarrow \neg F(y,z))$

into English, where F (a, b) means a and b are friends and the domain for x, y, and z consists of all students in your school.

Solution:

We first examine the expression $(F(x, y) \land F(x, z) \land (y = z)) \rightarrow \neg F(y, z)$. This expression says that if students x and y are friends, and students x and z are friends, and furthermore, if y and z are not the same student, then y and z are not friends.

It follows that the original statement, which is triply quantified, says that "there is a student x such that for all students y and all students z other than y, if x and y are friends and x and z are friends, then y and z are not friends. In other words,

"There is a student none of whose friends are also friends with each other."

MATHEMATICAL FOUNDATION FOR COMPUTER SCIENCE

Prepared by: Er. Ankit Kharel

Nepal college of information technology

RULES OF INTERFERENCE FOR QUANTIFIED STATEMENT

1. UNIVERSAL INSTANTIATION:

- P(c) is true , Where c is a particular member of the Domain, given the Premise $\forall_x [p(x)]$

$$\frac{\forall_x [p(x)]}{\therefore\ p(c)}$$

Example: We can conclude from the statement "All women are wise" that "Lisa is wise" where Lisa is a member of the domain of all women.

2. UNIVERSAL GENERALIZATION:

- $\forall_x [p(x)]$ is True, given the premise that P(c) is True for all elements c in the domain.

$$\frac{p(c) \quad \text{for an arbitrary } c}{\therefore \forall_x [p(x)]}$$

Example: The Domain consist of the dogs Fido, Ruby, Laika

"Fido is cute, Ruby is cute, Laika is cute".

Therefore, all dogs in the domain are cute.

3. EXISTENTIAL INSTANTIATION :

- There is an element c in the domain for which P (c) is true if we know that $\exists_x P(x)$ is true.

- We cannot select an arbitrary value of c here, but rather it must be a c for which P (c) is true. Usually we have no knowledge of what c is, only that it exists.

$$\frac{\exists_x [p(x)]}{\therefore p(c)} \quad c$$

for some element

Example: "There is someone who got an A in the course."

"Let's call her c and say that c got an A"

4. EXISTENTIAL GENERALIZATION :

- That is, if we know one element c in the domain for which P (c) is true, then we know that $\exists_x P(x)$ is true.

$$\frac{p(c) \quad \text{for some element } c}{\therefore \ \exists_x [p(x)]}$$

Example: "Michelle got an A in the class."

"Therefore, there is someone who got an A in the class."

Q.1) "All King are men"

"All men are mortal"

$\therefore$ "All Kings are mortal

Solution:

Defining variables:

K(x): "x is king"

M(x): "x is man"

M_0(x): "x is mortal"

Hypothesis: i) $\forall_x[K(x) \rightarrow M(x)$

ii) $\forall_x[M(x) \rightarrow M_o(x)$

Conclusion: $\therefore \forall_x[K(x) \rightarrow M_o(x)]$

STEPS	REASONS
1. $\forall_x[K(x) \rightarrow M(x)]$	GIVEN HYPOTHESIS
2. $K(c) \rightarrow M(c)$	UNIVERSAL INSTANTIATION ON 1
3. $\forall_x[M(x) \rightarrow M_o(x)]$	GIVEN HYPOTHESIS
4. $M(c) \rightarrow M_o(c)$	UNIVERSAL INSTANTIATION ON 3
5. $K(c) \rightarrow M_o(c)$	HYPOTHETICAL SYLLOGISM ON 2 & 4
6. $\forall_x[K(x) \rightarrow M_o(x)]$	UNIVERSAL GENERALIZATION ON 5

Q.2) Show that the premises "*Everyone in this discrete mathematics class has taken a course in computer science*" and "*Sita is a student in this class*" imply the conclusion "*Marla has taken a course in computer science.*"

Solution:

Defining variables:

D(x): "*x is in this discrete mathematics class*"

C(x): "*x has taken a course in computer science.*"

Hypothesis: i) $\forall_x[D(x) \rightarrow C(x)]$

ii) D(sita)

Conclusion: $\therefore$ C(sita)

STEPS	REASONS
1. $\forall_x[D(x) \rightarrow C(x)]$	GIVEN HYPOTHESIS
2. $D(sita) \rightarrow C(sita)$	UNIVERSAL INSTANTIATION ON 1
3. D(sita)	GIVEN HYPOTHESIS
4. C(sita)	MODUS PONENS ON 2 & 3

Q.3) Show that the premises *"A student in this class has not read the book,"* and *"Everyone in this class passed the first exam"* imply the conclusion *"Someone who passed the first exam has not read the book."*

<u>Solution:</u>

Defining variables:

C(x):"x is in this class"

B(x):"x has read the book"

P(x):"x passed the first exam"

<u>Hypothesis:</u> i) $\exists_x[C(x) \land \neg B(x)]$

ii) $\forall_x[C(x) \rightarrow P(x)]$

<u>Conclusion:</u> $\therefore \exists_x[P(x) \land \neg B(x)]$

STEPS	REASONS
1. $\exists_x[C(x) \land \neg B(x)]$	GIVEN HYPOTHESIS
2. $C(a) \land \neg B(a)$	EXISTENTIAL INSTANTIATION ON 1
3. $C(a)$	SIMPLIFICATION ON 2
4. $\forall_x[C(x) \rightarrow P(x)]$	GIVEN HYPOTHESIS
5. $C(a) \rightarrow P(a)$	UNIVERSAL INSTANTIATION FROM (4)
6. $P(a)$	MODUS PONENS FROM (3) AND (5)
7. $\neg B(a)$	SIMPLIFICATION ON 2
8. $P(a) \land \neg B(a)$	CONJUNCTION FROM (6) AND (7)
9. $\exists_x[P(x) \land \neg B(x)]$	EXISTENTIAL GENERALIZATION FROM (8)

Q.4) Show that the premises *"All rock music is loud"*, *"Some rock music exist"*, imply the conclusion *"Some Loud music exists"*

Solution:

Defining variables:

R(x):*"x is in rock music"*

L(x):*"x is loud music"*

Hypothesis: i) $\forall_x[R(x) \rightarrow L(x)]$

ii) $\exists_x[R(x)]$

Conclusion: $\therefore \exists_x[L(x)]$

STEPS	REASONS
1. $\forall_x[R(x) \rightarrow L(x)]$	GIVEN HYPOTHESIS
2. $R(c) \rightarrow L(c)$	UNIVERSAL INSTANTIATION ON 1
3. $\exists_x[R(x)]$	GIVEN HYPOTHESIS
4. $R\ (c)$	EXISTENTIAL INSTANTIATION ON 3
5. $L(c)$	MODUS PONENS FROM (2) AND (4)
6. $\exists_x[L(x)]$	EXISTENTIAL GENERALIZATION FROM (5)

Q.4) Show that the premises *"Every computer science student works harder than somebody", "Everyone who works harder than any other person gets less sleep than that person", "Maria is a computer science student"* Implies the conclusion *"Maria gets less sleep than someone else"*

Solution:

Defining variables:

C(x) : *"x is Computer science student"*

W(x, y) : *"x works harder than y"*

S(x, y) : "x gets less sleep than y"

C(m) : "Maria is a computer science student"

Hypothesis: i) $\forall_x \exists_y [C(x) \rightarrow W(x, y)]$

ii) $[\forall_x \exists_y W(x, y)] \rightarrow S(x, y)$

iii) $C(m)$

Conclusion: $\therefore \exists_y [S(m, y)]$

Hypothesis: i) $\forall_x \exists_y [C(x) \rightarrow W(x, y)]$

ii) $[\forall_x \exists_y W(x, y)] \rightarrow S(x, y)$

iii) $C(m)$

Conclusion: $\therefore \exists_y [S(m, y)]$

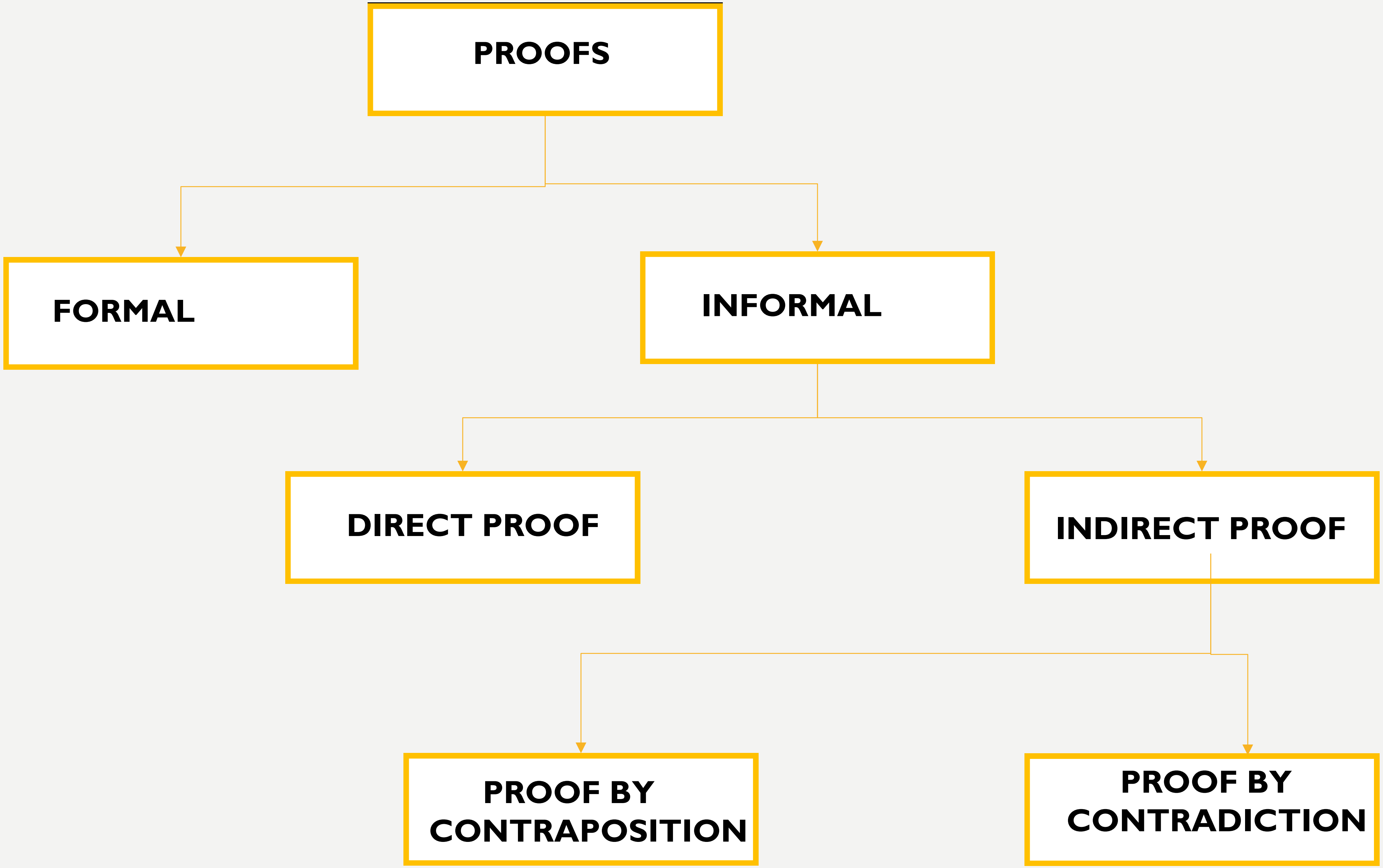
STEPS	REASONS
1. $\forall_x \exists_y [C(x) \rightarrow W(x, y)]$	GIVEN HYPOTHESIS
2. $\forall_x [C(x) \rightarrow W(x, c)]$	EXISTENTIAL INSTANTIATION ON 1
3. $C(m) \rightarrow W(m, c)$	UNIVERSAL INSTANTIATION ON 2
4. $[\forall_x \exists_y W(x, y)] \rightarrow S(x, y)$	GIVEN HYPOTHESIS
5. $[\forall_x W(x, c)] \rightarrow S(x, c)$	EXISTENTIAL INSTANTIATION ON 4
6. $W(m\ c) \rightarrow S(m, c)$	UNIVERSAL INSTANTIATION ON 5
7. $C(m) \rightarrow S(m, c)$	FROM 3 AND 6
8. $C(m)$	GIVEN HYPOTHESIS
9. $S(m, c)$	MODUS TOLLENS ON 7 AND 8
10. $\exists_y [S(m, y)]$	EXISTENTIAL GENERALIZATION FROM 9

Students who pass the course either do the homework or attend lecture;” “Bob did not attend every lecture;” “Bob passed the course.” Therefore “ Bob must have done the homework.”

MATHEMATICAL FOUNDATION FOR COMPUTER SCIENCE

Prepared by: Er. Ankit Kharel

Nepal college of information technology


PROOFS
FORMAL
INFORMAL
DIRECT PROOF
INDIRECT PROOF
PROOF BY CONTRAPOSITION
PROOF BY CONTRADICTION

PROOF:

- An argument used to establish the truth of mathematical statement is called Proof.
- Mathematical proofs use **deductive reasoning**, where a conclusion is drawn from multiple premises.
- The premises in the proof are called statements.
- While establishing the truth, different rules and already proven facts are used.
- <u>INDUCTIVE REASONONG :</u> Drawing a general conclusion from what we see around us. For example, if all the sheep you have ever seen were white, you might conclude that all sheep are white.
- <u>DEDUCTIVE REASONONG :</u>You start from a general statement you know for sure is true and draw conclusions about a specific case. For example, if you know for a fact that all sheep like to eat grass, and you also know that the creature standing in front of you is a sheep, then you know with certainty that it likes grass.

SOME TERMINOLOGIES:

1. THEOREM: A mathematical statement that is proved using rigorous mathematical reasoning. The process of showing a theorem to be correct is called a proof.

2. **AXIOM:** An axiom is a statement, usually considered to be self- evident, that is assumed to be True without proof. It is used as starting point in mathematical proof.

Example: Parallel lines in same plane , do not meet one another in either direction when extended infinitely.

3. COROLLARY: A corollary is the theorem that can be proven to be a Logical consequence of another theorem.

Example: If a+ b=c then an example of corollary is c= b- a.

SOME TERMINOLOGIES:

4. CONJECTURE : A conjecture is a mathematical statement that has not yet been rigorously proved. Conjectures arises when one notices a pattern that holds True for many cases.

Example: 2, 4, 6, ,8 ,10 ,12, ?

The next number is more likely to be 14.

5. AXIOM: It is generally minor, proven proposition which is used as a stepping stone to a larger result. It is also known as a "Helping Theorem" or " Auxillary Theorem"

Example: For all real numbers r, $|-r| = |r|$

FORMAL & INFORMAL PROOF:

Definition: A formal proof of a conclusion q given hypotheses p1, p2, . . . , pn is a sequence of steps, each of which applies some inference rule to hypotheses or previously proven statements (antecedents) to yield a new true statement (the consequent).

Informal proof : where more than one rule of inference may be used in each step, where steps may be skipped, where the axioms being assumed and rule of inference used are not explicitly stated.

1. DIRECT PROOF:

- A direct proof of a conditional statement $p \rightarrow q$ is constructed when the first step is the assumption that p is true; subsequent steps are constructed using rules of inference, with the final step showing that q must also be true

- A direct proof shows that a conditional statement $p \rightarrow q$ is true by showing that if p is true, then q must also be true, so that the combination p true and q false never occurs.

- In a direct proof, we assume that p is true and use axioms, definitions, and previously proven theorems, together with rules of inference, to show that q must also be true

Example:

1. Prove that "If n is odd, then n^2 is odd."

Solution:

Let,

p: Hypothesis: " n is odd"

q: Conclusion: "n^2 is odd"

Now, we assume Hypothesis is TRUE. i.e.

n is odd(TRUE)

By the definition of odd integer, we can write,

$n= 2k+1$; for integer k

Squaring both sides,

$$\begin{aligned} n^2 &= (2k+1)^2 \\ &= 4k^2 + 4k + 1 \\ &=2(2k^2 + 2k) +1 \\ &=2k_1 + 1; \text{ where } k_1 = (2k^2 + 2k) \text{ is an integer} \end{aligned}$$

Therefore, n2 is odd.

2. Prove that "If 'm' and 'n' are odd, then 'm + n' is even." [The sum of two odd numbers is even.]

Solution:

Let,

p: Hypothesis: " m and n are odd"

q: Conclusion: "m + n is even"

Now, we assume Hypothesis is TRUE. i.e.

m and n are odd(TRUE)

By the definition of odd integer, we can write,

$m = 2i+1$; for integer i

$n = 2j+1$; for integer j

Now,

$$\begin{aligned} m + n &= (2i+1) + (2j+1) \\ &= 2i+2j+2 \\ &= 2(i+j+1) \\ &= 2k \qquad ; \text{where } k = (i+j+1) \text{ is an integer} \end{aligned}$$

Therefore, m + n is even.

3. If x is an even integer, then $x^2 - 6x+5$ is odd.

Proof.

Suppose x is an even integer. Then $x = 2a$ for some $a \in Z$, by definition of an even integer.

So

$$\begin{aligned} x^2-6x+5 &= (2a)^2-6(2a)+5 \\ &= 4a^2-12a+5 \\ &= 4a^2-12a+4+1 \\ &= 2(2a^2-6a+2)+1. \end{aligned}$$

Therefore we have $x^2 - 6x+5 = 2b + 1$, where $b = 2a^2 - 6a+2 \in Z$.

Consequently $x^2 - 6x+5$ is odd, by definition of an odd number

4. If n is any even integer, then $(-1)^n = 1$.

Proof:

Suppose n is even integer. [We must show that $(-1)^n = 1$.].

Then by the definition of even numbers,

$n = 2k$ for some integer k

we have

$$\begin{aligned} (-1)^n &= (-1)^{2k} \\ &= ((-1)^2)^k \\ &= (1)^k \\ &= 1 \end{aligned}$$

This is what was to be shown. And this completes the proof.

1. INDIRECT PROOF:

- Direct proofs lead from the premises of a theorem to the conclusion. They begin with the premises, continue with a sequence of deductions, and end with the conclusion.

- However, we will see that attempts at direct proofs often reach dead ends. We need other methods of proving theorems of the form $\forall x(P(x) \rightarrow Q(x))$. Proofs of theorems of this type that are not direct proofs, that is, that do not start with the premises and end with the conclusion, are called indirect proofs

i. Proof by Contraposition
ii. Proof by Contradiction

1.1 PROOF BY CONTRAPOSITION:

- Proofs by contraposition make use of the fact that the conditional statement $p \to q$ is equivalent to its contrapositive, $\neg q \to \neg p$. This means that the conditional statement $p \to q$ can be proved by showing that its contrapositive, $\neg q \to \neg p$, is true.

- In a proof by contraposition of $p \to q$, we take $\neg q$ as a premise, and using axioms, definitions, and previously proven theorems, together with rules of inference, we show that $\neg p$ must follow

Q.1 Prove that if n is an integer and 3n + 2 is odd, then n is odd.

Solution:

We first attempt a direct proof.

If 3n + 2 is odd, then n is odd.($p \rightarrow q$)

p:“3n + 2 is odd”

q:“n is odd”

To construct a direct proof, we first assume that 3n + 2 is an odd integer.

This means that 3n + 2 = 2k + 1 for some integer k.

$3n + 2 = 2k + 1$

$n = (2k-1)/3$

DEAD END

$p \rightarrow q = \neg q \rightarrow \neg p$[if n is even then 3n+2 is even]

$\neg q$=“n is even”

$\neg p$= “3n + 2 is even”

Now , from the definition of an even integer

n=2k, for some integer k

Substituting 2k for n ,We get,

$3(2k) + 2$

$=6k + 2$

$=2(3k + 1)$

i.e. 3n + 2 is even because it is a multiple of 2

Therefore, 3n+2 is even.

1.2 PROOF BY CONTRADICTION:

- The basic idea is to assume that the statement we want to prove is false, and then show that this assumption leads to nonsense. We are then led to conclude that we were wrong to assume the statement was false, so the statement must be true.

Prove there exist no integer a, b for which 5a + 15b =1.

Solution:

Step 1: Assume there exist integer a and b for which 5a + 15b =1.

Now,

$$5a + 15b = 1$$
$$5(a + 3b) = 1$$
$$a + 3b = 1/5$$

Because a and b are integers, a + 3b must also be an integer **[CONTRADICTION]**

Therefore, There exist no integer a, b for which 5a + 15b = 1.

Proposition: If $a, b \in \mathbb{Z}$, then $a^2 - 4b \neq 2$.

Proof.

"If a and b are integers then, $a^2 - 4b \neq 2$"

Suppose this proposition is false. This conditional statement being false means:

There exist numbers a and b for which $a, b \in \mathbb{Z}$ is true but $a^2 - 4b \neq 2$ is false.

"If a and b are integers then, $a^2 - 4b = 2$"

Thus there exist integers $a, b \in \mathbb{Z}$ for which $a^2 - 4b = 2$.

From this equation we get

$a^2 = 4b + 2 = 2(2b + 1)$, so a2 is even. Since a^2 is even, it follows that a is even, so $a = 2c$ for some integer c.

Now plug $a = 2c$ back into the boxed equation $a^2 - 4b = 2$. We get

$(2c)^2 - 4b = 2$, so $4c^2 - 4b = 2$. Dividing by 2, we get $2c^2 - 2b = 1$.

Therefore $1 = 2(c^2 - b)$, and since $c^2 - b \in \mathbb{Z}$, it follows that 1 is even. Since we know 1 is not even, something went wrong. But all the logic after the first line of the proof is correct, so it must be that the first line was incorrect. In other words, we were wrong to assume the proposition was false. Thus the proposition is true.

Prove that for all integer n, if $n^3 + 5$ is odd then n is even.

Solution:

Here,

Assume the conclusion, i.e. n is odd

Because n is odd ,We can write,

$N = 2k + 1$

Putting value of n in $n^3 + 5$, We get

$=(2k + 1)^3 + 5$

$=8k^3 + 12k^2 + 6k +1 + 5$

$= 8k^3 + 12k^2 + 6k + 6$

$=2[8k^3 + 12k^2 + 6k + 6]$

=Even**[CONTRADICTION]**

Therefore, If $n^3 + 5$ is odd then n is even

Prove $\sqrt{2}$ is a irrational number using proof by contradiction.

Suppose $\sqrt{2}$ is rational. Then integers a and b exist so that $\sqrt{2} = a/b$.

Without loss of generality we can assume that a and b have no factors in common (i.e., the fraction is in simplest form).

Multiplying both sides by b and squaring,
we have $2b^2 = a^2$ so we see that a^2 is even.
This means that a is even so $a = 2m$ for some $m \in Z$.
Then $2b^2 = (2m)^2$
$= 4m^2$ which, after dividing by 2, gives $b^2 = 2m^2$ so b^2 is even. This means b is even.

We've seen that if $\sqrt{2} = a/b$ then both a and b must be even and so are both multiples of 2. This contradicts the fact that we know a and b can be chosen to have no common factors. Thus, $\sqrt{2}$ must not be rational, so $\sqrt{2}$ is irrational.

PRINCIPLE OF MATHEMATICAL INDUCTION:

- Let P(n) be a statement. Now, our concern is to show that P(n) is True using Mathematical Induction.

  a) First we show that P(n) is True for some initial value like n =0, 1 ….This is called the Basic Step.

  b) Then, we assume that P(n) is True for any arbitrary value k i.e. P(k) is True and show that P(n) is True for 'k + 1' i.e. P(k+1) is True. This step is called inductive step

Thus, Mathematical Induction can be defined as:

$[P(1) \wedge (P(k) \rightarrow P(k+1))] \rightarrow P(n)$

Q. Show that if n is positive integer then,

$$1 + 2 + \ldots\ldots\ldots + n = [n(n + 1)]/2$$

<u>Solution:</u>

Let, P(n) be the proposition that the sum of first n positive integer , $1 + 2 + \ldots\ldots\ldots + n = [n(n + 1)]/2$

<u>Basic Step:</u> When $n=1$,

$1= [1(1+1)]/2$

$1=1$(TRUE) i.e. P(1) is True.

<u>Inductive Step:</u> Assume P(k) holds for arbitrary integer k.

i.e. $1+2+\ldots\ldots+k = [k(k+1)]/2$

Under this assumption , it must be shown that P(k+1) is True

$$1 + 2 + \ldots + k + (k+1)= [(k+1)(k+2)]/2$$

L.H.S.

$1 + 2 + \ldots + k + (k+1)$

$=[k(k+1)]/2 + (k + 1)$

$=[k(k+1) + 2k + 2]/2$

$=[k(k+1) + 2(k+1)]/2$

$=[(k+1)(k+2)]/2$

=R.H.S

Therefore, P(n) is true.

Q. Use Mathematical induction to show that:

$$\mathbf{2 + 2^2 + \ldots\ldots\ldots\ldots\ldots\ldots + 2^n = 2^{n+1} - 2}$$

<u>Solution:</u>

Let, P(n) be the proposition that, $2 + 2^2 + \ldots\ldots\ldots\ldots\ldots\ldots + 2^n = 2^{n+1} - 2$

<u>Basic Step</u>: When n=1,

$2 = 2^2 - 2$

$2=2$(TRUE) i.e. P(1) is True.

<u>Inductive Step</u>: Assume P(k) holds for arbitrary integer k.

i.e. $2 + 2^2 + \ldots\ldots\ldots\ldots\ldots\ldots + 2^k = 2^{k+1} - 2$

Under this assumption , it must be shown that P(k+1) is True

$$2 + 2^2 + \ldots\ldots\ldots\ldots\ldots\ldots + 2^k + 2^{k+1} = 2^{(k+1)+1} - 2$$

L.H.S.

$$2 + 2^2 + \ldots\ldots\ldots\ldots\ldots\ldots + 2^k + 2^{k+1}$$
$$= 2^{k+1} - 2 + 2^{k+1}$$
$$= 2 . 2^{k+1} - 2$$
$$= 2^{(k+1)+1} - 2$$
$$= \text{R.H.S}$$

Therefore, P(n) is true.

Q. Use Mathematical induction to show that:

$8^n - 3^n$ is divisible by 5.[$n \geq 1$]

<u>Solution:</u>

Let, P(n) be the proposition that, $8^n - 3^n$ is divisible by 5

<u>Basic Step</u>: When n=1,

$8^1 - 3^1$ is divisible by 5

5 is divisible by 5(TRUE) i.e. P(1) is True.

<u>Inductive Step:</u> Assume P(k) holds for arbitrary integer k.

i.e. $8^k - 3^k$ is divisible by 5

Under this assumption , it must be shown that P(k+1) is True

i.e. $8^{k+1} - 3^{k+1}$ is divisible by 5

Now,

$8^{k+1} - 3^{k+1}$

$=8^k.8 - 3^k.3$

$=8^k(5+3) - 3^k.3$

$= 8^k.5 + 8^k.3 - 3^k.3$

$= 8^k.5 + 3(8^k - 3^k)$

Here, $8^k.5$ is multiple of 5 and $(8^k - 3^k)$is divisible by 5.

Therefore, P(n) is true.

Question 1. *Prove using mathematical induction that for all $n \geq 1$,*

$$1 + 4 + 7 + \cdots + (3n - 2) = \frac{n(3n-1)}{2}.$$

Solution.

For any integer $n \geq 1$, let P_n be the statement that

$$1 + 4 + 7 + \cdots + (3n - 2) = \frac{n(3n-1)}{2}.$$

Base Case. The statement P_1 says that

$$1 = \frac{1(3-1)}{2},$$

which is true.

Inductive Step. Fix $k \geq 1$, and suppose that P_k holds, that is,

$$1 + 4 + 7 + \cdots + (3k - 2) = \frac{k(3k-1)}{2}.$$

It remains to show that P_{k+1} holds, that is,

$$1 + 4 + 7 + \cdots + (3(k+1) - 2) = \frac{(k+1)(3(k+1)-1)}{2}.$$

$$\begin{aligned}
1 + 4 + 7 + \cdots + (3(k+1) - 2) &= 1 + 4 + 7 + \cdots + (3(k+1) - 2) \\
&= 1 + 4 + 7 + \cdots + (3k + 1) \\
&= 1 + 4 + 7 + \cdots + (3k - 2) + (3k + 1) \\
&= \frac{k(3k-1)}{2} + (3k + 1) \\
&= \frac{k(3k-1) + 2(3k+1)}{2} \\
&= \frac{3k^2 - k + 6k + 2)}{2} \\
&= \frac{3k^2 + 5k + 2)}{2} \\
&= \frac{(k+1)(3k+2)}{2} \\
&= \frac{(k+1)(3(k+1)-1)}{2}.
\end{aligned}$$

Therefore P_{k+1} holds.

Thus, by the principle of mathematical induction, for all $n \geq 1$, P_n holds. □

Question 2. *Use the Principle of Mathematical Induction to verify that, for n any positive integer, $6^n - 1$ is divisible by 5.*

Solution.

For any $n \geq 1$, let P_n be the statement that $6^n - 1$ is divisible by 5.

Base Case. The statement P_1 says that

$$6^1 - 1 = 6 - 1 = 5$$

is divisible by 5, which is true.

Inductive Step. Fix $k \geq 1$, and suppose that P_k holds, that is, $6^k - 1$ is divisible by 5.

It remains to show that P_{k+1} holds, that is, that $6^{k+1} - 1$ is divisible by 5.

$$\begin{aligned} 6^{k+1} - 1 &= 6(6^k) - 1 \\ &= 6(6^k - 1) - 1 + 6 \\ &= 6(6^k - 1) + 5. \end{aligned}$$

By P_k, the first term $6(6^k - 1)$ is divisible by 5, the second term is clearly divisible by 5. Therefore the left hand side is also divisible by 5. Therefore P_{k+1} holds.

Thus by the principle of mathematical induction, for all $n \geq 1$, P_n holds.

Question 3. *Verify that for all $n \geq 1$, the sum of the squares of the first $2n$ positive integers is given by the formula*

$$1^2 + 2^2 + 3^2 + \cdots + (2n)^2 = \frac{n(2n+1)(4n+1)}{3}$$

Solution.

For any integer $n \geq 1$, let P_n be the statement that

$$1^2 + 2^2 + 3^2 + \cdots + (2n)^2 = \frac{n(2n+1)(4n+1)}{3}.$$

Base Case. The statement P_1 says that

$$1^2 + 2^2 = \frac{(1)(2(1)+1)(4(1)+1)}{3} = \frac{3(5)}{3} = 5,$$

which is true.

Inductive Step. Fix $k \geq 1$, and suppose that P_k holds, that is,

$$1^2 + 2^2 + 3^2 + \cdots + (2k)^2 = \frac{k(2k+1)(4k+1)}{3}.$$

It remains to show that P_{k+1} holds, that is,

$$1^2 + 2^2 + 3^2 + \cdots + (2(k+1))^2 = \frac{(k+1)(2(k+1)+1)(4(k+1)+1)}{3}.$$

$$\begin{aligned}
1^2 + 2^2 + 3^2 + \cdots + (2(k+1))^2 &= 1^2 + 2^2 + 3^2 + \cdots + (2k+2)^2 \\
&= 1^2 + 2^2 + 3^2 + \cdots + (2k)^2 + (2k+1)^2 + (2k+2)^2 \\
&= \frac{k(2k+1)(4k+1)}{3} + (2k+1)^2 + (2k+2)^2 && \text{(by } P_k\text{)} \\
&= \frac{k(2k+1)(4k+1)}{3} + \frac{3(2k+1)^2 + 3(2k+2)^2}{3} \\
&= \frac{k(2k+1)(4k+1) + 3(2k+1)^2 + 3(2k+2)^2}{3} \\
&= \frac{k(8k^2+6k+1) + 3(4k^2+4k+1) + 3(4k^2+8k+4)}{3} \\
&= \frac{(8k^3+6k^2+k) + (12k^2+12k+3) + (12k^2+24k+12)}{3} \\
&= \frac{8k^3+30k^2+37k+15}{3}
\end{aligned}$$

On the other side of P_{k+1},

$$\frac{(k+1)(2(k+1)+1)(4(k+1)+1)}{3} = \frac{(k+1)(2k+2+1)(4k+4+1)}{3}$$

$$\begin{aligned}
&= \frac{(k+1)(2k+3)(4k+5)}{3} \\
&= \frac{(2k^2+5k+3)(4k+5)}{3} \\
&= \frac{8k^3+30k^2+37k+15}{3}.
\end{aligned}$$

Therefore P_{k+1} holds.

Thus, by the principle of mathematical induction, for all $n \geq 1$, P_n holds. $\square$

Question 4. *Consider the sequence of real numbers defined by the relations*

$$x_1 = 1 \text{ and } x_{n+1} = \sqrt{1 + 2x_n} \text{ for } n \geq 1.$$

Use the Principle of Mathematical Induction to show that $x_n < 4$ for all $n \geq 1$.

Solution.

For any $n \geq 1$, let P_n be the statement that $x_n < 4$.

Base Case. The statement P_1 says that $x_1 = 1 < 4$, which is true.

Inductive Step. Fix $k \geq 1$, and suppose that P_k holds, that is, $x_k < 4$.

It remains to show that P_{k+1} holds, that is, that $x_{k+1} < 4$.

$$\begin{aligned} x_{k+1} &= \sqrt{1 + 2x_k} \\ &< \sqrt{1 + 2(4)} \\ &= \sqrt{9} \\ &= 3 \\ &< 4. \end{aligned}$$

Therefore P_{k+1} holds.

Thus by the principle of mathematical induction, for all $n \geq 1$, P_n holds.

Question 5. *Show that* $n! > 3^n$ *for* $n \geq 7$.

Solution.

For any $n \geq 7$, let P_n be the statement that $n! > 3^n$.

Base Case. The statement P_7 says that $7! = 7 \times 6 \times 5 \times 4 \times 3 \times 2 \times 1 = 5040 > 3^7 = 2187$, which is true.

Inductive Step. Fix $k \geq 7$, and suppose that P_k holds, that is, $k! > 3^k$.

It remains to show that P_{k+1} holds, that is, that $(k+1)! > 3^{k+1}$.

$$\begin{aligned}
(k+1)! &= (k+1)k! \\
&> (k+1)3^k \\
&\geq (7+1)3^k \\
&= 8 \times 3^k \\
&> 3 \times 3^k \\
&= 3^{k+1}.
\end{aligned}$$

Therefore P_{k+1} holds.

Thus by the principle of mathematical induction, for all $n \geq 1$, P_n holds.

Question 6. *Let* $p_0 = 1$, $p_1 = \cos\theta$ *(for* θ *some fixed constant) and* $p_{n+1} = 2p_1p_n - p_{n-1}$ *for* $n \geq 1$. *Use an extended Principle of Mathematical Induction to prove that* $p_n = \cos(n\theta)$ *for* $n \geq 0$.

Solution.

For any $n \geq 0$, let P_n be the statement that $p_n = \cos(n\theta)$.

Base Cases. The statement P_0 says that $p_0 = 1 = \cos(0\theta) = 1$, which is true. The statement P_1 says that $p_1 = \cos\theta = \cos(1\theta)$, which is true.

Inductive Step. Fix $k \geq 0$, and suppose that both P_k and P_{k+1} hold, that is, $p_k = \cos(k\theta)$, and $p_{k+1} = \cos((k+1)\theta)$.

It remains to show that P_{k+2} holds, that is, that $p_{k+2} = \cos((k+2)\theta)$.

We have the following identities:

$$\cos(a+b) = \cos a \cos b - \sin a \sin b$$

$$\cos(a-b) = \cos a \cos b + \sin a \sin b$$

Therefore, using the first identity when $a = \theta$ and $b = (k+1)\theta$, we have

$$\cos(\theta + (k+1)\theta) = \cos\theta\cos(k+1)\theta - \sin\theta\sin(k+1)\theta,$$

and using the second identity when $a = (k+1)\theta$ and $b = \theta$, we have

$$\cos((k+1)\theta - \theta) = \cos(k+1)\theta\cos\theta + \sin(k+1)\theta\sin\theta.$$

Therefore,

$$\begin{aligned}
p_{k+2} &= 2p_1p_{k+1} - p_k \\
&= 2(\cos\theta)(\cos((k+1)\theta)) - \cos(k\theta) \\
&= (\cos\theta)(\cos((k+1)\theta)) + (\cos\theta)(\cos((k+1)\theta)) - \cos(k\theta) \\
&= \cos(\theta + (k+1)\theta) + \sin\theta\sin(k+1)\theta + (\cos\theta)(\cos((k+1)\theta)) - \cos(k\theta) \\
&= \cos((k+2)\theta) + \sin\theta\sin(k+1)\theta + (\cos\theta)(\cos((k+1)\theta)) - \cos(k\theta) \\
&= \cos((k+2)\theta) + \sin\theta\sin(k+1)\theta + \cos((k+1)\theta - \theta) - \sin(k+1)\theta\sin\theta - \cos(k\theta) \\
&= \cos((k+2)\theta) + \cos(k\theta) - \cos(k\theta) \\
&= \cos((k+2)\theta).
\end{aligned}$$

Therefore P_{k+2} holds.

Thus by the principle of mathematical induction, for all $n \geq 1$, P_n holds.

Question 7. *Consider the famous Fibonacci sequence* $\{x_n\}_{n=1}^{\infty}$, *defined by the relations* $x_1 = 1$, $x_2 = 1$, *and* $x_n = x_{n-1} + x_{n-2}$ *for* $n \geq 3$.

(a) Compute x_{20}.

(b) Use an extended Principle of Mathematical Induction in order to show that for $n \geq 1$,

$$x_n = \frac{1}{\sqrt{5}}\left[\left(\frac{1+\sqrt{5}}{2}\right)^n - \left(\frac{1-\sqrt{5}}{2}\right)^n\right].$$

(c) Use the result of part (b) to compute x_{20}.

Solution.

(a)

$$1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233, 377, 610, 987, 1597, 2584, 4181, 6765$$

(b) For any $n \geq 1$, let P_n be the statement that

$$x_n = \frac{1}{\sqrt{5}}\left[\left(\frac{1+\sqrt{5}}{2}\right)^n - \left(\frac{1-\sqrt{5}}{2}\right)^n\right].$$

Base Case. The statement P_1 says that

$$\begin{aligned}
x_1 &= \frac{1}{\sqrt{5}}\left[\left(\frac{1+\sqrt{5}}{2}\right)^1 - \left(\frac{1-\sqrt{5}}{2}\right)^1\right] \\
&= \frac{1}{\sqrt{5}}\left[\frac{1+\sqrt{5}}{2} - \frac{1-\sqrt{5}}{2}\right] \\
&= \frac{1}{\sqrt{5}}\left[\frac{1+\sqrt{5}-1+\sqrt{5}}{2}\right] \\
&= \frac{1}{\sqrt{5}}\left[\frac{2\sqrt{5}}{2}\right] \\
&= 1,
\end{aligned}$$

which is true. The statement P_2 says that

$$x_2 = \frac{1}{\sqrt{5}}\left[\left(\frac{1+\sqrt{5}}{2}\right)^2 - \left(\frac{1-\sqrt{5}}{2}\right)^2\right]$$

$$\begin{aligned}
&= \frac{1}{\sqrt{5}}\left[\left(\frac{1+2\sqrt{5}+5}{4}\right) - \left(\frac{1-2\sqrt{5}+5}{4}\right)\right] \\
&= \frac{1}{\sqrt{5}}\left[\left(\frac{1+2\sqrt{5}+5-1+2\sqrt{5}-5}{4}\right)\right] \\
&= \frac{1}{\sqrt{5}}\left[\frac{4\sqrt{5}}{4}\right] \\
&= 1,
\end{aligned}$$

which is again true.

<u>Inductive Step.</u> Fix $k \geq 1$, and suppose that P_k and P_{k+1} both hold, that is,

$$x_k = \frac{1}{\sqrt{5}}\left[\left(\frac{1+\sqrt{5}}{2}\right)^k - \left(\frac{1-\sqrt{5}}{2}\right)^k\right],$$

and

$$x_{k+1} = \frac{1}{\sqrt{5}}\left[\left(\frac{1+\sqrt{5}}{2}\right)^{k+1} - \left(\frac{1-\sqrt{5}}{2}\right)^{k+1}\right].$$

It remains to show that P_{k+2} holds, that is, that

$$x_{k+2} = \frac{1}{\sqrt{5}}\left[\left(\frac{1+\sqrt{5}}{2}\right)^{k+2} - \left(\frac{1-\sqrt{5}}{2}\right)^{k+2}\right].$$

$$\begin{aligned}
x_{k+2} &= x_k + x_{k+1} \\
&= \frac{1}{\sqrt{5}}\left[\left(\frac{1+\sqrt{5}}{2}\right)^k - \left(\frac{1-\sqrt{5}}{2}\right)^k\right] + \frac{1}{\sqrt{5}}\left[\left(\frac{1+\sqrt{5}}{2}\right)^{k+1} - \left(\frac{1-\sqrt{5}}{2}\right)^{k+1}\right] \\
&= \frac{1}{\sqrt{5}}\left[\left(\frac{1+\sqrt{5}}{2}\right)^k + \left(\frac{1+\sqrt{5}}{2}\right)^{k+1} - \left(\frac{1-\sqrt{5}}{2}\right)^k - \left(\frac{1-\sqrt{5}}{2}\right)^{k+1}\right] \\
&= \frac{1}{\sqrt{5}}\left[\left(\frac{1+\sqrt{5}}{2}\right)^k \left(1+\frac{1+\sqrt{5}}{2}\right) - \left(\frac{1-\sqrt{5}}{2}\right)^k \left(1+\frac{1-\sqrt{5}}{2}\right)\right] \\
&= \frac{1}{\sqrt{5}}\left[\left(\frac{1+\sqrt{5}}{2}\right)^k \left(\frac{3+\sqrt{5}}{2}\right) - \left(\frac{1-\sqrt{5}}{2}\right)^k \left(\frac{3-\sqrt{5}}{2}\right)\right] \\
&= \frac{1}{\sqrt{5}}\left[\left(\frac{1+\sqrt{5}}{2}\right)^k \left(\frac{6+2\sqrt{5}}{4}\right) - \left(\frac{1-\sqrt{5}}{2}\right)^k \left(\frac{6-2\sqrt{5}}{4}\right)\right]
\end{aligned}$$

$$= \frac{1}{\sqrt{5}}\left[\left(\frac{1+\sqrt{5}}{2}\right)^{k}\left(\frac{1+2\sqrt{5}+5}{4}\right)-\left(\frac{1-\sqrt{5}}{2}\right)^{k}\left(\frac{1-2\sqrt{5}+5}{4}\right)\right]$$
$$= \frac{1}{\sqrt{5}}\left[\left(\frac{1+\sqrt{5}}{2}\right)^{k}\left(\frac{1+\sqrt{5}}{2}\right)^{2}-\left(\frac{1-\sqrt{5}}{2}\right)^{k}\left(\frac{1-\sqrt{5}}{2}\right)^{2}\right]$$
$$= \frac{1}{\sqrt{5}}\left[\left(\frac{1+\sqrt{5}}{2}\right)^{k+2}-\left(\frac{1-\sqrt{5}}{2}\right)^{k+2}\right].$$

Therefore P_{k+2} holds.

Thus by the principle of mathematical induction, for all $n \geq 1$, P_n holds.

(c) Plugging $n = 20$ in a calculator yields the answer quickly.

Mathematical Foundation for Computer Science

Prepared by: Er. Ankit Kharel

Nepal college of information technology

RECURRENCE RELATIONS

- Recursive Definition of Sequences.
- Solution of Linear Recursive Relation
- Solution of Non-linear Recurrence Relation.

INTRODUCTION:

Consider the following:

a) Start with number 5

b) Given any term , add 3 to get next term

If we list the term using above rule then we obtain,

5, 8, 11, 14, 17, ……………----------------------(i)

If we denote (i) as a_1 , a_2 , a_3 , a_4 , ……..,We may rephrase above instruction as:

$$\underbrace{a_n = a_{n-1} + 1}_{\text{RECURRENCE RELATION}} \quad ; \text{ with initial condition, } a_1 = 5, n \geq 2$$

If a sequence can be expressed by an equation in terms of previous element then it is called the Recurrence Relation and the equation that satisfies the recurrence relation is called solution of recurrence relation.

INTRODUCTION:

Fibonacci Sequence:

0, 1, 1, 2, 3, 5, 8, 13, 21, ...

It can be generated by using following recurrence relation,

$$F_n = F_{n-1} + F_{n-2} \; ; F_0 = 0, F_1 = 1$$

- $a_n = 2a_{n-1} - a_{n-2}$; $a_1=3$, $a_2=6$……… for n= 2, 3, 4…..

  check whether $a_n = 3n$ is its solution ?

  $a_n = 2\,[3\,\{n-1\}] - 3\,[n-2] = 6n - 2 - 3n + 2 = 3n$

Hence, $a_n = 3n$ is its solution.

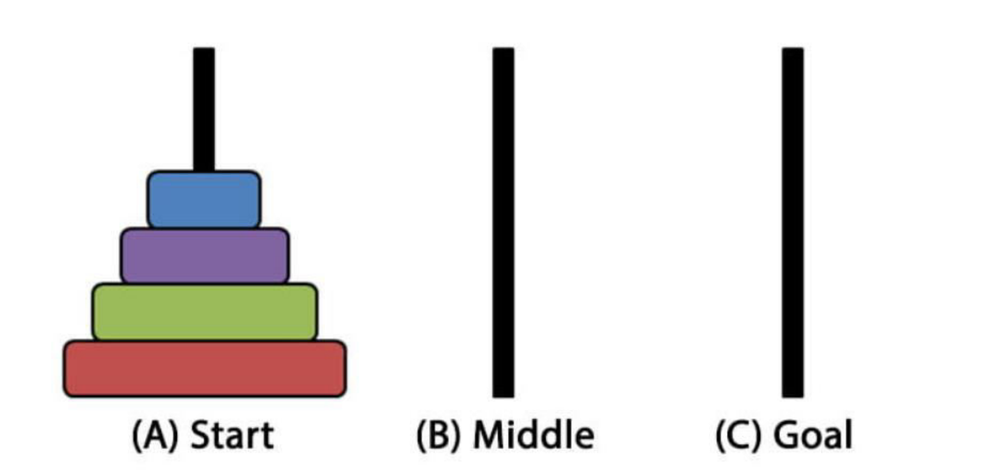
TOWER OF HANOI:

- Tower of Hanoi is a mathematical puzzle where we have three rods and n disks. The objective of the puzzle is to move the entire stack to another rod, obeying the following simple rules:

  1) Only one disk can be moved at a time.

  2) Each move consists of taking the upper disk from one of the stacks and placing it on top of another stack i.e. a disk can only be moved if it is the uppermost disk on a stack.

  3) No disk may be placed on top of a smaller disk.



TOWER OF HANOI:

Let, H_n be the total moves required to move n disk from peg-1 to peg-3.

i) First with the help of peg-2 & peg-3, move (n-1) disks from peg-1 are arranged to peg-2.
This requires, H_{n-1} moves.

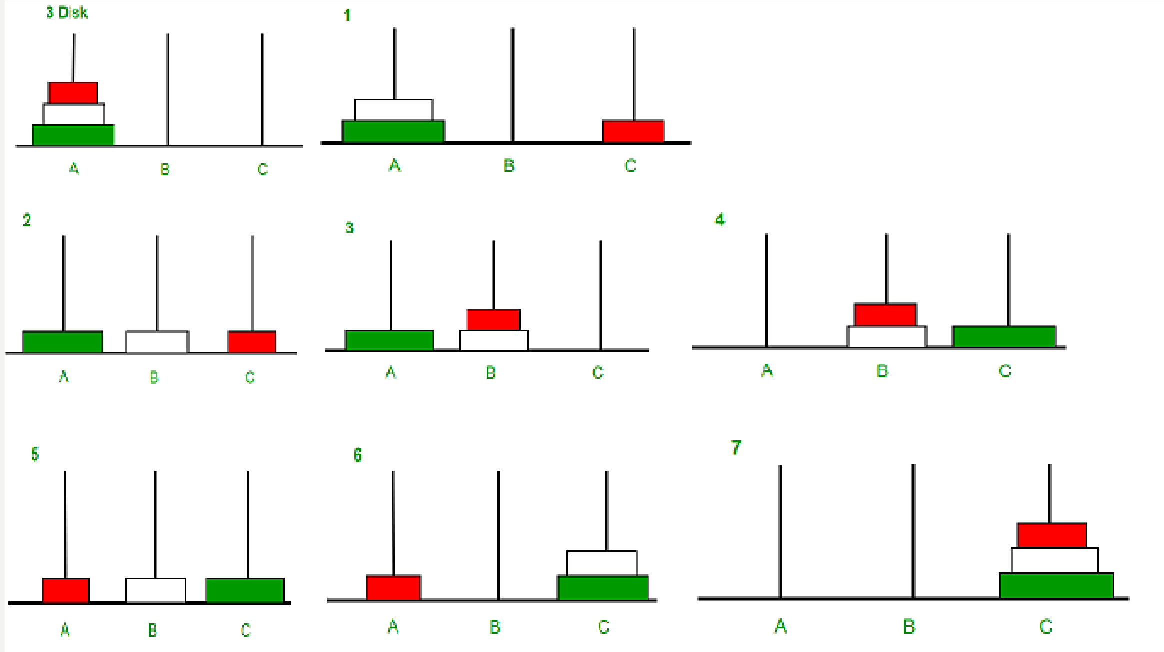
ii) Then, largest disk from peg-1 is moved to peg-3, which requires **1** move.

iii) Finally, (n-1) disks of peg-2 are moved to peg-3 with the help of peg-1 & peg-2.
This requires further H_{n-1} moves.

Hence, We can define a recurrence relation as:

$$H_n = H_{n-1} + 1 + H_{n-1}$$

$$H_n = 2H_{n-1} + 1$$



TOWER OF HANOI:

Now,

$H_n = 2H_{n-1} + 1$

$= 2[\mathbf{2H_{n-2} + 1}] + 1$

$= 2^2H_{n-2} + 2 + 1$

$= 2^2[\mathbf{2H_{n-3} + 1}] + 2 + 1$

$= 2^3H_{n-3} + 2^2 + 2^1 + 2^0$

.

.

$= 2^{n-1}H_1 + 2^{n-2} + \ldots\ldots\ldots\ldots + 2^2 + 2^1 + 2^0$

$= 2^{n-1} + 2^{n-2} + \ldots\ldots\ldots\ldots + 2^2 + 2^1 + 2^0$

This is a Geometric series with common ratio(r) = $(2^{n-1} / 2^{n-2}) = 2$

The sum can be calculated as,

$$S_n = \frac{a[rn - 1]}{r-1}$$

$$= \frac{1[2^n - 1]}{2-1}$$

$S_n = 2^n - 1$. This is solution of Tower of Hanoi.

Suppose you deposit Rs1000 at an interest rate of 5% compounded annually. What is the value of investment at the end of 4 years?

Solution:

Here, initial investment is(I_0) = Rs. 1000

At the end of ,

a) Year 1= $\mathbf{I_1} = I_0 + 5\%$ of $I_0 = I_0\,[1+\frac{5}{100}] = 1\mathbf{.05I_0} = 1.05*1000 = 1050$

b) Year 2= $\mathbf{I_2} = I_1 + 5\%$ of $I_1 = I_1\,[1+\frac{5}{100}] = \mathbf{1.05I_1} = 1.05*1050 = 1102.5$

c) Year 3= $\mathbf{I_3} = I_2 + 5\%$ of $I_2 = I_2\,[1+\frac{5}{100}] = \mathbf{1.05I_2} = 1.05*1102.5 = 1157.63$

d) Year 4= $\mathbf{I_4} = I_3 + 5\%$ of $I_3 = I_3\,[1+\frac{5}{100}] = \mathbf{1.05I_3} = 1.05*1157.63 = 1215.51$

RECURRENCE RELATION:

$I_n = I_{n-1} * 1.05$

$$\mathbf{I_n = I_{n-1}\,[1+\frac{r}{100}]}$$

Deriving **General Solution:**

$I_1 = I_0\,[1+\frac{r}{100}]$

$I_2 = I_1\,[1+\frac{r}{100}] = I_0\,[1+\frac{r}{100}][1+\frac{r}{100}] = I_0\,[1+\frac{r}{100}]^2$

$I_3 = I_2\,[1+\frac{r}{100}] = I_0\,[1+\frac{r}{100}]^2\,[1+\frac{r}{100}] = I_0\,[1+\frac{r}{100}]^3$

$I_4 = I_3\,[1+\frac{r}{100}] = I_0\,[1+\frac{r}{100}]^3\,[1+\frac{r}{100}] = I_0\,[1+\frac{r}{100}]^4$

.

.

.

$$\mathbf{I_n = I_0\,[1+\frac{r}{100}]^n}$$

Q. Suppose that a person invests Rs. 2000 at 14% compounded annually.
a) Find the recurrence relation.
b) Find the initial condition.
c) Find A_1 , A_2, A_3 .
d) Find an explicit formula .
e) How long will it take for a person to double the initial investment?

Solution:

Let, A_n be the amount after n years

Initial investment(A_0) = Rs. 2000

a) Recurrence Relation:

$A_1 = A_0 + 14\%$ of A_0

i.e. $A_1 = (1.14)A_0$

$A_2 = (1.14)\,A_1$

$\mathbf{A_n = (1.14)A_{n-1}}$

b) Initial condition:

Since the initial investment is A_0 = Rs. 2000 . **A_0 = Rs. 2000 is the initial condition.**

c) A_1 , A_2, A_3

i) $A_1 = (1.14)A_0 = 1.14*2000=2280$

ii) $A_2 = (1.14)A_1 = 1.14*2280=2599.2$

iii) $A_3 = (1.14)A_2 = 1.14*2599.2=2963.088$

d) Explicit Formula:

We have,

$A_1 = (1.14)A_0$

$A_2 = (1.14)A_1 = (1.14)^2 A_0$

$A_2 = (1.14)A_2 = (1.14)^3 A_0$

.

.

.

$\mathbf{A_n = (1.14)^n A_0 = (1.14)^n A_{n-1}}$

e)Years to double the investment:

Initial Investment(A_0) = 2000

Final Investment(A_n) = $2A_0$ = 4000

Using the explicit formula,

$A_n = (1.14)^n\ 2000$

$4000 = (1.14)^n\ 2000$

$(1.14)^n = 2$

n = 5.29 Years

Q. A patient is injected with 160ml of a drug. Every 6 hours 25% of the drug passes out of her bloodstream. To compensate, a further 20ml dose is given every 6 hours.

a) Find the recurrence relation for the amount of drug in the bloodstream

b) Use the relation to find the amount of drug remaining after 24 hours.

<u>Solution:</u>

a) Let initial dose=U_0 = 160ml

After 6 hours 25% of drug passes out. So remaining = 75% and every hour 20ml is added

Now,

$U_1 = (0.75)U_0 + 20$

$U_2 = (0.75)U_1 + 20$

$\mathbf{U_n = (0.75)U_{n-1} + 20}$**, is the recurrence relation**

b) Drug remaining after 24 hours:

(After 6 hours) $U_1 = (0.75)U_0 + 20 = 0.75*160 + 20 = 140$

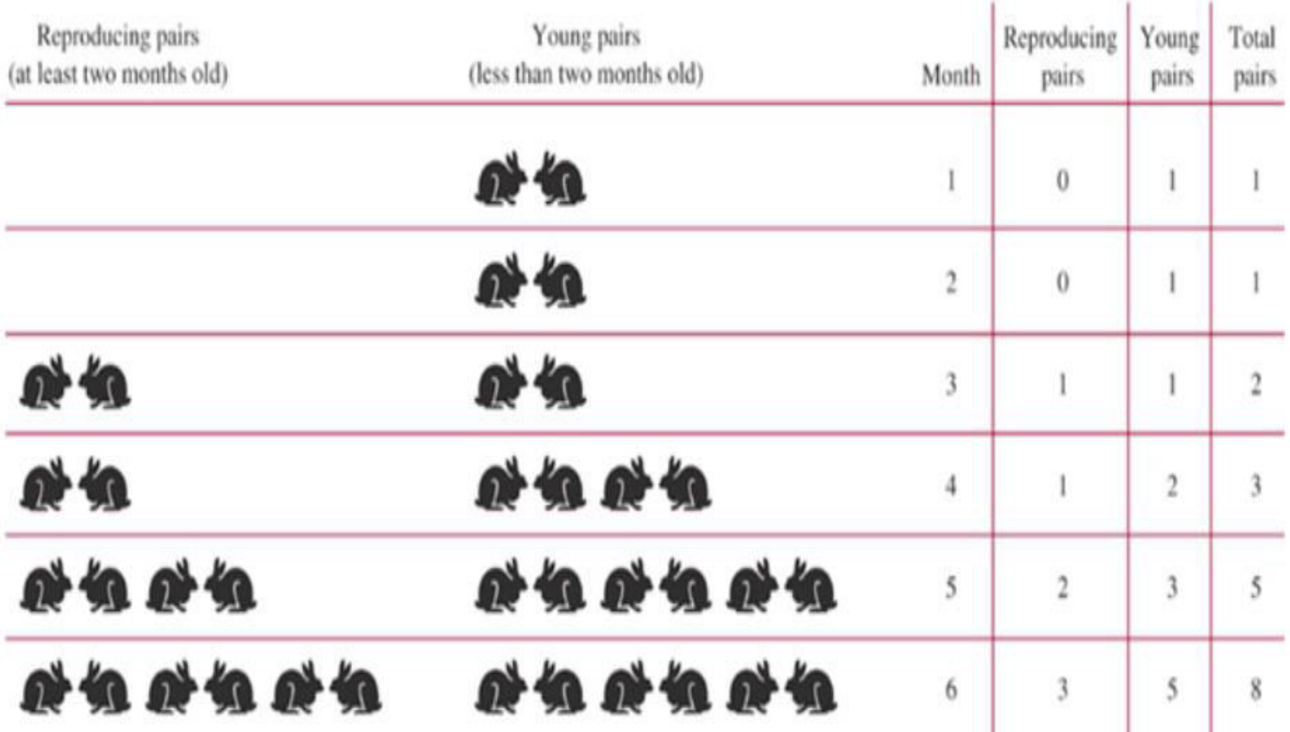
(After 12 hours) $U_2 = (0.75)U_1 + 20 = 0.75*140 + 20 = 125$

(After 18 hours) $U_3 = (0.75)U_2 + 20 = 0.75*125 + 20 = 113.75$

(After 24 hours) $U_4 = (0.75)U_3 + 20 = 0.75*113.75 + 20 = 105.3125$

RABBITS POPULATION:

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 of rabbits produces another pair each month. Assume that none of the rabbits die. How many rabbits are there after n months?



Reproducing pairs (at least two months old)	Young pairs (less than two months old)	Month	Reproducing pairs	Young pairs	Total pairs
		1	0	1	1
		2	0	1	1
		3	1	1	2
		4	1	2	3
		5	2	3	5
		6	3	5	8

FIGURE 1 Rabbits on an Island.

Let f_n denote the number of pairs of rabbits after n months.

$f_1 = 1$ { reproducing pairs = 0, young pairs = 1 }

$f_2 = 1$ { reproducing pairs = 0, young pairs = 1 }

$f_3 = 2$ { reproducing pairs = 1, young pairs = 1 }

$f_4 = 3$ { reproducing pairs = 1, young pairs = 2 }

$f_5 = 5$ { reproducing pairs = 2, young pairs = 3 }

The number of pairs of rabbits after n months fn is equal to the number of pairs of rabbits from the previous month fn-1 plus the number of pairs of newborn rabbits, which equals fn-2, since each newborn pair comes from a pair that is at least two months old, so

$$F_n = F_{n-1} + F_{n-2} \text{ for } n >= 3.$$

An employee joined a company in 2019 with a starting salary of NRs.750000 annually. Every year this employee receives a raise of NRs.50000 plus 3% of the salary of the previous year. Set up a recurrence relation for the salary of the employee after n years from 2019. Find explicit solution. Also find the annual salary of the employee in 2029.

Solution:

Starting Salary(S_0) = Rs. 750000

Raise in salary(R) = 50000 + 0.03S; where S is the salary of previous Year.

Now,

Salary after one year(S_1) = $S_0 + 50000 + 0.03S_0 = 50000 + 1.03S_0$

Salary after two year(S_2) = $S_1 + 50000 + 0.03S_1 = 50000 + 1.03S_1$

Salary after three year(S_3) = $S_2 + 50000 + 0.03S_2 = 50000 + 1.03S_2$

Therefore recurrence relation is given by:

$$\mathbf{S_n = 50000 + 1.03S_{n-1}}$$

$= 50000 + 1.03[50000 + 1.03S_{n-2}]$

$=50000 + (1.03)50000 + (1.03)^2S_{n-2}$

$=50000 + (1.03)50000 + (1.03)^2[50000 + 1.03S_{n-3}]$

$=50000 + (1.03)50000 + (1.03)^2 50000 + (1.03)^3S_{n-3}$

.

.

$=50000 + (1.03)50000 + (1.03)^2 50000 + (1.03)^3 50000 + \ldots\ldots. + (1.03)^{n-1}50000 + (1.03)^n S_0$

.

$=50000 + (1.03)50000 + (1.03)^2 50000 + (1.03)^3 50000 + \ldots\ldots + (1.03)^{n-1} 50000 + (1.03)^n S_0$

$=[50000 + (1.03)50000 + (1.03)^2 50000 + (1.03)^3 50000 + \ldots\ldots + (1.03)^{n-1} 50000] + (1.03)^n S_0$

Using formula for Geometric Sequence,

$$=\frac{a[rn-1]}{r-1}$$; Here common ratio(r) = 1.03, a=50000

$$=\frac{50000[1.03^n-1]}{1.03-1}$$

$$=\frac{50000[1.03^n-1]}{0.03}$$

Therefore,

$$\mathbf{S_n = \frac{50000[1.03^n-1]}{0.03} + (1.03)^n S_0}$$ **is the required solution.**

Now , The salary of the employee in 2029 is:

$$S_{10} = \frac{50000[1.0310-1]}{0.03} + (1.03)^{10} S_0$$

$$= \frac{50000[1.0310-1]}{0.03} + (1.03)^{10} * 750000$$

=Rs. 1,581,131.24

MATHEMATICAL FOUNDATION FOR COMPUTER SCIENCE

Prepared by: Er. Ankit Kharel

Nepal college of information technology

RECURRENCE RELATIONS

- Solution of Linear Homogeneous Recursive Relation

LINEAR HOMOGENEOUS RECURRENCE RELATION:

A linear homogeneous recurrence relation of degree k with constant coefficients is a recurrence relation of the form

$$a_n = c_1 a_{n-1} + c_2 a_{n-2} + \cdot \cdot \cdot + c_k a_{n-k}$$ Where,

$C_1, C_2, \ldots\ldots\ldots C_k$ are called constant coefficient and $C_k \neq 0$.

- **Linear** refers to the Fact that $a_{n-1}, a_{n-2} \ldots\ldots a_{n-k}$ appear in separate terms and to first power.
- **Homogeneous** refers to the fact that the total degree of each term is the same (thus there is no constant term)
- **Constant Coefficients** refers to the fact that $c_1, c_2, \ldots, c_k$ are fixed real numbers that do not depend on n.
- **Degree** k refers to the fact that the expression for an contains the previous k terms $a_{n-1}, a_{n-2}, \ldots, a_{n-k}$
- Examples: $a_n = 5a_{n-1} - 6a_{n-2}$ (Degree 2) , $a_n - 3a_{n-1} + 3a_{n-2} + a_{n-3} = 0$(Degree 3)

SOLUTION OF LINEAR HOMOGENEOUS RECURRENCE RELATION:

Let, $\mathbf{a_n = c_1 a_{n-1} + c_2 a_{n-2} + \cdots + c_k a_{n-k}}$ is a Linear Homogeneous recurrence relation of degree K.

- The basic approach for solving linear homogeneous recurrence relations is to look for solutions of the form $a_n = r^n$, where r is a constant .
- A sequence $a_n = r^n$ is said to be the solution if and only if it satisfies the given recurrence relation.

$$r^n = c_1 r^{n-1} + c_2 r^{n-2} + \ldots + c_k r^{n-k}$$

Dividing Both sides by r^{n-k},

$$\frac{r^n}{r^{n-k}} = \frac{c_1 r^{n-1} + c_2 r^{n-2} + \ldots + c_k r^{n-k}}{r^{n-k}}$$

$$r^k = c_1 r^{k-1} + c_2 r^{k-2} + \ldots + c_k$$

$$r^k - c_1 r^{k-1} - c_2 r k^{-2} - \ldots - ck \qquad \text{(i)}$$

Equation (i) is called the characteristic equation of the Recurrence Relation and The roots are called Characteristics Root.

SOLUTION OF LINEAR HOMOGENEOUS RECURRENCE RELATION:

Find the characteristic equation and roots of the given recurrence relation.

$$a_n = 5a_{n-1} - 6a_{n-2}$$

Solution:

Let, the solution be $a_n = r^n$. Then this solution must satisfy above recurrence relation.

$$r^n = 5r^{n-1} - 6\,r^{n-2}$$

Dividing Both sides by r^{n-2} We get,

$$\frac{r^n}{r^{n-2}} = \frac{5r^{n-1} - 6rn^{-2}}{r^{n-2}}$$

$$r^2 = 5r - 6$$

$r^2 - 5r + 6 = 0$, which is the characteristic equation.

On solving this equation we get,

$r=2$, 3 which is the characteristic roots

SOLUTION OF LINEAR HOMOGENEOUS RECURRENCE RELATION:

THEOREM 1:

Let, $\mathbf{a_n = c_1 a_{n-1} + c_2 a_{n-2}}$ be the linear homogeneous recurrence relation of degree 2 and its characteristics Equation : $r^2 - c_1 r - c_2 = 0$

- If r_1 and r_2 are distinct roots of the characteristics equation then, the solution is of the form,

$$\boxed{a_n = \alpha_1 {r_1}^n + \alpha_2 {r_2}^n}$$

- If r_1 and r_2 are equal roots of the characteristics equation then, the solution is of the form,

$$\boxed{a_n = \alpha_1 r^n + n.\alpha_2 r^n}$$

- If roots of the characteristics equation are complex in the form: $\alpha + i\beta$ then, the solution is of the form

$$\boxed{a_n = [\alpha_1 \cos(n\theta) + \alpha_2 \sin(n\theta)]R^n}$$

$$\mathbf{R} = \sqrt{\alpha^2 + \beta^2}$$

$$\theta = tan^{-1}\left(\frac{\beta}{\alpha}\right)$$

1. What is the solution of the recurrence relation : $a_n = 5a_{n-1} - 6a_{n-2}$ with $a_0=3, a_1=5$.

Solution:

Given recurrence relation:

$$a_n = 5a_{n-1} - 6a_{n-2} \text{ ----------------(i)}$$

The characteristics equation are,

$$r^2 - 5r + 6 = 0$$

$$r^2 - 3r - 2r + 6 = 0$$

$$r(r-3) - 2(r-3) = 0$$

$$(r-2)(r-3) = 0$$

$$r_1=2, r_2=3$$

Since the roots are distinct and real. The solution is in the form:

$$\mathbf{a_n} = \boldsymbol{\alpha_1 r_1{}^n + \alpha_2 r_2{}^n}$$

$$\mathbf{a_n} = \boldsymbol{\alpha_1 2^n + \alpha_2 3^n} \quad \text{------------------(ii)}$$

Now, applying initial condition:

a) $n = 0$, $a_0 = 3$

$a_0 = \alpha_1 2^0 + \alpha_2 3^0$

$\alpha_1 + \alpha_2 = 3$ **---------------(iii)**

b) n = 1 , $a_1 = 5$

$a_1 = \alpha_1 2^1 + \alpha_2 3^1$

$2\alpha_1 + 3\alpha_2 = 3$**----------------(iv)**

Solving equation (iv) and (v), We get,

$\alpha_1 = 4 \ and \ \ \alpha_2 = 1$

Putting the value of α_1 and α_2 in Equation iii, we get

$$\boxed{a_n = 4.2^n - 1.3^n}$$

Which is the required solution.

2. What is the solution of the recurrence relation : $a_n = 6a_{n-1} - 9a_{n-2}$ with $a_0=5, a_1=7$.

Solution:

Given recurrence relation:

$a_n = 6a_{n-1} - 9a_{n-2}$ **----------------(i)**

The characteristics equation are,

$r^2 - 6r + 9 = 0$

$r^2 - 3r - 3r + 9 = 0$

$r(r - 3) - 3(r-3) = 0$

$(r - 3)(r - 3) = 0$

$r_1=3, r_2=3$

Since the roots are equal and real. The solution is in the form:

$$\mathbf{a_n} = \boldsymbol{\alpha_1 r^n + n\alpha_2 r^n}$$

$$\mathbf{a_n} = \boldsymbol{\alpha_1 3^n + n\alpha_2 3^n} \quad \text{------------------(ii)}$$

Now, applying initial condition:

a) $n = 0$, $a_0 = 5$

$a_0 = \alpha_1 3^0 + 0 * \alpha_2 3^0$

$\alpha_1 = 5$**---------------(iii)**

b) n = 1 , a_1 = 7

$a_1 = \alpha_1 3^1 + 1 * \alpha_2 3^1$

$3\alpha_1 + 3\alpha_2 = 7$**----------------(iv)**

Solving equation (iv) and (v), We get,

$$\alpha_1 = 5 \ \ and \ \ \alpha_2 = \frac{-8}{3}$$

Putting the value of α_1 and α_2 in Equation iii, we get

$$\boxed{\mathbf{a_n = 5.3^n - n\frac{8}{3}.3^n}}$$

Which is the required solution.

3.Derive an Explicit Formula For Fibonacci series.

Solution:

The recurrence relation for Fibonacci series is given by:

$a_n = a_{n-1} + a_{n-2}$ with initial condition $a_0=0$ and $a_2 = 1$ **-----------------(i)**

The characteristics equation are,

$r^2 - r - 1 = 0$

The characteristics roots are:

$$r = \frac{-b \pm \sqrt{b^2-4ac}}{2a}$$

$$r = \frac{1 \pm \sqrt{5}}{2}$$

$$r_1 = \frac{1+\sqrt{5}}{2} \text{ and } r_2 = \frac{1-\sqrt{5}}{2}$$

Since the roots are distinct and real. The solution is in the form:

$$\mathbf{a_n} = \boldsymbol{\alpha_1 r_1{}^n + \alpha_2 r_2{}^n}$$

$$\mathbf{a_n} = \boldsymbol{\alpha_1}\left(\frac{1+\sqrt{5}}{2}\right)^n + \boldsymbol{\alpha_2}\left(\frac{1-\sqrt{5}}{2}\right)^n \quad \textbf{------------------(ii)}$$

Now, applying initial condition:

a) $n = 0$, $a_0 = 0$

$$a_0 = \alpha_1 \left(\frac{1+\sqrt{5}}{2}\right)^0 + \alpha_2 \left(\frac{1-\sqrt{5}}{2}\right)^0$$

$$\alpha_1 + \alpha_2 = 0 \text{------------------------------}\textbf{(iii)}$$

b) $n = 1$, $a_1 = 1$

$$a_0 = \alpha_1 \left(\frac{1+\sqrt{5}}{2}\right)^1 + \alpha_2 \left(\frac{1-\sqrt{5}}{2}\right)^1$$

$$\alpha_1 \left(\frac{1+\sqrt{5}}{2}\right)^1 + \alpha_2 \left(\frac{1-\sqrt{5}}{2}\right)^1 = 1 \text{------------------------------}\textbf{(iv)}$$

Solving equation (iv) and (v), We get,

$$\alpha_1 = \frac{1}{\sqrt{5}} and \quad \alpha_2 = \frac{-1}{\sqrt{5}}$$

Putting the value of α_1 and α_2 in Equation iii, we get

$$\boxed{\mathbf{a_n = \frac{1}{\sqrt{5}} \left(\frac{1+\sqrt{5}}{2}\right)^{\mathit{n}} - \frac{1}{\sqrt{5}} (\frac{1-\sqrt{5}}{2})^{\mathit{n}}}}$$

is the required solution.

4. What is the solution of the recurrence relation : $a_n + 2a_{n-1} + 2a_{n-2} = 0$ with $a_0 = 0, a_1 = -1$.

Solution:

Given recurrence relation:

$$a_n + 2a_{n-1} + 2a_{n-2} = 0 \text{ ----------------(i)}$$

The characteristics equation are,

$$r^2 + 2r + 2 = 0$$

$$r = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

$$= \frac{-2 \pm \sqrt{4 - 4.1.2}}{2.1}$$

$$= \frac{-2 \pm \sqrt{-4}}{2}$$

$$= \frac{-2 \pm \sqrt{i^2 4}}{2}$$

$$= \frac{-2 \pm 2i}{2}$$

$$r = -1 \pm i \text{--------------------------}[\alpha + i\beta]$$

Since the roots are imaginary. The solution is in the form:

$$\mathbf{a_n = [\alpha_1 \cos(n\theta) + \alpha_2 \sin(n\theta)]R^n} \text{------------------(ii)}$$

$$\mathbf{R} = \sqrt{\alpha^2 + \beta^2}$$

$$\boldsymbol{\theta = tan^{-1}(\frac{\beta}{\alpha})}$$

$$r = -1 \pm i$$

$$\alpha = -1 \quad , \quad \beta = 1$$

$$\mathbf{R} = \sqrt{\alpha^2 + \beta^2} = \sqrt{(-1)^2 + 1^2} = \sqrt{2}$$

$$\theta = tan^{-1}\left(\frac{1}{-1}\right) = tan^{-1}(-1) = \frac{3\pi}{4}$$

Now, we have

$$a_n = [\alpha_1 \cos\left(n\frac{3\pi}{4}\right) + \alpha_2 \sin\left(n\frac{3\pi}{4}\right)] (\sqrt{2})^n \text{ -------------------(iii)}$$

Using initial condition

a) $n = 0, a_0 = 5$

$$a_0 = [\alpha_1 \cos\left(0 * \frac{3\pi}{4}\right) + \sin\left(0 * \frac{3\pi}{4}\right)] (\sqrt{2})^0$$

$\alpha_1 = 0$

b) $n = 1, a_1 = -1$

$$a_1 = [\alpha_1 \cos\left(1 * \frac{3\pi}{4}\right) + \sin\left(1 * \frac{3\pi}{4}\right)] (\sqrt{2})^1$$

$\alpha_2 = -1$

Putting the value of α_1 , α_2 in equation (iii) , We get

$$\boxed{a_n = -[\sin\left(n\frac{3\pi}{4}\right)] (\sqrt{2})^n}$$

SOLUTION OF LINEAR HOMOGENEOUS RECURRENCE RELATION:

THEOREM 2:

Let, $\mathbf{a_n = c_1 a_{n-1} + c_2 a_{n-2} + c_3 a_{n-3}}$ be the linear homogeneous recurrence relation of degree 3 and its characteristics Equation : $r^3 - c_1 r^2 - c_2 r - c_3 = 0$

- If all roots are distinct then, the solution is of the form,

$$\boxed{a_n = \alpha_1 r_1{}^n + \alpha_2 r_2{}^n + \alpha_3 r_3{}^n}$$

- If two roots are equal then, the solution is of the form,

$$\boxed{a_n = [\alpha_1 + n\alpha_2] r^n + \alpha_3 r_2{}^n}$$

- If all the roots are same ,Then the solution is of the form

$$\boxed{a_n = [\alpha_1 + n\alpha_2 + n^2\alpha_3] r^n}$$

5. What is the solution of the recurrence relation : $a_n = 6a_{n-1} - 11a_{n-2} + 6a_{n-3}$ with $a_0=2$, $a_1=5$, $a_2=15$.

Solution:

Given recurrence relation:

$$a_n = 6a_{n-1} - 11a_{n-2} + 6a_{n-3} \text{-----------------(i)}$$

The characteristics equation are,

$$r^3 - 6r^2 + 11r - 6 = 0$$

On solving above equation, We get,

$$r_1=1, r_2=2 , r_3=3$$

Since **all the roots are distinct.** The solution is in the form:

$$\mathbf{a_n = \alpha_1 r_1{}^n + \alpha_2 r_2{}^n + \alpha_3 r_3{}^n}$$

$$\mathbf{a_n = \alpha_1 1^n + \alpha_2 2^n + \alpha_3 3^n \text{-----------------(ii)}}$$

Now, applying initial condition:

a) $n = 0$, $a_0 = 2$

$a_0 = \alpha_1 1^0 + \alpha_2 2^0 + \alpha_3 3^0$

$\alpha_1 + \alpha_2 + \alpha_3 = 2$ **---------------(iii)**

b) n = 1 , a_1 = 5

$a_1 = \alpha_1 1^1 + \alpha_2 2^1 + \alpha_3 3^1$

$\alpha_1 + 2\alpha_2 + 3\alpha_3 = 5$**----------------(iv)**

c) n = 2, a_2 = 15

$a_2 = \alpha_1 1^2 + \alpha_2 2^2 + \alpha_2 3^2$

$\alpha_1 + 4\alpha_2 + 9\alpha_3 = 15$**-----------------(v)**

Solving equation (iii) , (iv) and (v). We get,

$\alpha_1 = 1$, $\alpha_2 = -1$ and $\alpha_3 = 2$

Putting the value of α_1 , α_2 , α_3 in Equation ii, we get

$$\boxed{\mathbf{a_n = 1 - 2^n + 2.3^n}}$$

Which is the required solution.

6. What is the solution of the recurrence relation : $a_n - 3a_{n-1} + 3a_{n-2} - a_{n-3}$ with $a_0=1, a_1=-2$, $a_2=-1$.

Solution:

Given recurrence relation:

$$a_n - 3a_{n-1} + 3a_{n-2} - a_{n-3} \text{-----------------}(i)$$

The characteristics equation are,

$$r^3 - 3r^2 + 3r - 1 = 0$$

On solving above equation, We get,

$$r_1=1, r_2=1, r_3=1$$

Since all the roots are same. The solution is in the form:

$$a_n = [\alpha_1 + n\alpha_2 + n^2\alpha_3]r^n$$

$$a_n = [\alpha_1 + n\alpha_2 + n^2\alpha_3]1^n \text{-----------------}(ii)$$

Now, applying initial condition:

a) $n = 0, a_0 = 1$

$a_0 = \alpha_1 + 0 * \alpha_2 + 0^2 * \alpha_3$

$\alpha_1 = 1$--(iii)

b) n = 1 , a_1 = -2

$a_1 = \alpha_1 + \alpha_2 + \alpha_3$

$\alpha_1 + \alpha_2 + \alpha_3$ = -2----------------------**(iv)**

c) n = 2, a_2 = -1

$a_2 = \alpha_1 + 2\alpha_2 + 4\alpha_2$

$\alpha_1 + 2\alpha_2 + 4\alpha_3$ = -1-----------------**(v)**

Solving equation (iii) , (iv) and (v). We get,

$$\alpha_1 = \mathbf{1}\ ,\ \alpha_2 = \frac{\mathbf{-11}}{\mathbf{2}}\, and \quad \alpha_3 = \frac{\mathbf{5}}{\mathbf{2}}$$

Putting the value of α_1 , α_2 , α_3 in Equation ii, we get

$$\boxed{\mathbf{a_n = 1 - \frac{11}{2}n + \frac{5}{2}n}}$$

Which is the required solution.

7. What is the solution of the recurrence relation : $a_n = 5a_{n-1} - 7a_{n-2} + 3a_{n-3}$ with $a_0=1, a_1=9$, $a_2=15$.

Solution:

Given recurrence relation:

$$\mathbf{a_n = 5a_{n-1} - 7a_{n-2} + 3a_{n-3}} \text{ ----------------(i)}$$

The characteristics equation are,

$$r^3 - 5r^2 + 7r - 3 = 0$$

On solving above equation, We get,

$$r=1, r=1 , r_2=3$$

Since all the roots are same. The solution is in the form:

$$\mathbf{a_n = [\alpha_1 + n\alpha_2]\, r^n + \alpha_3 r_2^{\,n}}$$

$$\mathbf{a_n = [\alpha_1 + n\alpha_2]\, 1^n + \alpha_3 3^n} \text{ ----------------(ii)}$$

Now, applying initial condition:

a) $n = 0 , a_0 = 1$

$a_0 = \alpha_1 + 0 * \alpha_2 + \alpha_3 * 3^0$

$\alpha_1 + \alpha_3 = 1$ **---(iii)**

b) n = 1 , a_1 = 9

$a_1 = \alpha_1 + \alpha_2 + 3\alpha_3$

$\alpha_1 + \alpha_2 + 3\alpha_3$ **= 9---------------------------(iv)**

c) n = 2, a_2 = -1

$a_2 = \alpha_1 + 2\alpha_2 + 9\alpha_3$

$\alpha_1 + 2\alpha_2 + 9\alpha_3$ = 15**--------------------(v)**

Solving equation (iii) , (iv) and (v). We get,

$$\alpha_1 = \frac{\mathbf{3}}{\mathbf{2}},\ \alpha_2 = 9\ \ and\ \ \ \ \alpha_3 = \frac{\mathbf{-1}}{\mathbf{2}}$$

Putting the value of α_1 , α_2 , α_3 in Equation ii, we get

$$\boxed{\mathbf{a_n = [\frac{3}{2} + 9\boldsymbol{n}] - \frac{1}{2}(3)^n}}$$

Which is the required solution.

MATHEMATICAL FOUNDATION FOR COMPUTER SCIENCE

Prepared by: Er. Ankit Kharel

Nepal college of information technology

RECURRENCE RELATIONS

- Solution of Linear non-Homogeneous Recursive Relation

LINEAR HOMOGENEOUS RECURRENCE RELATION:

A linear non homogeneous recurrence relation of degree k with constant coefficients is a recurrence relation of the form

$$a_n = c_1a_{n-1} + c_2a_{n-2} + \cdot\cdot\cdot + c_ka_{n-k} + f(n)$$ Where,

C_1 , C_2 , ……….C_k are called constant coefficient and $C_k \neq 0$.

Here,

- $a_n = c_1a_{n-1} + c_2a_{n-2} + \cdot\cdot\cdot + c_ka_{n-k}$ is called associated Homogeneous part.
- **f(n)** is called non-homogenous part.

- The solution of associated homogeneous part is called General Solution and the solution of non-homogenous part is called particular solution.

- The final solution of linear non-homogenous recurrence relation is sum of General solution and particular solution.

$$a_n = a_n(h) + a_n(p) \text{-----(i)}$$ where, $a_n(h)$=soluton of homogenous part

$a_n(p)$=solution of non-homogenous part

METHOD TO FIND PARTICULAR SOLUTION:

1) When $\mathbf{f(n) = b^n}$ and b is not the root of the characteristic equation

$$\boxed{a_n(p) = Ab^n}$$

2) When $\mathbf{f(n) = b^n}$ and b is root of the characteristic equation with multiplicity m

$$\boxed{a_n(p) = An^m b^n}$$

1. What is the solution of the recurrence relation : $a_n = 5a_{n-1} - 6a_{n-2} + 7^n$

Solution:

Given recurrence relation:

$$a_n = 5a_{n-1} - 6a_{n-2} + 7^n \text{------------------(i)}$$

The associated homogeneous part is,

$$a_n = 5a_{n-1} - 6a_{n-2}$$

The characteristics equation are,

$$r^2 - 5r + 6 = 0$$

$$r^2 - 3r - 2r + 6 = 0$$

$$r(r-3) - 2(r-3) = 0$$

$$(r-2)(r-3) = 0$$

$$r_1 = 2, r_2 = 3$$

Since the roots are distinct and real. The solution is in the form:

$$\mathbf{a_n(h)} = \alpha_1 r_1^{\,n} + \alpha_2 r_2^{\,n}$$

$$\mathbf{a_n(h)} = \alpha_1 2^n + \alpha_2 3^n \text{ ------------------(ii)}$$

The non-homogeneous part is:

$a_n = 7^n$ [here 7 is not the characteristics root]

So, the solution is of the form:

$$a_n = A7^n$$

Putting value of a_n in equation (i) we get,

$$A7^n = 5\,A7^{n-1} - 6\,A7^{n-2} + 7^n$$

Dividing both sides by 7^n

$$A = \frac{5A}{7} - \frac{6A}{7^2} + 1$$

$$49A = 35A - 6A + 49$$

$$A = \frac{49}{20}$$

Hence the particular solution is:

$$a_n(p) = \frac{49}{20}7^n$$

Therefore, The final solution is:

$$\mathbf{a_n = a_n(h) + a_n(p)}$$

$$\boxed{\mathbf{a_n} = \alpha_1 2^n + \alpha_2 3^n + \frac{49}{20}\mathbf{7^n}}$$

Which is the required solution.

2. What is the solution of the recurrence relation : $2a_n = 3a_{n-1} - a_{n-2} + 2^n$ with initial condition $a_0 = 2$, $a_1 = 3$.

Solution:

Given recurrence relation:

$$2a_n = 3a_{n-1} - a_{n-2} + 2^n \text{ -----------------(i)}$$

The associated homogeneous part is,

$$2a_n = 3a_{n-1} - a_{n-2}$$

The characteristics equation are,

$$2r^2 - 3r + 1 = 0$$

$$2r^2 - 2r - r + 1 = 0$$

$$2r(r-1) - 1(r-1) = 0$$

$$r_1 = \frac{1}{2}, r_2 = 1$$

Since the roots are distinct and real. The solution is in the form:

$$\mathbf{a_n(h)} = \alpha_1 r_1^{\,n} + \alpha_2 r_2^{\,n}$$

$$\mathbf{a_n(h)} = \alpha_1 \left(\frac{1}{2}\right)^n + \alpha_2 1^n \text{ ------------------(ii)}$$

The non-homogeneous part is:

$a_n = 2^n$ [here 2 is not the characteristics root]

So, the solution is of the form:

$$a_n = A2^n$$

Putting value of a_n in equation (i) we get,

$$2A2^n = 3A2^{n-1} - A2^{n-2} + 2^n$$

Dividing both sides by 2^n

$$2A = \frac{3A}{2} - \frac{A}{2^2} + 1$$

$$8A = 6A - A + 4$$

$$A = \frac{4}{3}$$

Hence the particular solution is:

$$a_n(p) = \frac{4}{3}2^n$$

Therefore, The final solution is:

$$\mathbf{a_n = a_n(h) + a_n(p)}$$

$$\mathbf{a_n = \alpha_1\left(\frac{1}{2}\right)^n + \alpha_2 + \frac{4}{3}2^n} \text{ ---(iii)}$$

Using the initial condition:

a) $n = 0$, $a_0 = 2$

$$\mathbf{a_0} = \alpha_1 \left(\frac{1}{2}\right)^0 + \alpha_2 + \frac{4}{3}2^0$$

$$\alpha_1 + \alpha_2 = \frac{2}{3} \text{------------------(iv)}$$

b) $n = 1$, $a_1 = 3$

$$\mathbf{a_1} = \alpha_1 \left(\frac{1}{2}\right)^1 + \alpha_2 + \frac{4}{3}2^1$$

$$\alpha_1 + 2\,\alpha_2 = \frac{2}{3} \quad \text{------------------(v)}$$

Solving equation (iv) and (v), we get,

$$\alpha_2 = 0 \text{ and } \alpha_1 = \frac{2}{3}$$

Putting value of α_1 and α_2 in equation (iii) we get,

$$\boxed{\mathbf{a_n} = \frac{2}{3}\left(\frac{1}{2}\right)^n + \frac{4}{3}2^n}$$

which is required equation

3. What is the solution of the recurrence relation : $a_n - 3a_{n-1} + 2a_{n-2} = 2^n$ with initial condition $a_0 = 0$, $a_1 = 1$.

Solution:

Given recurrence relation:

$$a_n - 3a_{n-1} + 2a_{n-2} = 2^n \text{ -----------------(i)}$$

The associated homogeneous part is,

$$a_n - 3a_{n-1} + 2a_{n-2} = 0$$

The characteristics equation are,

$$r^2 - 3r + 2 = 0$$

$$r^2 - 2r - r + 2 = 0$$

$$r(r - 2) - 1(r-2) = 0$$

$$r_1 = 1 , \ r_2 = 2$$

Since the roots are distinct and real. The solution is in the form:

$$\mathbf{a_n(h)} = \alpha_1 r_1^n + \alpha_2 r_2^n$$

$$\mathbf{a_n(h)} = \alpha_1(1)^n + \alpha_2 2^n \quad \text{-------------------(ii)}$$

The non-homogeneous part is:

$a_n = 2^n$ [here 2 is the characteristics root with multiplicity 1]

So, the solution is of the form:

$$a_n = An2^n$$

Putting value of a_n in equation (i) we get,

$$An2^n - 3A(n-1)2^{n-1} + 2A(n-2)2^{n-2} = 2^n$$

Dividing both sides by 2^n

$$An - \frac{3A(n-1)}{2} + \frac{2A(n-2)}{2^2} = 1$$

$$4An - 6An + 6A + 2An - 4A = 4$$

$$A = 2$$

Hence the particular solution is:

$$a_n(p) = 2n2^n = n2^{n+1}$$

Therefore, The final solution is:

$$\mathbf{a_n = a_n(h) + a_n(p)}$$

$$\mathbf{a_n} = \boldsymbol{\alpha_1}(1)^n + \boldsymbol{\alpha_2}(2)^n + n2^{n+1}\text{------------------------------------}\mathbf{(iii)}$$

Using the initial condition:

a) $n = 0$, $a_0 = 0$

$$a_0 = \alpha_1(1)^0 + \alpha_2 2^0 + 0 * 2^{0+1}$$

$$\alpha_1 + \alpha_2 = 0 \text{------------------(iv)}$$

b) $n = 1$, $a_1 = 2$

$$a_1 = \alpha_1(1)^1 + \alpha_2(2)^1 + 1 * 2^{1+1}$$

$$\alpha_1 + 2\alpha_2 = -2 \text{-----------------(v)}$$

Solving equation (iv) and (v), we get,

$$\alpha_1 = 2 \text{ and } \alpha_2 = -2$$

Putting value of α_1 and α_2 in equation (iii) we get,

$$\boxed{a_n = 2(1)^n - 2(2)^n + n2^{n+1}}$$

which is required equation

4. What is the solution of the recurrence relation : $a_n = 2a_{n-1} + 3^n$ with initial condition $a_1 = 5$.

Solution:

Given recurrence relation:

$$\mathbf{a_n = 2a_{n-1} + 3^n} \text{ -----------------(i)}$$

The associated homogeneous part is,

$$a_n = 2a_{n-1}$$

The characteristics equation are,

$$r - 2 = 0$$

$$r = 2$$

Since the roots is distinct and real. The solution is in the form:

$$\mathbf{a_n(h)} = \boldsymbol{\alpha_1 r^n}$$

$$\mathbf{a_n(h)} = \boldsymbol{\alpha_1}(2)^{\boldsymbol{n}} \quad \text{------------------(ii)}$$

The non-homogeneous part is:

$a_n = 3^n$ [here 3 is not the characteristic root]

So, the solution is of the form:

$$a_n = A3^n$$

Putting value of a_n in equation (i) we get,

$$A3^n = 2A3^{n-1} + 3^n$$

Dividing both sides by 3^n

$$A = \frac{2A}{3} + 1$$

$$A = 3$$

Hence the particular solution is:

$$a_n(p) = 3.3^n = 3^{n+1}$$

Therefore, The final solution is:

$$\mathbf{a_n = a_n(h) + a_n(p)}$$

$$\mathbf{a_n} = \alpha_1(2)^n + \mathbf{3^{n+1}}\text{-----------------------------------}\mathbf{(iii)}$$

Using the initial condition:

a) $n = 1$, $a_1 = 5$

$a_0 = \alpha_1(2)^1 + 3^{1+1}$

$5 = 2\alpha_1 + 9$

$\alpha_1 = -2$

Putting value of $\boldsymbol{\alpha_1}$ in equation (iii) we get,

$$\boxed{\begin{aligned} \mathbf{a_n} &= \mathbf{(-2)}(2)^n + \mathbf{3^{n+1}} \\ \mathbf{a_n} &= \mathbf{3^{n+1} - 2^{n+1}} \end{aligned}}$$

which is required equation

5. What is the solution of the recurrence relation : $a_n - 4a_{n-1} + 4a_{n-2} = 2^n$.

Solution:

Given recurrence relation:

$$a_n - 4a_{n-1} + 4a_{n-2} = 2^n \text{ ------------------(i)}$$

The associated homogeneous part is,

$$a_n - 4a_{n-1} + 4a_{n-2} = 0$$

The characteristics equation are,

$$r^2 - 4r + 4 = 0$$

$$r^2 - 2r - 2r + 4 = 0$$

$$r(r - 2) - 2(r-2) = 0$$

$$r_1 = 2 , \ r_2 = 2$$

Since the roots are equal and real. The solution is in the form:

$$\mathbf{a_n(h)} = \alpha_1 r^n + n\alpha_2 r^n$$

$$\mathbf{a_n(h)} = [\alpha_1 + n\alpha_2]2^n \quad \text{-------------------(ii)}$$

The non-homogeneous part is:

$a_n = 2^n$ [here 2 is the characteristics root with multiplicity 2]

So, the solution is of the form:

$$a_n = An^2 2^n$$

Putting value of a_n in equation (i) we get,

$$An^2 2^n - 4[A(n-1)^2 2^{n-1}] + 4[A(n-2)^2 2^{n-2}] = 2^n$$

Dividing both sides by 2^n

$$An^2 - \frac{4[A(n-1)2]}{2} + \frac{4[A(n-2)2]}{2^2} = 1$$

$$4An^2 - 8[An^2 - 2An + A] + 4[An^2 - 4An + 4A] = 4$$

$$4An^2 - 8An^2 + 16An - 8A + 4An^2 - 16An + 16A = 4$$

$$A = \frac{1}{2}$$

Hence the particular solution is:

$$a_n(p) = \frac{1}{2} n^2 2^n = n^2 2^{n-1}$$

Therefore, The final solution is:

$$\mathbf{a_n = a_n(h) + a_n(p)}$$

$\boxed{\mathbf{a_n} = [\alpha_1 + n\alpha_2](2)^n + n^2 2^{n-1}}$ is the required equation

METHOD TO FIND PARTICULAR SOLUTION:

2) When **f(n) is a polynomial in n**

(i)If f(n) is first degree polynomial in n

$$\boxed{a_n(p) = A_0 + A_1 n}$$

(ii) If f(n) is Second degree polynomial in n

$$\boxed{a_n(p) = A_0 + A_1 n + n^2 A_2}$$

(iii) If f(n) is Third degree polynomial in n

$$\boxed{a_n(p) = A_0 + A_1 n + n^2 A_2 + n^3 A_3}$$

6. What is the solution of the recurrence relation : $a_n - 5a_{n-1} + 6a_{n-2} = 2 + n$ with initial condition $a_0 = 1$, $a_1 = 1$.

Solution:

Given recurrence relation:

$$a_n - 5a_{n-1} + 6a_{n-2} = 2 + n \text{ -----------------}\mathbf{(i)}$$

The associated homogeneous part is,

$$a_n - 5a_{n-1} + 6a_{n-2} = 0$$

The characteristics equation are,

$$r^2 - 5r + 6 = 0$$

$$r^2 - 3r - 2r + 6 = 0$$

$$r(r - 3) - 2(r - 3) = 0$$

$$r_1 = 2 , \ r_2 = 3$$

Since the roots are distinct and real. The solution is in the form:

$$\mathbf{a_n(h)} = \alpha_1 r_1^n + \alpha_2 r_2^n$$

$$\mathbf{a_n(h)} = \alpha_1 2^n + \alpha_2 3^n \text{ -------------------}\mathbf{(ii)}$$

The non-homogeneous part is:

$a_n = 2 + n$ [here f(n) is a polynomial degree of 1]

So, the solution is of the form:

$$a_n = A_0 + A_1 n$$

Putting value of a_n in equation (i) we get,

$$[A_0 + A_1 n] – 5[A_0 + A_1(n-1)] + 6[A_0 + A_1(n-2)] = 2 + n$$

$$[A_0 + A_1 n] – [5A_0 + 5A_1 n – 5A_1] + [6A_0 + 6A_1 n - 12A_1)] = 2 + n$$

$$A_0 - 5A_0 + 6A_0 + 5A_1 - 12A_1 + A_1 n - 5A_1 n + 6A_1 n = 2 + n$$

$$\mathbf{[2A_0 - 7A_1]} + \mathbf{2A_1}n = 2 + n$$

Equating coefficient of n^0 and n^1, we get,

$$2A_0 - 7A_1 = 2 \text{ ----------------------------}\mathbf{(iii)}$$

$$2A_1 = 1 \text{-----------------------------}\mathbf{(iv)}$$

Solving equation (iii) and (iv), we get,

$$A_0 = \frac{11}{4} \text{ and } A_1 = \frac{1}{2}$$

Hence the particular solution is:

$$\mathbf{a_n(p) = \frac{11}{4} + \frac{1}{2}n}$$

Therefore, The final solution is:

$$a_n = a_n(h) + a_n(p)$$

$$a_n = \alpha_1 2^n + \alpha_2 3^n + \frac{11}{4} + \frac{1}{2}n \text{ ---------------------------------------(v)}$$

Using the initial condition:

a) $n = 0$, $a_0 = 1$

$$a_0 = \alpha_1 2^0 + \alpha_2 3^0 + \frac{11}{4} + \frac{1}{2}0$$

$$\alpha_1 + \alpha_2 = \frac{-7}{4} \text{ -----------------(iv)}$$

b) $n = 1$, $a_1 = 3$

$$a_1 = \alpha_1 2^1 + \alpha_2 3^1 + \frac{11}{4} + \frac{1}{2}1$$

$$2\alpha_1 + 3\alpha_2 = \frac{-9}{4} \text{ ----------(v)}$$

Solving equation (iv) and (v), we get,

$$\alpha_1 = -3 \text{ and } \alpha_2 = \frac{5}{4}$$

Putting value of α_1 and α_2 in equation (v) we get,

$$\boxed{a_n = (-3)2^n + \frac{5}{4}3^n + \frac{11}{4} + \frac{1}{2}n}$$

7. What is the solution of the recurrence relation : $a_n - 4a_{n-1} + 4a_{n-2} = (1 + n)^2$.

Solution:

Given recurrence relation:

$$\mathbf{a_n - 4a_{n-1} + 4a_{n-2} = (1 + n)^2} \text{ ----------------(i)}$$

The associated homogeneous part is,

$$a_n - 4a_{n-1} + 4a_{n-2} = 0$$

The characteristics equation are,

$$r^2 - 4r + 4 = 0$$

$$r^2 - 2r - 2r + 4 = 0$$

$$r(r - 2) - 2(r - 2) = 0$$

$$r_1 = 2 ,\ r_2 = 2$$

Since the roots are equal and real. The solution is in the form:

$$\mathbf{a_n(h) = \alpha_1 r^n + n\alpha_2 r^n}$$

$$\mathbf{a_n(h) = \alpha_1 2^n + \alpha_2 3^n} \text{ ------------------(ii)}$$

The non-homogeneous part is:

$a_n = (1 + n)^2$ [Here f(n) is a polynomial degree of 2]

So, the solution is of the form:

$$a_n = A_0 + A_1 n + A_2 n^2$$

Putting value of a_n in equation (i) we get,

$$[A_0 + A_1 n + A_2 n^2] - 4[A_0 + A_1(n-1) + A_2(n-1)^2] + 4[A_0 + A_1(n-2) + A_2(n-2)^2] = (1+n)^2$$

$$\mathbf{[A_0 - 4A_1 + 12A_2]} + \mathbf{[A_1 - 8A_2]}n + \mathbf{A_2}n^2 = 1 + 2n + n^2$$

Equating coefficient of n^0, n^1, n^2, we get,

$$A_0 - 4A_1 + 12A_2 = 1 \text{ ---------------------------}\mathbf{(iii)}$$

$$A_1 - 8A_2 = 2\text{----------------------------}\mathbf{(iv)}$$

$$A_2 = 1\text{------------------------------------}\mathbf{(v)}$$

Solving equation (iii) , (iv) , (v) , we get,

$$A_0 = 29 \quad , \quad A_1 = 10 \quad , \quad A_2 = 1$$

Hence the particular solution is:

$$\mathbf{a_n(p) =} 29 + 10n + n^2$$

Total solution is:

$$\mathbf{a_n = a_n(h) + a_n(p)}$$

$\mathbf{a_n(h)} = \alpha_1 2^n + \alpha_2 3^n \mathbf{+ 29 + 10n + n^2}$ **,** is the required equation

METHOD TO FIND PARTICULAR SOLUTION:

3) When **f(n) is a constant i.e f(n) = k**
   We can write $\mathbf{f(n) = k.(1)^n}$
   (i)If 1 is not the characteristic root then,

$$\boxed{a_n(p) = A_0}$$

   (ii) If 1 is the characteristic root with multiplicity m then,

$$\boxed{a_n(p) = A_0 n^m}$$

8. What is the solution of the recurrence relation : $a_n - 2a_{n-1} + a_{n-2} = 7$.

Solution:

Given recurrence relation:

$$\mathbf{a_n - 2a_{n-1} + a_{n-2} = 7} \text{ ----------------(i)}$$

The associated homogeneous part is,

$$a_n - 2a_{n-1} + a_{n-2} = 0$$

The characteristics equation are,

$$r^2 - 2r + 1 = 0$$

$$r^2 - r - r + 1 = 0$$

$$r(r - 1) - r(r - 1) = 0$$

$$r_1 = 1 \text{ , } r_2 = 1$$

Since the roots are equal and real. The solution is in the form:

$$\mathbf{a_n(h) = \alpha_1 r^n + n\alpha_2 r^n}$$

$$\mathbf{a_n(h) = \alpha_1 1^n + \alpha_2 1^n} \text{ -----------------(ii)}$$

The non-homogeneous part is:

$a_n = 7(1)^n$ [Here 1 is the characteristic root with multiplicity 2]

So, the solution is of the form:

$$a_n = A_0 n^2$$

Putting value of a_n in equation (i) we get,

$$A_0 n^2 - 2A_0(n-1)^2 + A_0 (n-2)^2 = 7$$

$$A_0 = \frac{7}{2}$$

Hence the particular solution is:

$$\mathbf{a_n(p)} = \frac{7}{2} n^2$$

Total solution is:

$$\mathbf{a_n = a_n(h) + a_n(p)}$$

$\boxed{\mathbf{a_n(h)} = \alpha_1 + \alpha_2 + \frac{7}{2} n^2}$ is the required equation

METHOD TO FIND PARTICULAR SOLUTION:

4. When **f(n) = b^n** $\emptyset(n)$ where **b** is constant and $\emptyset(n)$ is polynomial in n of degree k.

i) If **b** is not the root of CE. Then the particular solution is,

$$\boxed{a_n(p) = b^n[A_0 + A_1 n + A_2 n^2 + A_3 n^3 + \ldots\ldots\ldots + A_{t-1} n^t]}$$

ii) If **b** is the root of CE with multiplicity **m**. Then the particular solution is,

$$\boxed{a_n(p) = n^m b^n[A_0 + A_1 n + A_2 n^2 + A_3 n^3 + \ldots\ldots\ldots + A_{t-1} n^t]}$$

9. What is the solution of the recurrence relation : $a_n + a_{n-1} = 3n2^n$.

Solution:

Given recurrence relation:

$$\mathbf{a_n + a_{n-1} = 3n2^n} \text{ ----------------(i)}$$

The associated homogeneous part is,

$$a_n + a_{n-1} = 0$$

The characteristics equation are,

$$r + 1 = 0$$

$$r = -1$$

Since the root is real. The solution is in the form:

$$\mathbf{a_n(h)} = \boldsymbol{\alpha_1 r^n}$$

$$\mathbf{a_n(h)} = \boldsymbol{\alpha_1}(-\mathbf{1})^n \text{ -----------------(ii)}$$

The non-homogeneous part is:

$a_n = 3n2^n$ [Here 2 is not the characteristic root and degree of polynomial is one]

So, the solution is of the form:

$$a_n = 2^n[A_0 + A_1 n]$$

Putting value of a_n in equation (i) we get,

$$2^n[A_0 + A_1 n] + 2^{n-1}[A_0 + A_1(n-1)] = 3n2^n$$

$$2^n[A_0 + A_1 n + \frac{A_0}{2} + \frac{A_0 n}{2} - \frac{A_1}{2}] = 3n2^n$$

$$[\frac{3A_0}{2} - \frac{A_1}{2}] + \frac{3A_1}{2}n = 3n$$

Equating on both sides we get,

$$\frac{3A_0}{2} - \frac{A_1}{2} = 0 \text{-------------------------} \textbf{(iii)}$$

$$\frac{3A_1}{2} = 3 \text{-------------------------} \textbf{(iv)}$$

Solving equation (iii) and (iv) we get,

$$A_0 = \frac{2}{3} \text{ and } A_1 = 2$$

Hence the Total solution is:

$$\boxed{\mathbf{a_n = \alpha_1(-1)^n + 2^n[\frac{2}{3} + 2n]}}$$

METHOD TO FIND PARTICULAR SOLUTION:

5. When **f(n) = b^n +** $\emptyset(n)$ where **b** is constant and $\emptyset(n)$ is polynomial in n of degree t.

i) If **b** is not the root of CE. Then the particular solution is,

$$\boxed{a_n(p) = A_0 b^n + A_1 + A_2 n + A_3 n^2 + A_4 n^3 + \ldots\ldots\ldots\ldots + A_{t-1} n^t}$$

ii) If **b** is the root of CE with multiplicity **m** . Then the particular solution is,

$$\boxed{a_n(p) = A_0 n^m b^n + A_1 + A_2 n + A_3 n^2 + A_4 n^3 + \ldots\ldots\ldots\ldots + A_{t-1} n^t}$$

9. What is the solution of the recurrence relation : $a_n - 5a_{n-1} + 6a_{n-2} = 2^n + n$.

Solution:

Given recurrence relation:

$$\mathbf{a_n - 5a_{n-1} + 6a_{n-2} = 2^n + n} \text{ ----------------(i)}$$

The associated homogeneous part is,

$$a_n - 5a_{n-1} + 6a_{n-2} = 0$$

The characteristics equation are,

$$r^2 - 5r + 6 = 0$$

$$r^2 - 3r - 2r + 6 = 0$$

$$(r - 2)(r - 3) = 0$$

$$r_1 = 2 \text{ and } r_2 = 3$$

Since the root are distinct real. The solution is in the form:

$$\mathbf{a_n(h) = \alpha_1 r_1^n + \alpha_1 r_2^n}$$

$$\mathbf{a_n(h) = \alpha_1 2^n + \alpha_1 3^n} \text{ ------------------(ii)}$$

The non-homogeneous part is:

$a_n = n + 2^n$ [Here 2 is the characteristic root with multiplicity 1 and degree of polynomial is one]

So, the solution is of the form:

$$a_n = A_0 n2^n + A_1 + A_2 n$$

Putting value of a_n in equation (i) we get,

$$[A_0 n2^n + A_1 + A_2 n] - 5[A_0(n-1)2^{n-1} + A_1 + A_2(n-1)] + 6[A_0(n-2)2^{n-2} + A_1 + A_2(n-2)] = 2^n + n$$

$$-\frac{1}{2}A_0 2^n + 2A_2 n + (2A_1 - 7A_2) = 2^n + n$$

Equating on both sides we get,

$$A_0 = -2 \ , \quad A_2 = \frac{1}{2} \ , \quad A_1 = \frac{7}{4}$$

Therefore particular solution is,

$$a_n(p) = -2n2^n + \frac{7}{4} + \frac{1}{2}n$$

Hence the Total solution is:

$$\mathbf{a_n = a_n(h) + a_n(p)}$$

$$\boxed{\mathbf{a_n} = \alpha_1 2^n + \alpha_1 3^n - \mathbf{n2^{n+1}} + \frac{1}{2}\mathbf{n} + \frac{7}{4}}$$

MATHEMATICAL FOUNDATION FOR COMPUTER SCIENCE

Prepared by: Er. Ankit Kharel

Nepal college of information technology

FINITE STATE AUTOMATA

- *Sequential Circuits and Finite state Machine*
- *Finite State Automata*
- *Non-deterministic Finite State Automata*
- *Language and Grammars*
- *Language and Automata*
- *Regular Expression*

COMBINATIONAL CIRCUITS:

- Combination circuit is a circuit where the output depends only on the present value of input. Combinatorial circuits can be constructed using solid-state devices, called gates.

- A combinatorial circuit has no memory; previous inputs and the state of the system do not affect the output of a combinatorial circuit.

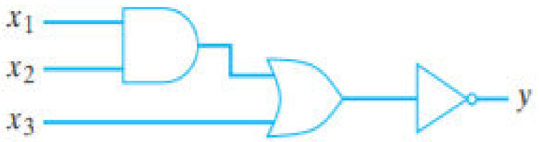


Figure 11.1.4 A combinatorial circuit.

The logic table for this combinatorial circuit follows.

x_1	x_2	x_3	y
1	1	1	0
1	1	0	0
1	0	1	0
1	0	0	1
0	1	1	0
0	1	0	1
0	0	1	0
0	0	0	1

SEQUENTIAL CIRCUITS:

- Sequential circuit is a circuit where the output depends on the present value of input as well as the sequence of past input.
- A Sequential circuit is a combination of combinational circuit and a storage element.
- The sequential circuits use current input variables and previous input variables which are stored and provides the data to the circuit on the next clock cycle.

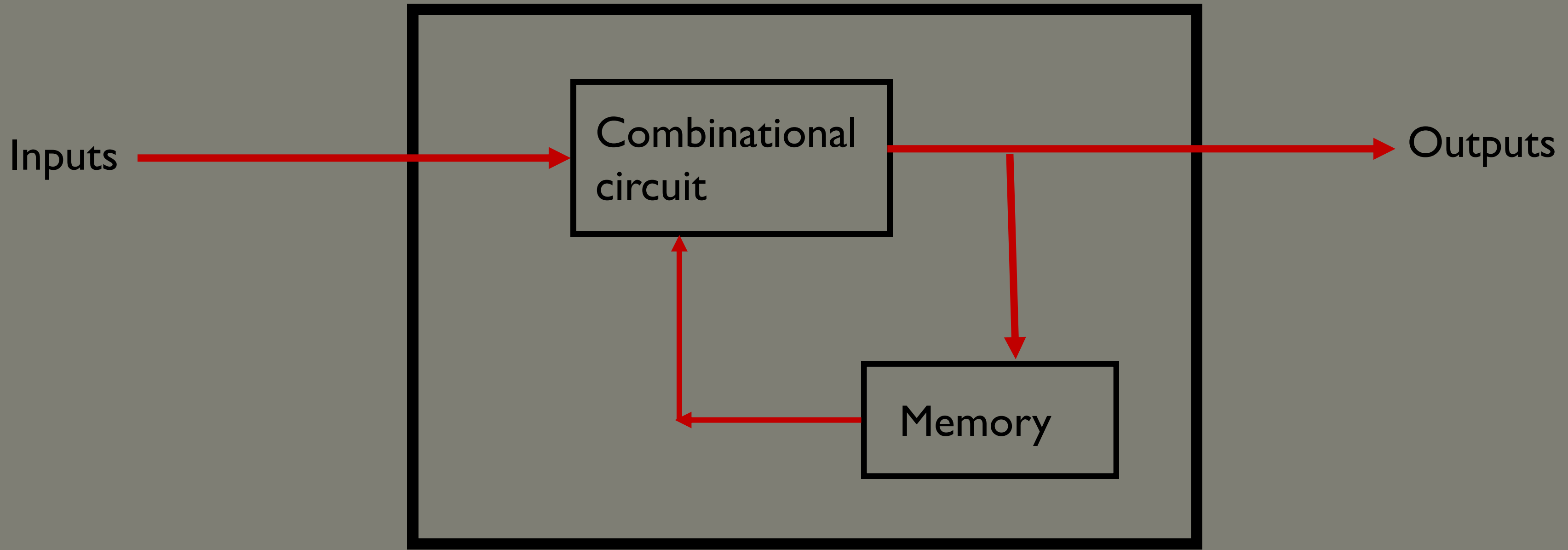


Fig. Sequential circuit

SEQUENTIAL CIRCUITS:

- We will assume that the state of the system changes only at time $t = 0, 1, \ldots$. A simple way to introduce sequencing in circuits is to introduce a **unit time delay**.
- A unit time delay accepts as input a bit x_t at time t and outputs x_{t-1}, the bit received as input at time $t - 1$.
- The sequential circuits use current input variables and previous input variables which are stored and provides the data to the circuit on the next clock cycle.

Fig. Unit time delay

- As an example of the use of the unit time delay, we discuss the **serial adder.**

SERIAL ADDER:

- A serial adder accepts as input two binary numbers
$$x = 0x_N x_{N-1} \cdots x_0 \text{ and } y = 0y_N y_{N-1} \cdots y_0$$ and output s the sum as:
$$Z = z_{N+1} z_N \cdots z_0$$

Example:

$$\begin{array}{rl} x = 1011 & (x_3x_2x_1x_0) \\ + \; y = 0101 & (y_3y_2y_1y_0) \\ \hline z = 10000 & (z_4z_3z_2z_1z_0) \end{array}$$

- The numbers x and y are input sequentially in pairs, x0, y0;...; x_N, y_N; . The sum is output $z_0, z_1, \ldots, z_{N+1}$.

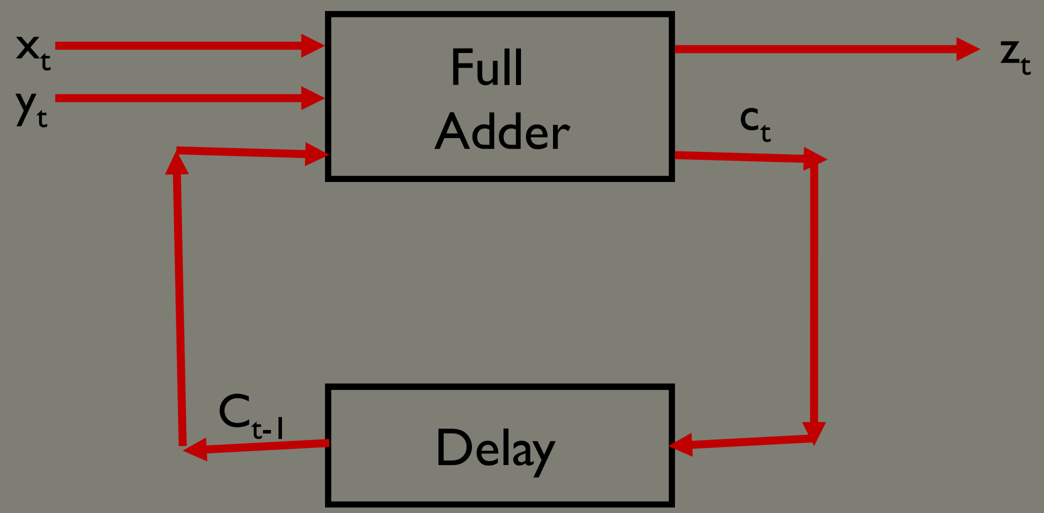


Fig. Serial adder circuit

SERIAL ADDER:

- Addition of x = 010 , y = 011.

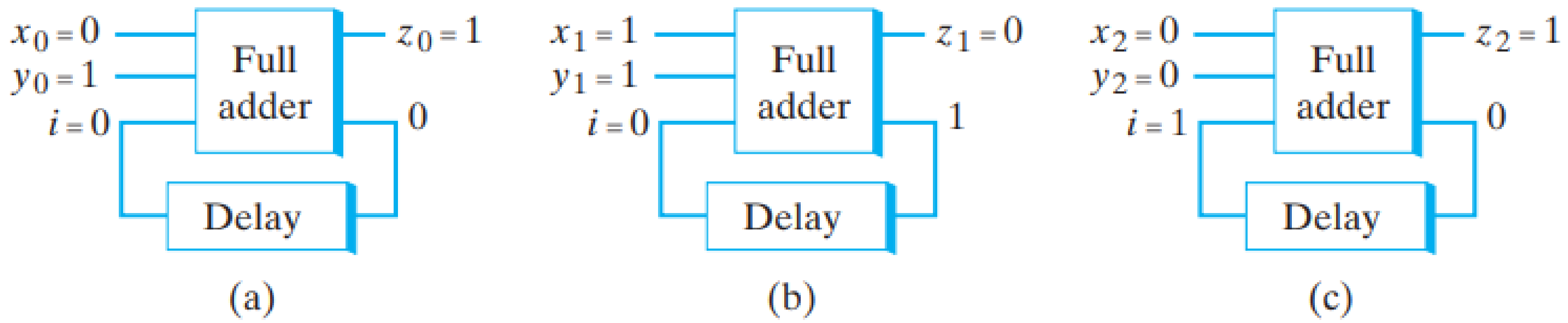


Figure 12.1.3 Computing 010 + 011 with the serial-adder circuit.

- First, we give input as x_0 = 0 and y_0 = 1 to the full adder. The adder computes z_0= (0+1)1 and sends 0 as carry to the delay.
- Second, we give input as x_1 = 1 and y_1 = 1 to the full adder. Now, the delay sends the input 0 and the adder sums z_1 = (1+1+0) 0 and carry 1 is send to the delay.
- Third, we give input as x_2 = 0 and y_2 = 0 to the full adder. Now, the delay sends the input 1 and the adder sums z_1 = (0+0+1) 1 and carry 0 is send to the delay.
- Finally , the output is : z=x+y=101

FINITE STATE MACHINE:

- A finite state machine is a model of computation based on a hypothetical machine made of one or more states. It is used to simulate sequential logic and some computer program.

- We have a fixed set of states that machine can be in.

- The machine can only be in one state at a time.

- Every state has a set of transition and every transition is associated with an input and pointing to a state

Example:

TRAFFIC LIGHT:

A simple traffic light system can be modeled with a finite state machine. Let's look at each core component and identify what it would be for traffic light.

a) **States:** A traffic light generally has three sates: RED, GREEN and YELLOW.

b) **Initial state:** GREEN(suppose)

c) **Accepting state:** In real world traffic lights run indefinitely, so there would be no accepting sate for this.

FINITE STATE MACHINE:

Example:

TRAFFIC LIGHT:

A simple traffic light system can be modeled with a finite state machine. Let's look at each core component and identify what it would be for traffic light.

a) **States:** A traffic light generally has three sates: RED, GREEN and YELLOW.

b) **Initial state:** GREEN(suppose)

c) **Accepting state:** In real world traffic lights run indefinitely, so there would be no accepting sate for this.

d) **Alphabets:** Positive integer representing seconds

e) **Transition:**

- If we are in state **GREEN**, wait for 360s and then go to state **YELLOW**
- If we are in state **YELLOW**, wait for 10s and then go to state **RED**
- If we are in state **RED**, wait for 120s and then go to state **GREEN**

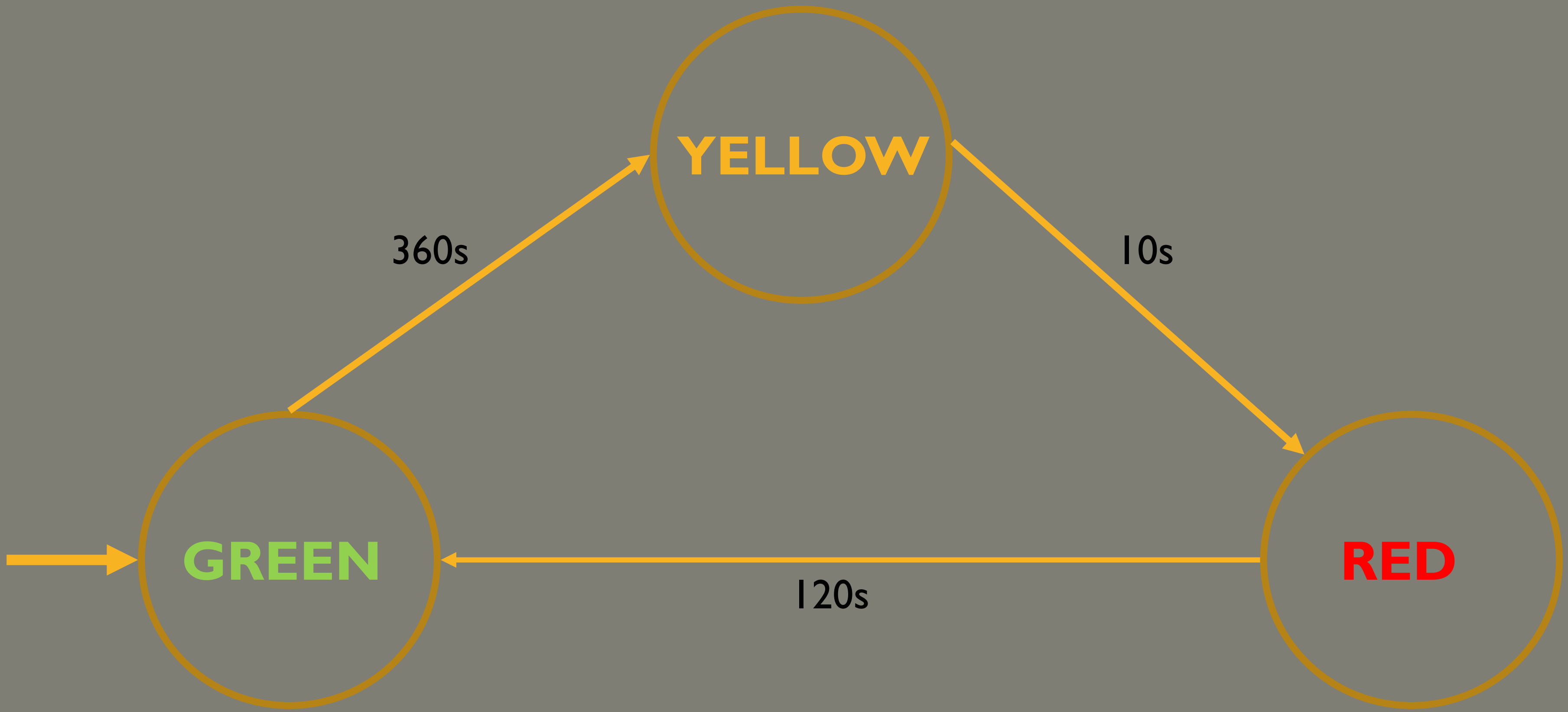


Fig: Transition diagram

FINITE STATE MACHINE:

Formal Definition of Finite State Machine(FSM)

A *finite-state machine* is an abstract model of a machine with a primitive internal memory. A finite-state machine M consists of

(a) A finite set **I** of input symbols.
(b) A finite set **O** of output symbols.
(c) A finite set **S** of states.
(d) A next-state function **f** from $S \times I$ into S.
(e) An output function **g** from $S \times I$ into O.
(f) An initial state $\boldsymbol{\sigma} \in S$.

We write $M = (I, O, S, f, g, \sigma)$

FINITE STATE MACHINE:

Let, I = {a, b}, O = {0, 1}, and S = $\{\sigma_0, \sigma_1\}$. Define the pair of functions f : S×I → S and g: S × I → O by the rules given in Table.

$S \backslash I$	f: a	f: b	g: a	g: b
σ_0	σ_0	σ_1	0	1
σ_1	σ_1	σ_1	1	0

Solution:

(a) State Transition Function(STF):

$f(\sigma_0, a) = \sigma_0$
$f(\sigma_0, b) = \sigma_1$
$f(\sigma_1, a) = \sigma_1$
$f(\sigma_1, b) = \sigma_1$

(b) Machine Function(MF)

$g(\sigma_0, a) = 0$
$g(\sigma_0, b) = 1$
$g(\sigma_1, a) = 1$
$g(\sigma_1, b) = 0$.

The next state Function and Output function can also be defined by the Transition Diagram.

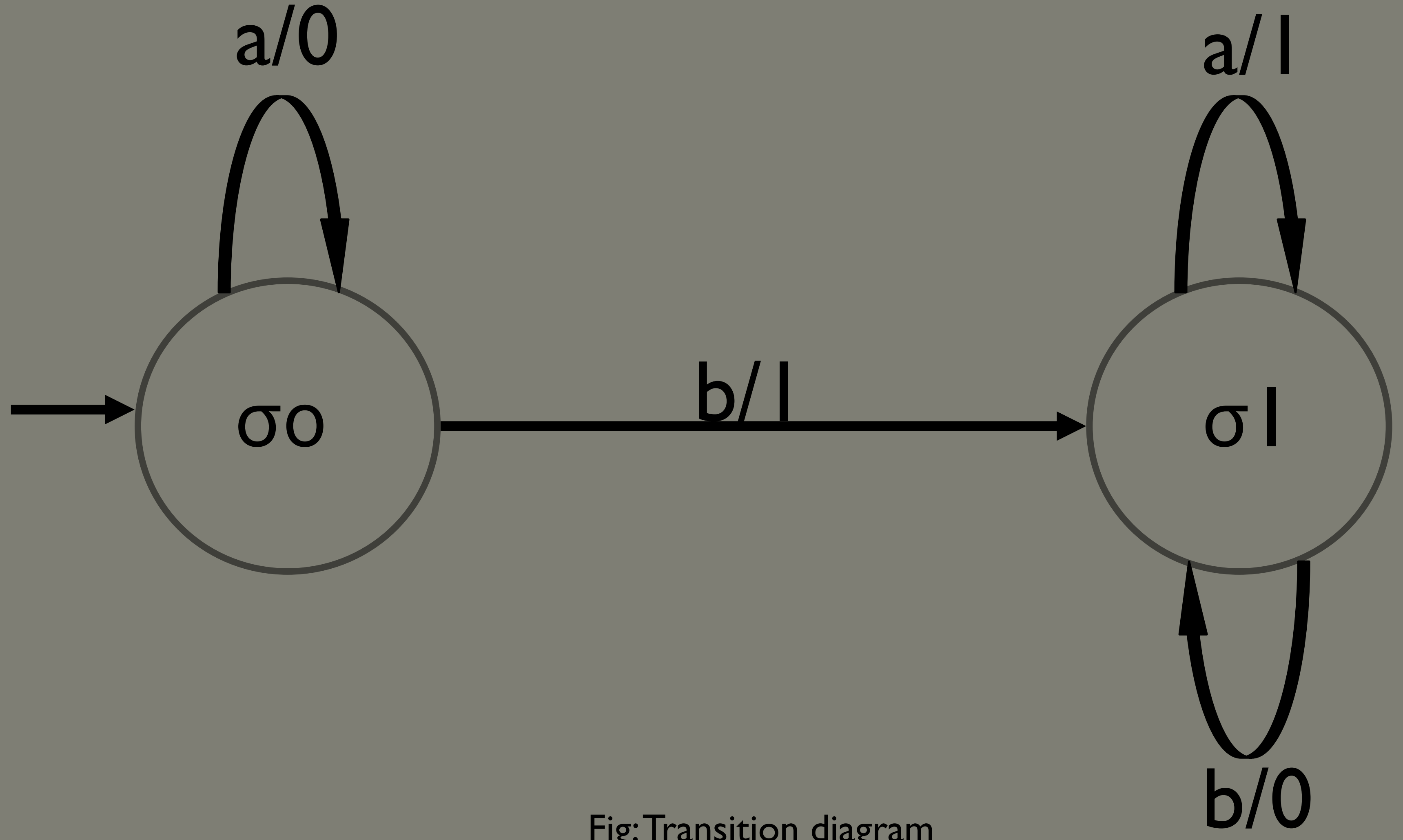


Fig: Transition diagram

Definition: Let M = (I, O, S, f, g, σ) be a finite-state machine. The transition diagram of M is a digraph G whose vertices are the members of S. An arrow designates the initial state σ. A directed edge (σ1, σ2) exists in G if there exists an input i with f(σ1, i) = σ2. In this case, if g(σ1, i) = o, the edge (σ1, σ2) is labeled i/o

FINITE STATE MACHINE:

Find the output string corresponding to the input string "aababba" for below finite-state machine.

	f		g	
$\mathcal{S}$ \ $\mathcal{I}$	a	b	a	b
σ_0	σ_0	σ_1	0	1
σ_1	σ_1	σ_1	1	0

Solution:

Initial State	Input	Output State	Output
σ_0	a	σ_0	0
σ_0	a	σ_0	0
σ_0	b	σ_1	1
σ_1	a	σ_1	1
σ_1	b	σ_1	0
σ_1	b	σ_1	0
σ_1	a	σ_1	1

The Output is:

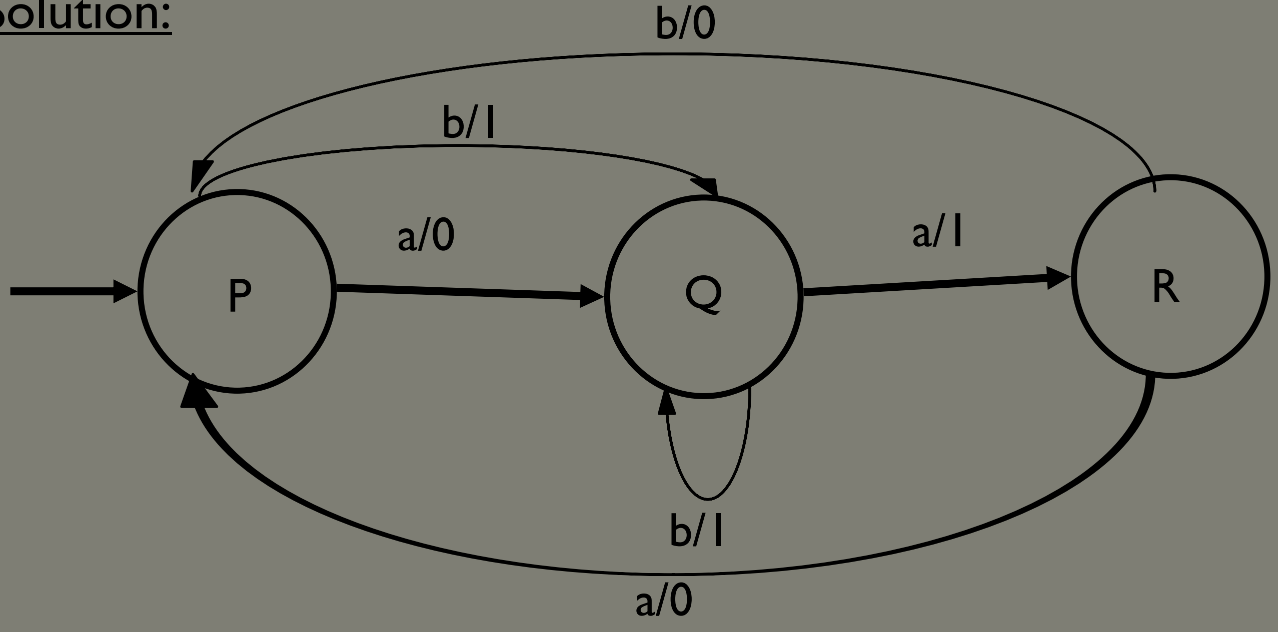
0011001

FINITE STATE MACHINE:

Draw the transition diagram of the finite state machine M, where, I = {a, b} O = {0, 1} S = {P, Q, R} σ = P and transition given by below table. Find the output string corresponding to the input string "aabbaba"

	f		g	
S/I	a	b	a	b
P	Q	Q	0	1
Q	R	Q	1	1
R	P	P	0	0

Solution:



Initial State	Input	Output State	Output
P	a	Q	0
Q	a	R	1
R	b	P	0
P	b	Q	1
Q	a	R	1
R	b	P	0
P	a	Q	0

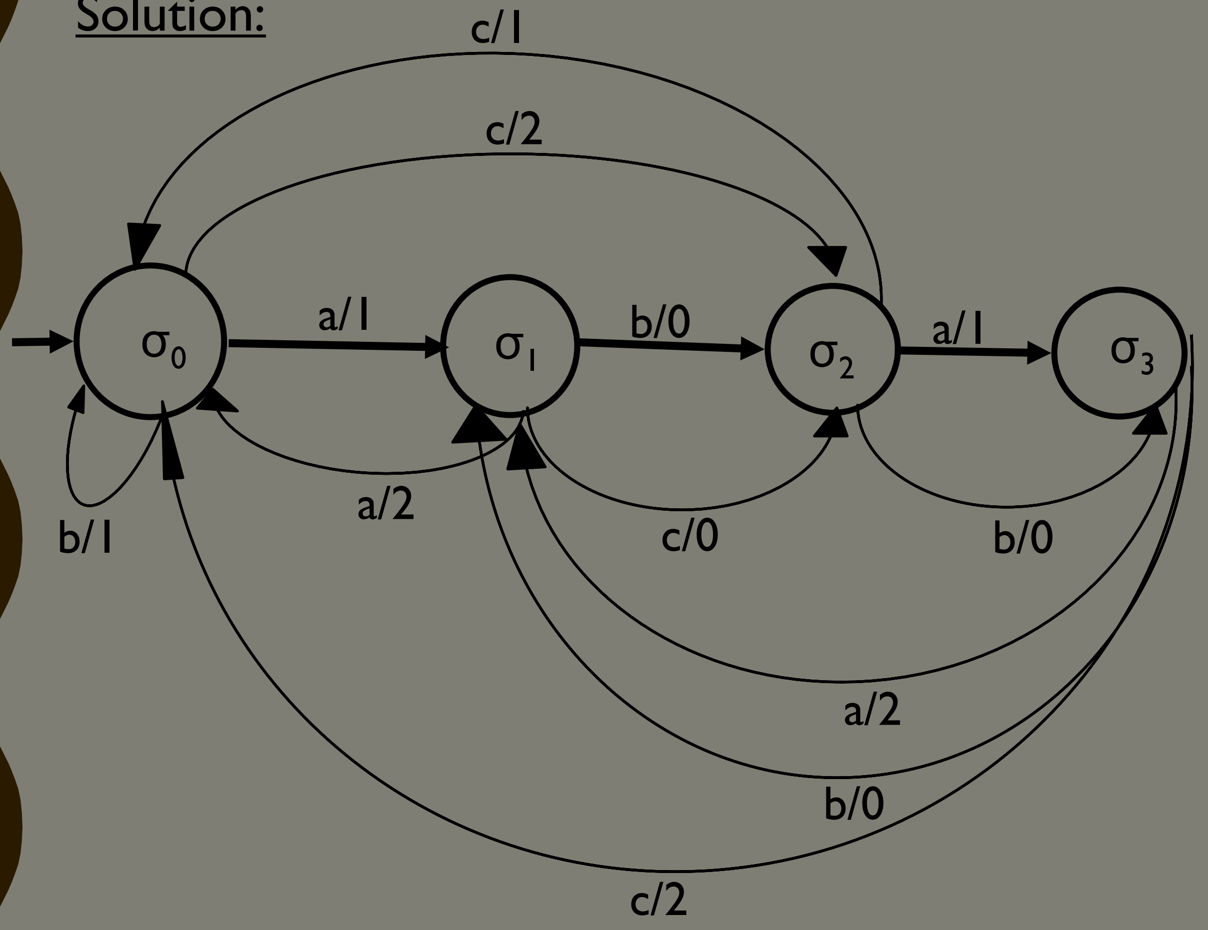
Output: 0101100

FINITE STATE MACHINE:

Draw the transition diagram of the finite state machine M, where, $I = \{a, b, c\}$ $O = \{0, 1, 2\}$ $S = \{\sigma_0, \sigma_1, \sigma_2, \sigma_3\}$, $\sigma = \sigma_0$ and transition given by below table. Find the output string corresponding to the input string "aabaab"

S \ $\mathcal{I}$	f: a	f: b	f: c	g: a	g: b	g: c
σ_0	σ_1	σ_0	σ_2	1	1	2
σ_1	σ_0	σ_2	σ_2	2	0	0
σ_2	σ_3	σ_3	σ_0	1	0	1
σ_3	σ_1	σ_1	σ_0	2	0	2

Solution:



1. $\mathcal{I} = \{a, b\}, \mathcal{O} = \{0, 1\}, \mathcal{S} = \{\sigma_0, \sigma_1\}$

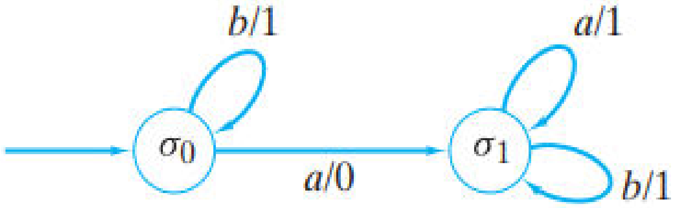
	f		g	
$\mathcal{S} \backslash \mathcal{I}$	a	b	a	b
σ_0	σ_1	σ_1	1	1
σ_1	σ_0	σ_1	0	1

2. $\mathcal{I} = \{a, b\}, \mathcal{O} = \{0, 1\}, \mathcal{S} = \{\sigma_0, \sigma_1\}$

	f		g	
$\mathcal{S} \backslash \mathcal{I}$	a	b	a	b
σ_0	σ_1	σ_0	0	0
σ_1	σ_0	σ_0	1	1

In Exercises 6–10, find the sets $\mathcal{I}$, $\mathcal{O}$, and $\mathcal{S}$, the initial state, and the table defining the next-state and output functions for each finite-state machine.

6.



MATHEMATICAL FOUNDATION FOR COMPUTER SCIENCE

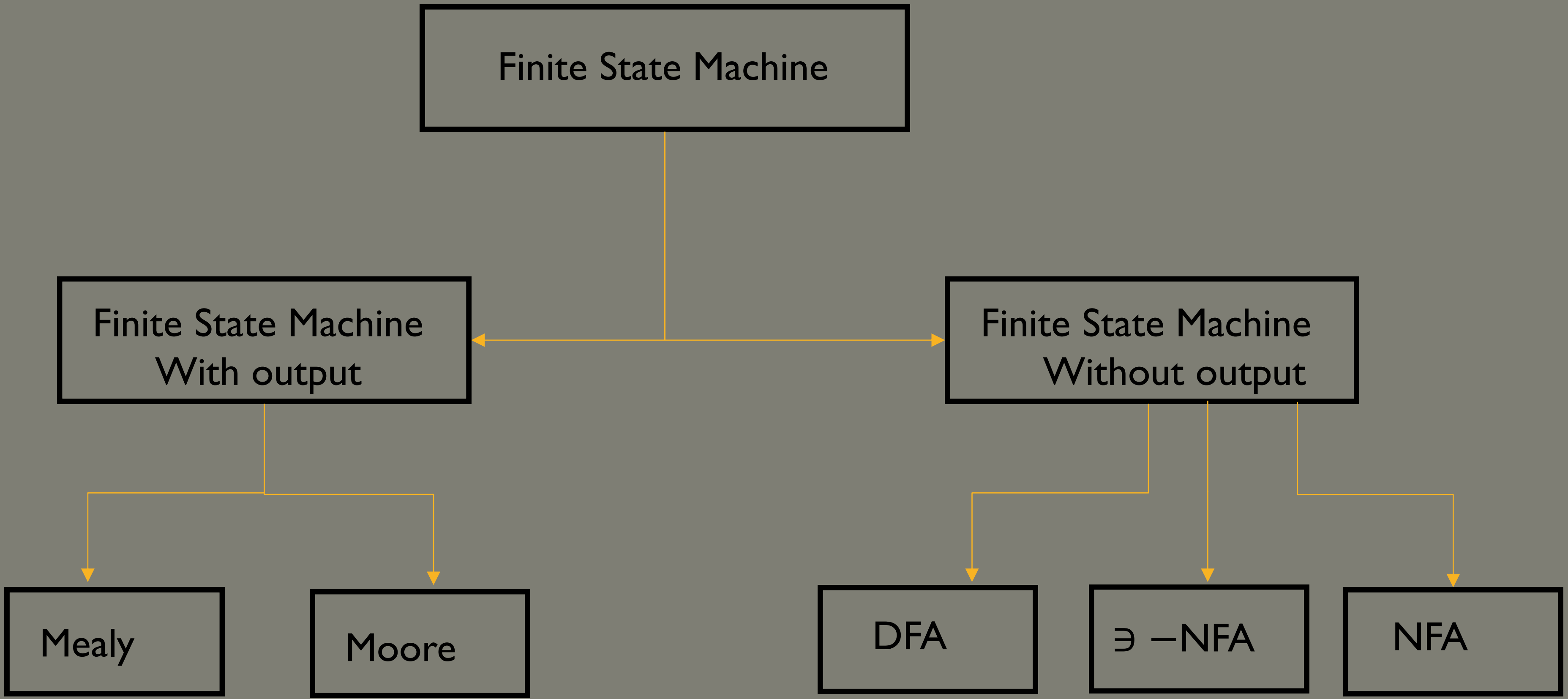
Prepared by: Er. Ankit Kharel

Nepal college of information technology

Finite State Automata

- *Sequential Circuits and Finite state Machine*
- *Finite State Automata*
- *Non-deterministic Finite State Automata*
- *Language and Grammars*
- *Language and Automata*
- *Regular Expression*

FINITE STATE MACHINE(FSM):



FEW TERMINOLOGIES:

1. **Alphabet(Σ)**: It is a collection of input symbol. Example: Binary alphabet Σ= {1,0} , Σ= {a, b}

2. **String(w) :** It is a sequence of input symbol. Length of string is denoted by |w|.
   Example: w = aababa, w=01010101
   An empty string(Length 0) is denoted by ε or λ.

3. **Language(L):** It is a collection of string over given alphabet
   Example: Σ= {a, b} , L_1 = {set of all string of length 2 }, L_2= {set of all strong that begin with a}
   L_1 = {ab, ba , aa, bb} L_2 = {a , aa, ab, aaab, abab,……………..}

3. **Kleen closure(Σ^*):** The Kleene star, $\sum$*, is a unary operator on a set of symbols or strings, $\sum$, that gives the infinite set of all possible strings of all possible lengths over $\sum$ including **λ**.
   Example − If $\sum$ = {a, b}, $\sum^*$ = {λ, a, b, aa, ab, ba, bb,………..}

4. **Positive closure(Σ^+):** The set $\sum^+$ is the infinite set of all possible strings of all possible lengths over $\sum$ excluding λ.
   Example − If $\sum$ = {a, b}, $\sum^+$ = {a, b, aa, ab, ba, bb,……….}

DETERMINISTIC FINITE STATE AUTOMATA(DFA):

- DFA is a Finite State Machine that accepts or rejects a given string of symbols, by running through a state sequence uniquely determined by the string. *Deterministic* refers to the uniqueness of the computation run.

- Formal Definition of Deterministic Finite State Automata(DFA)

  A deterministic finite state automata is defined as

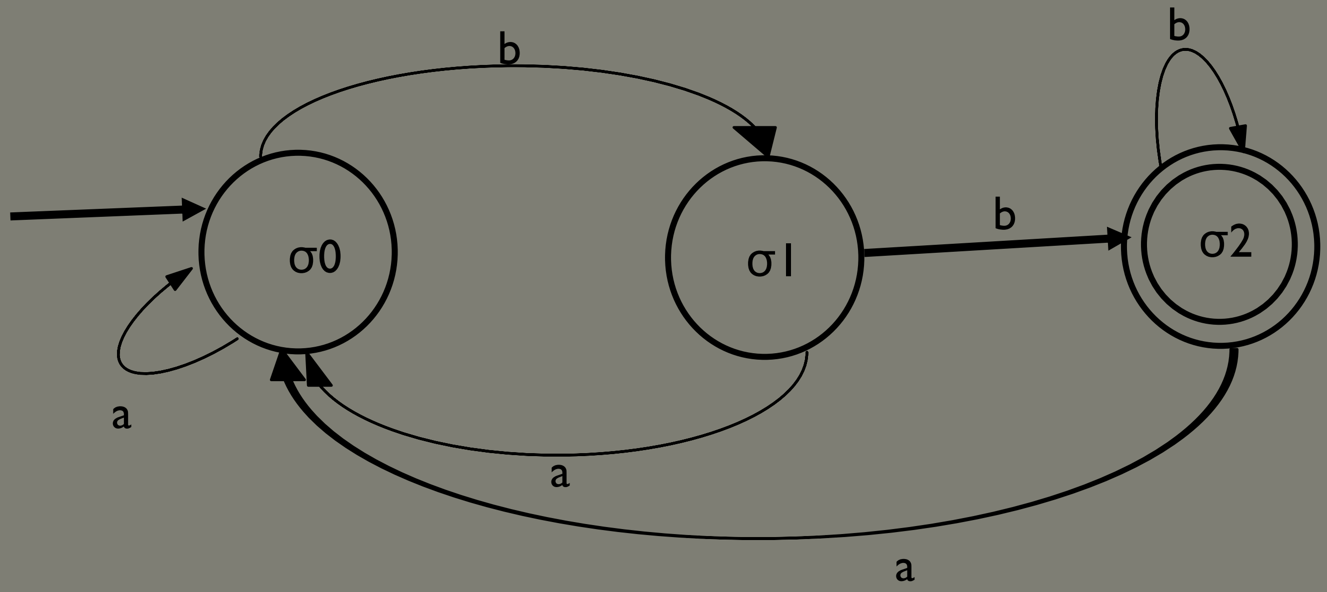
  $$\mathbf{M = (I,\ S,\ f,\ \sigma,\ A)}$$

  (a) **I** = A finite set of input symbols.
  (c) **S** = A finite set of states.
  (d) **f** =A next-state Transition function defines as $S \times I$ into S.
  (e) **σ** =An initial state $\boldsymbol{\sigma} \in S$.
  (f) **A** = set of accepting state or final state

DETERMINISTIC FINITE STATE AUTOMATA(DFA):

The transition diagram of the finite-state automaton A = (I, S, f, A, σ), where I = {a, b}, S = {σ0, σ1, σ2}, A = {σ2}, σ = σ0, and f is given by the following table

$\mathcal{S}$ \ $\mathcal{I}$	f: a	f: b
σ_0	σ_0	σ_1
σ_1	σ_0	σ_2
σ_2	σ_0	σ_2



Is the string "abaa" accepted by the finite-state automaton?

ACCEPTANCE BY DFA:

Let $A = (I, S, f, A, \sigma)$ be a finite-state automaton.

Let $w = x_1 \cdots x_n$ be a string over I. If there exist states $\sigma_0, \ldots, \sigma_n$ satisfying

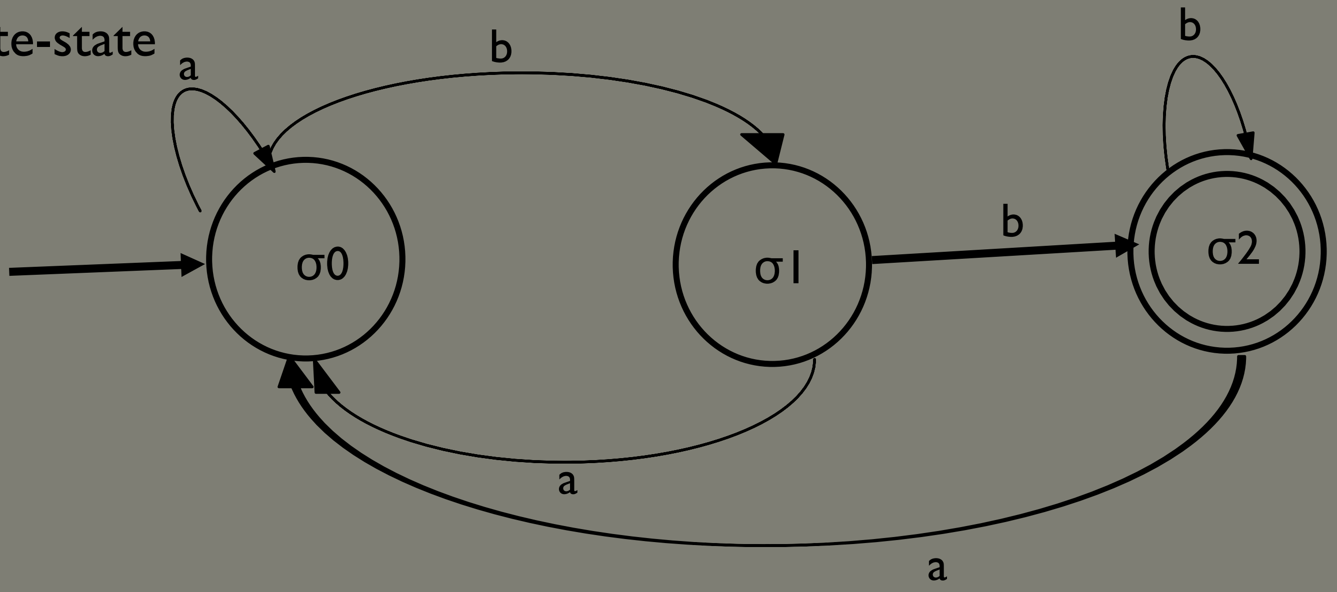
(a) $\sigma_0 = \sigma$

(b) $f(\sigma_i - 1, x_i) = \sigma_i$ for $i = 1, \ldots, n$

(c) $\sigma_n \in A$,

we say that w is accepted by A. The null string is accepted if and only if $\sigma \in A$.

Is the string "abaa" accepted by the finite-state automaton?

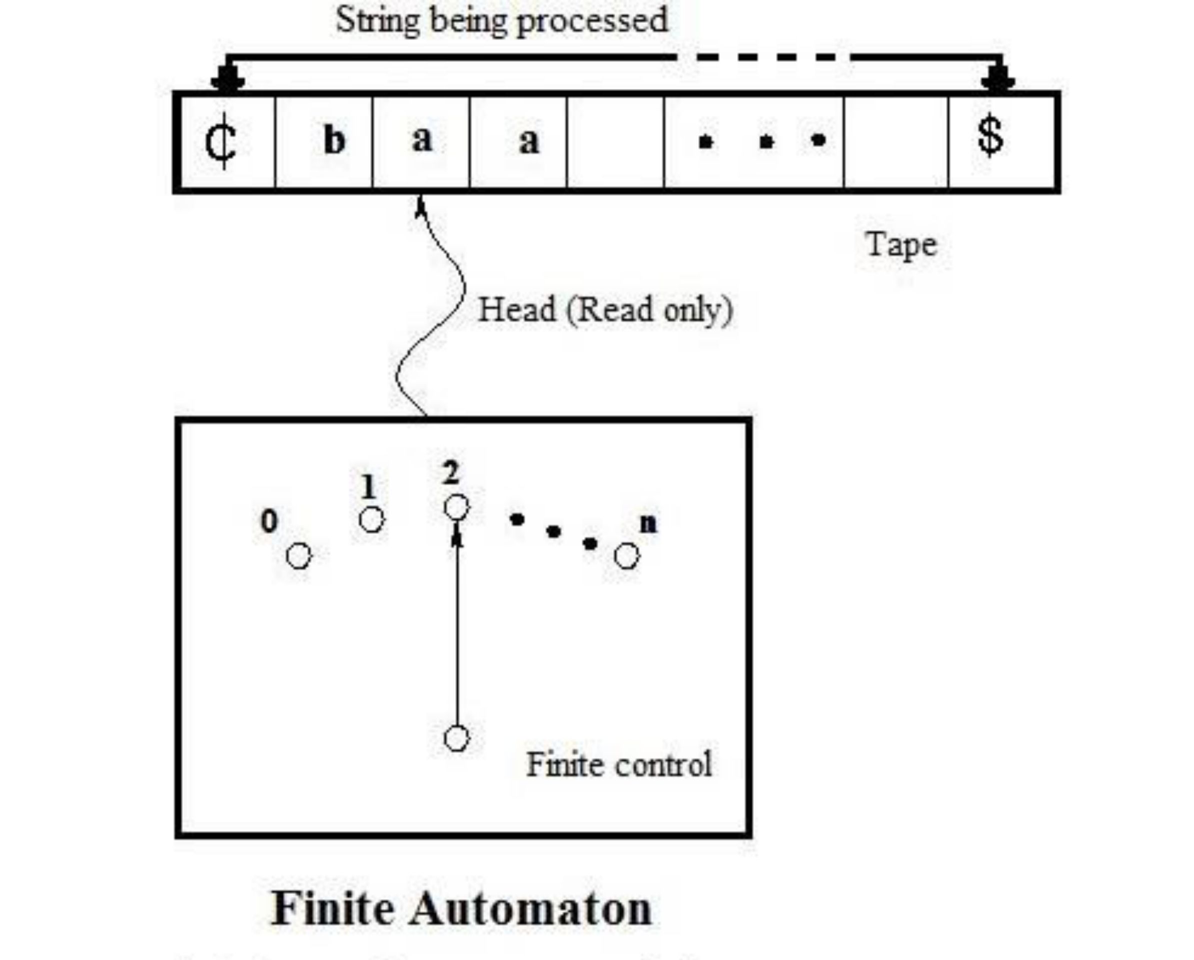


S\I	input	Next state
σ0	a	σ0
σ0	b	σ1
σ1	a	σ0
σ0	a	σ0

Since, σ0 is not the final state the string is not accepted

PROCESSING OF STRING BY DFA:

The thought process of a finite automaton can be graphically represented as;



Block diagram of Finite Automaton (FA)

PROCESSING OF STRING BY DFA:

The various components consist by a finite automata is as follows;

a) **Input tape:** The input tape has the left end and extends to the right end. It is divided into squares and each square containing a single symbol from the input alphabet ∑. The end squares of the tape contain the end markers ₵ at the left end and the end marker $ at the right end. The absence of endmarkers indicates that the tape is of infinite length.

b) **Read only Head:** The tape has a read only head that examines only one square at a time and can move one square either to the left or to the right. At the beginning of the operation the head is always at the leftmost square of the input tape. For further analysis, machine restrict the moment of the read only head from left to right direction only and one square every time when it reads a symbols.

c) **Finite control:** There is a finite control which determines the state of the automaton and also controls the movement of the head. The input to the finite control will usually be the symbol under the read head (assumed a) and the present state of the machine (assumed q) to give the following outputs;
A motion of Read head along the tape to the next square.
The next state of the finite state machine given by $\delta(q, a)$

1.Construc a DFA which accepts a language of all strings containing 'a' over Σ={a, b}.

<u>Solution:</u>

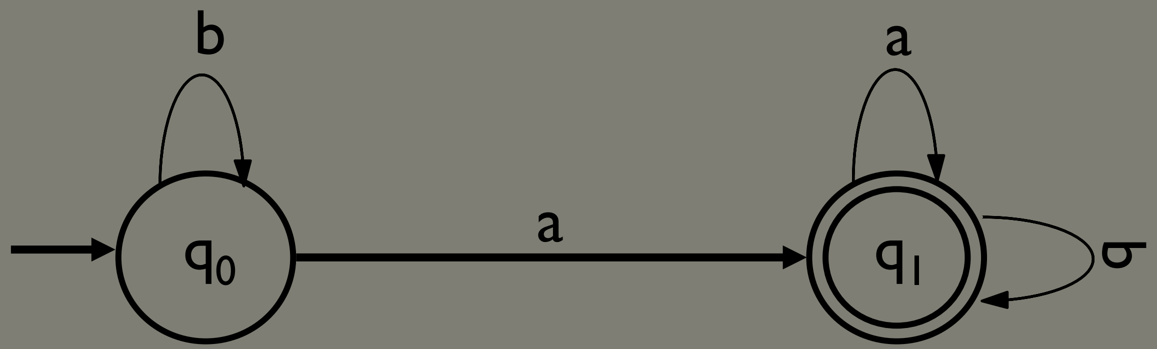


fig: <u>Transition diagram</u>

The required FSA is,

M = {I, S, f, σ, A} where,

I = {a , b} is the set of input symbols

S = {q_0 , q_1 } is the set of finite states

σ = q_0 is an initial state

A = {q_1} is the final accepting state

f:S*I→S is the next state transition function defined by following table

S \ I	a	b
q_0	q_1	q_0
q_1	q_1	q_1

2.Construc a DFA which accepts a language of all strings starting with ‘a’ over Σ={a, b}.

Solution:

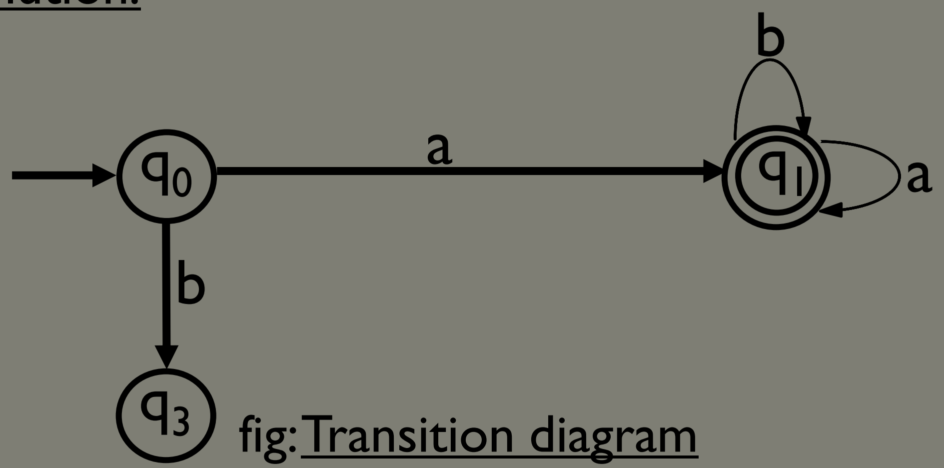


fig:Transition diagram

The required FSA is,

M = {I, S, f, σ, A} where,

I = {a , b} is the set of input symbols

S = {q_0 , q_1 } is the set of finite states

σ = q_0 is an initial state

A = {q_1} is the final accepting state

f:S*I→S is the next state transition function defined by following table

S \ I	a	b
q_0	q_1	q_3
q_1	q_1	q_1

3. Construc a DFA which accepts a language of all strings containing odd number of 'a' over Σ={a, b}.

Solution:

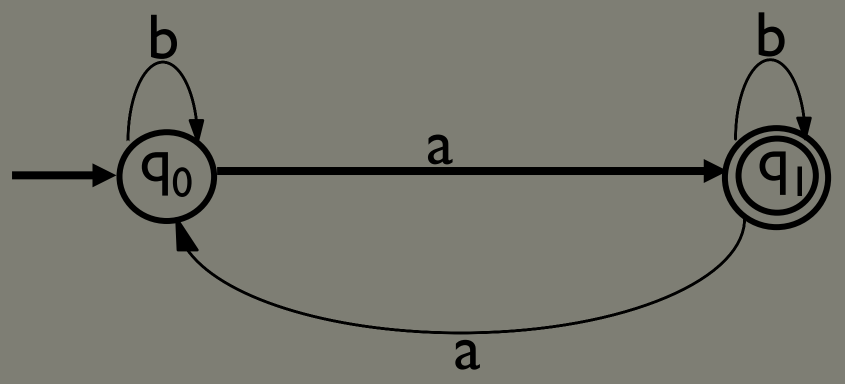


fig: Transition diagram

The required FSA is,

M = {I, S, f, σ, A} where,

I = {a , b} is the set of input symbols

S = {q_0 , q_1 } is the set of finite states

σ = q_0 is an initial state

A = {q_1} is the final accepting state

f:S*I→S is the next state transition function defined by following table

S \ I	a	b
q_0	q_1	q_0
q_1	q_0	q_1

4. Construc a DFA which accepts a language of all strings containing substring ‘abaab’ over Σ={a, b}.

Solution:

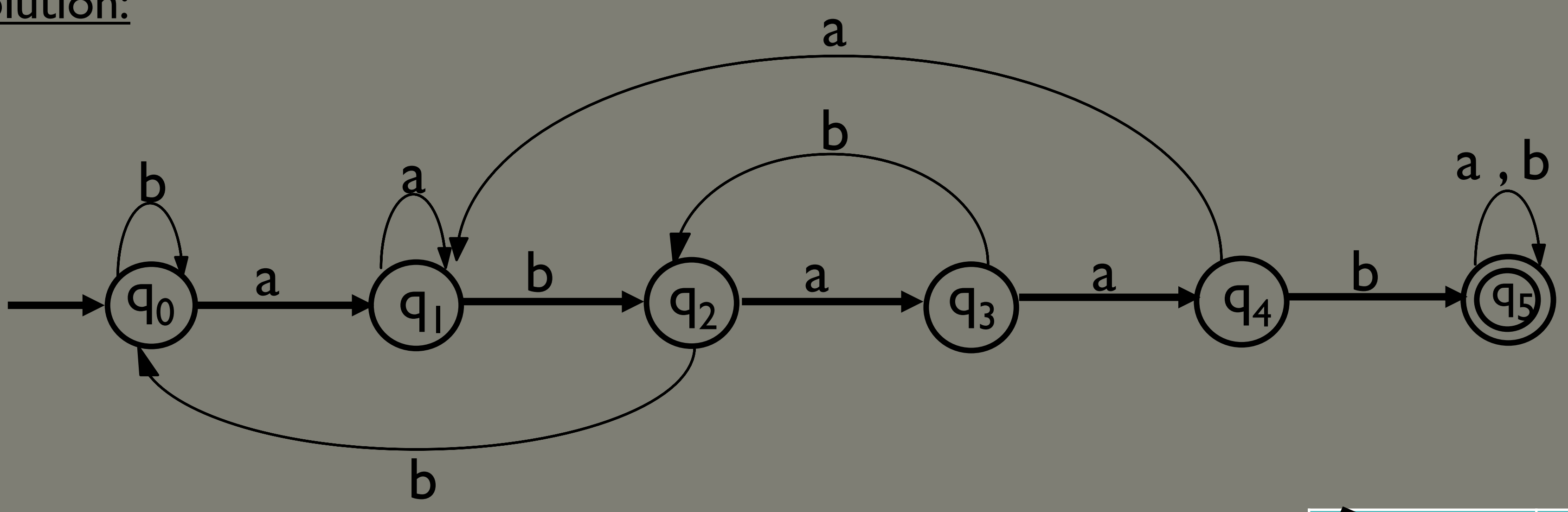


fig: Transition diagram

The required FSA is,

M = {I, S, f, σ, A} where,

I = {a , b} is the set of input symbols

S = {q_0 , q_1 , q_2 , q_3 , q_4 , q_5 } is the set of finite states

σ = q_0 is an initial state

A = {q_5} is the final accepting state

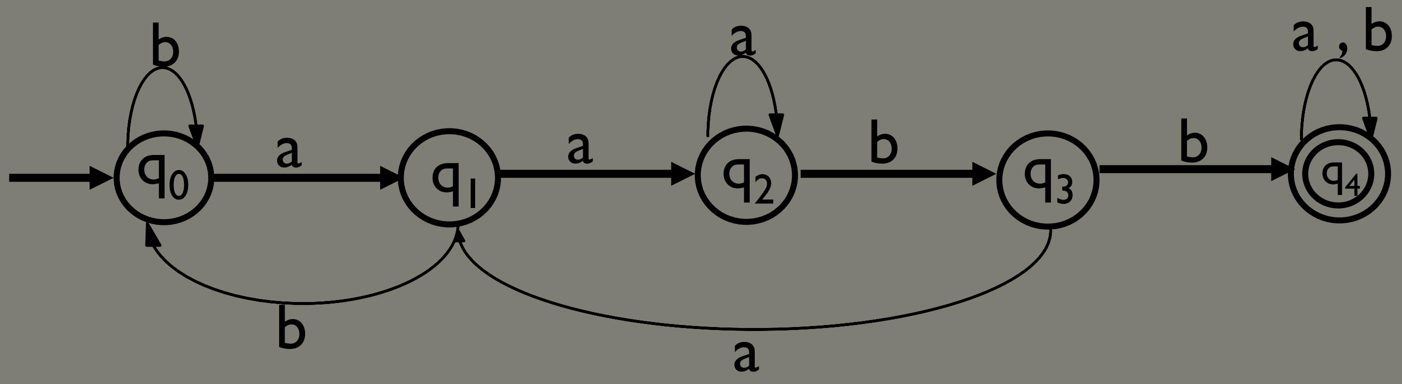
f:S*I→S is the next state transition function defined by following table

S \ I	a	b
q_0	q_1	q_0
q_1	q_1	q_2
q_2	q_3	q_0
q_3	q_4	q_2
q_4	q_1	q_5
q_5	q_5	q_5

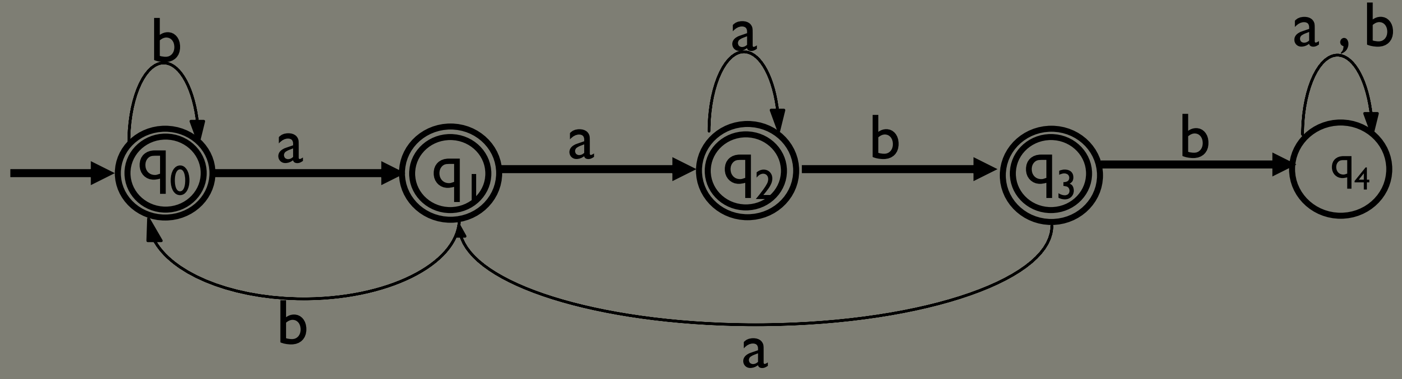
5.Construc a DFA which accepts a language of all strings that does not contain substring ‘aabb’ over Σ={a, b}.

Solution:

1. First construct the DFA that accepts string containing substring “aabb”:



2. Now, Flip the final and non final state:



MATHEMATICAL FOUNDATION FOR COMPUTER SCIENCE

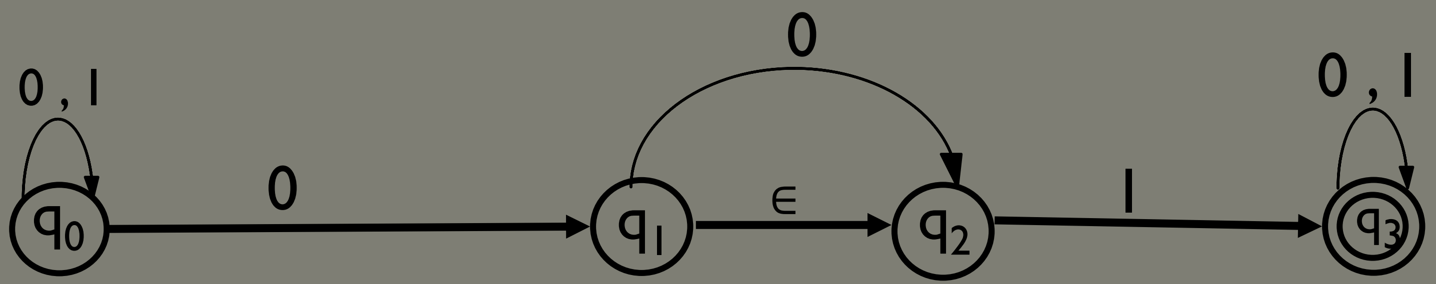
Prepared by: Er. Ankit Kharel

Nepal college of information technology

FINITE STATE AUTOMATA

- *Sequential Circuits and Finite state Machine*
- *Finite State Automata*
- *Non-deterministic Finite State Automata*
- *Language and Grammars*
- *Language and Automata*
- *Regular Expression*

NON-DETERMINISTIC FINITE STATE AUTOMATA(NFA):



Consider the machine shown in figure above. Like DFA it has finitely many states and transitions labelled by symbols from an input alphabet. However Above Figure has important difference when compared with DFA model:

- State q_0 has two outgoing transition labelled with 0.
- States q_1 and q_2 have missing transition. q_1 has no transition labelled 1, while q_2 has no transition labelled 0.
- State q_1 has transition that is labelled not by an input symbol but by $\in$.

NON-DETERMINISTIC FINITE STATE AUTOMATA(NFA):

Key Difference Between NFA and DFA:

a) An NFA can have multiple transitions for a symbol from the same state but DFA can only have one transition for each symbol.

b) An NFA is not required to have a transition for each symbol where as for DFA there should be transition for each symbol.

c) NFA can have a transition for an empty string where as DFA cannot transition on empty string.

FORMAL DEFINITION OF NFA:

A NFA , N is a quin – Tuple(5 Tuple) defined as,

N = {I, S, f, σ, A} where,

I is the set of input symbols

S is the set of finite states

σ is an initial state

A is the final accepting state

$f{:}S*I \rightarrow 2^{s}$ is the next state transition function

A string 'w' is said to be accepted by NFA if there exist at least one transition path on which we start and ends at final state.

- *Every DFA is NFA and every NFA can be converted to DFA.*
- *Power of Both DFA and NFA is same.*

1. Construc a NFA which accepts a language of all strings stating with 'ab' over Σ={a, b}.

Solution:

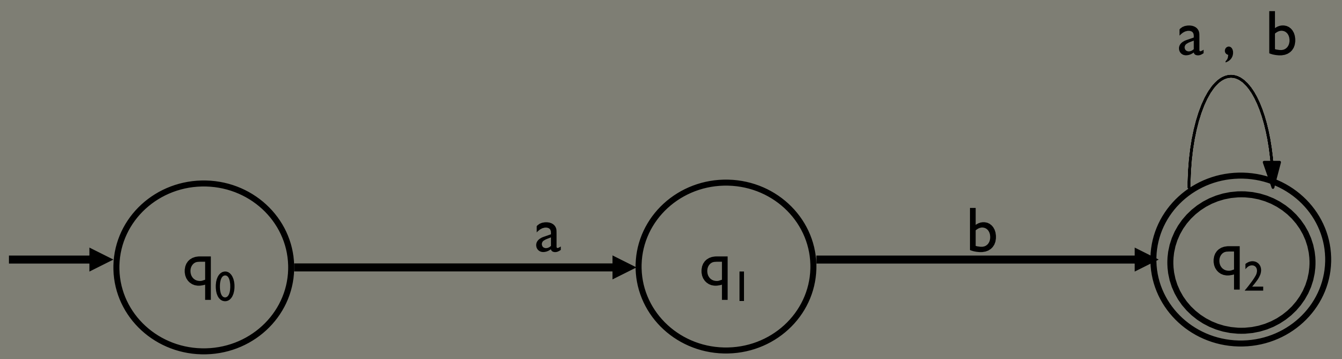


fig: Transition diagram

The required NFA is,

N = {I, S, f, σ, A} where,

I = {a , b} is the set of input symbols

S = {q_0 , q_1 , q_2 } is the set of finite states

σ = q_0 is an initial state

A = {q_2} is the final accepting state

f:S*I→2^s is the next state transition function defined by following table

S \ I	a	b
q_0	q_1	Ø
q_1	Ø	q_2
q_2	q_2	q_2

2.Construc a NFA which accepts a language of all strings ending with 'bab' over Σ={a, b}.

Solution:

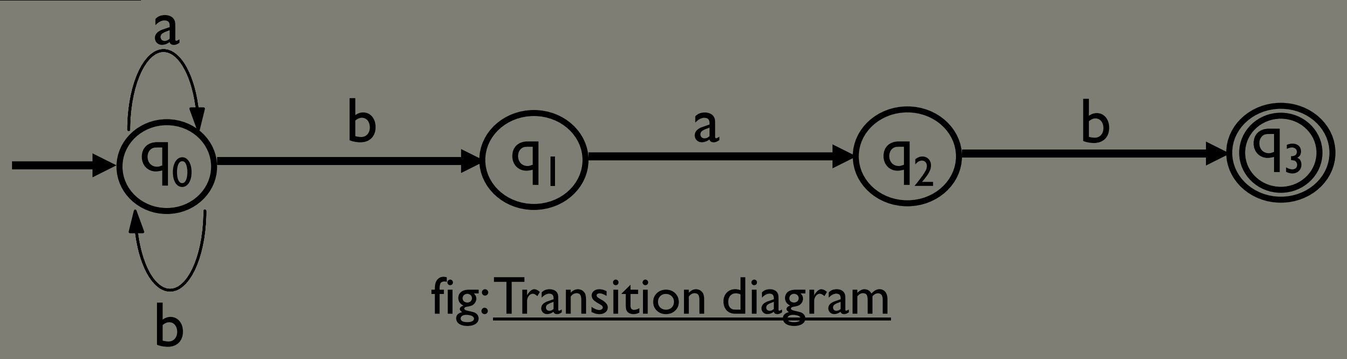


fig:Transition diagram

The required FSA is,

N = {I, S, f, σ, A} where,

I = {a , b} is the set of input symbols

S = {q_0 , q_1 , q_2 , q_3 } is the set of finite states

σ = q_0 is an initial state

A = {q_3} is the final accepting state

f:S*I→2^s is the next state transition function defined by following table

S \ I	a	b
q_0	q_0	q_0 , q_1
q_1	q_2	Ø
q_2	Ø	q_3
q_3	Ø	Ø

3.Construc a NFA which accepts a language of all strings starting and ending with ‘a’ over Σ={a, b}.

<u>Solution:</u>

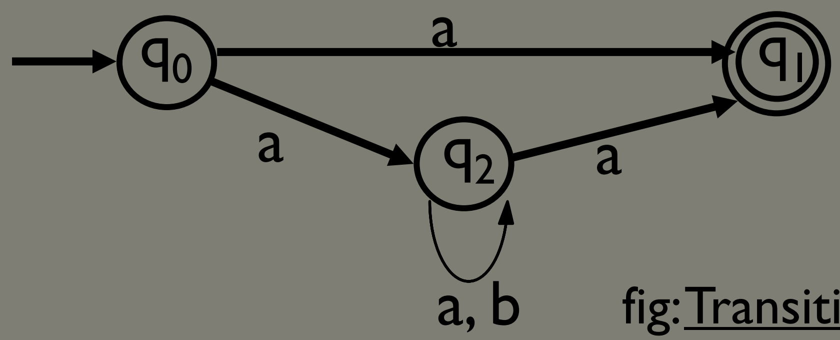


fig: <u>Transition diagram</u>

The required FSA is,

N = {I, S, f, σ, A} where,

I = {a , b} is the set of input symbols

S = {q_0 , q_1 , q_2 } is the set of finite states

σ = q_0 is an initial state

A = {q_1} is the final accepting state

f:S*I→2^s is the next state transition function defined by following table

S \ I	a	b
q_0	q_1, q_2	Ø
q_1	Ø	Ø
q_2	q_2 , q_1	q_2

4.Construc a NFA which accepts a language of all strings containing substring 'abaab' over Σ={a, b}.

<u>Solution:</u>

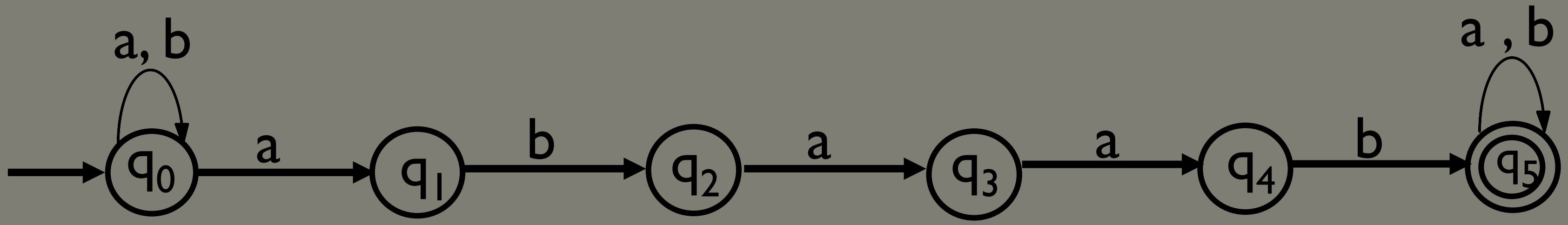


fig: <u>Transition diagram</u>

The required FSA is,

N = {I, S, f, σ, A} where,

I = {a , b} is the set of input symbols

S = {q_0 , q_1 , q_2 , q_3 , q_4 , q_5 } is the set of finite states

σ = q_0 is an initial state

A = {q_5} is the final accepting state

f:S*I→2^s is the next state transition function defined by following table

S \ I	a	b
q_0	q_0 , q_1	q_0
q_1	∅	q_2
q_2	q_3	∅
q_3	q_4	∅
q_4	∅	q_5
q_5	q_5	q_5

CONVERSION OF NFA TO DFA:

We use subset construction method:

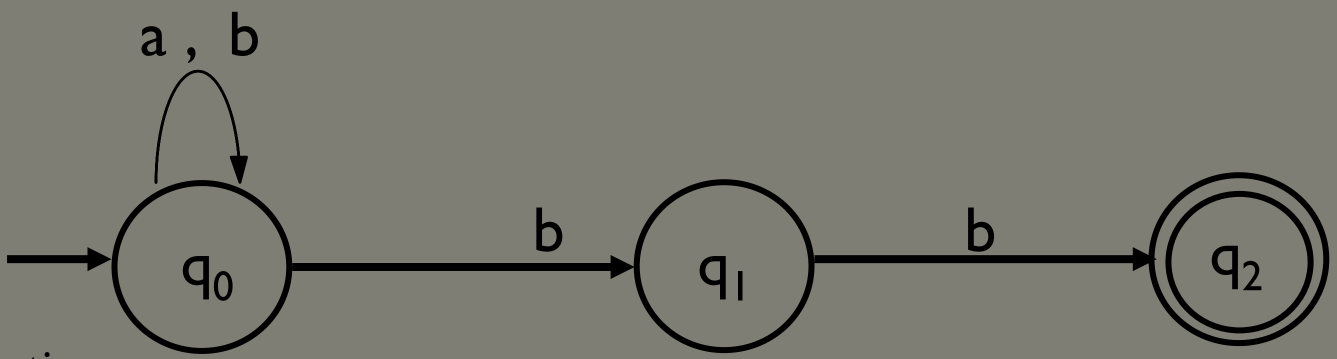
1) Construct a transition table of given NFA.

2) Identify all the new states from the transition table and find the transition for each new state in term of input symbols.

3) This process is continued until transaction for all the new states are identified.

4) Finally, draw a transition diagram by using all the states obtained.

1. Convert the following NFA to DFA:



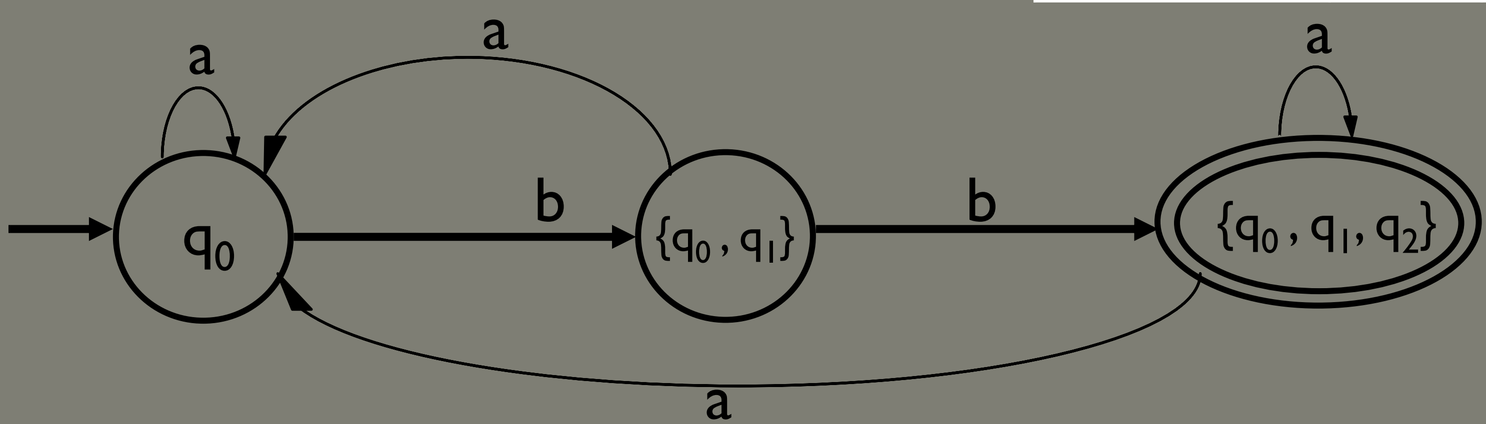
Solution:

a. Transition table for NFA:

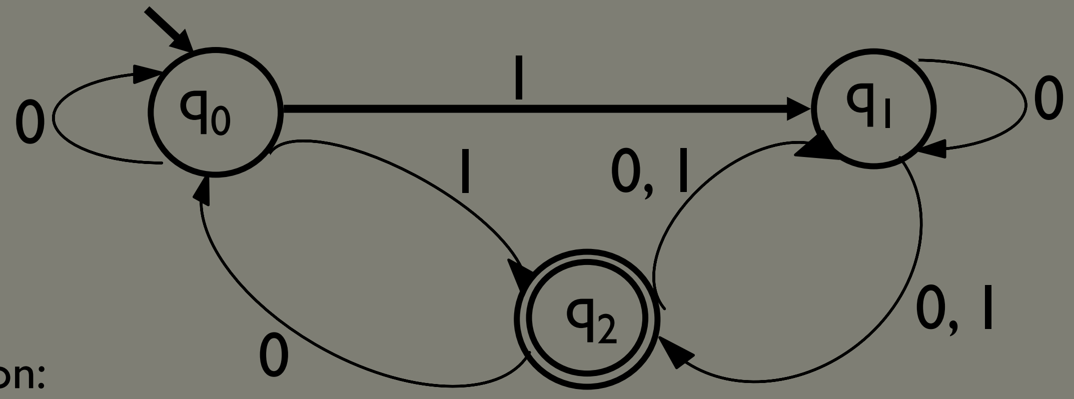
S\I	a	b
q_0	q_0	q_0 , q_1
q_1	Ø	q_2
q_2	Ø	Ø

b. Identifying new states and transition for DFA:

S\I	a	b
q_0	q_0	$\{q_0 , q_1\}$
$\{q_0 , q_1\}$	q_0	$\{q_0 , q_1, q_2\}$
$\{q_0 , q_1, q_2\}$ (Final)	q_0	$\{q_0 , q_1, q_2\}$



2. Convert the following NFA to DFA:



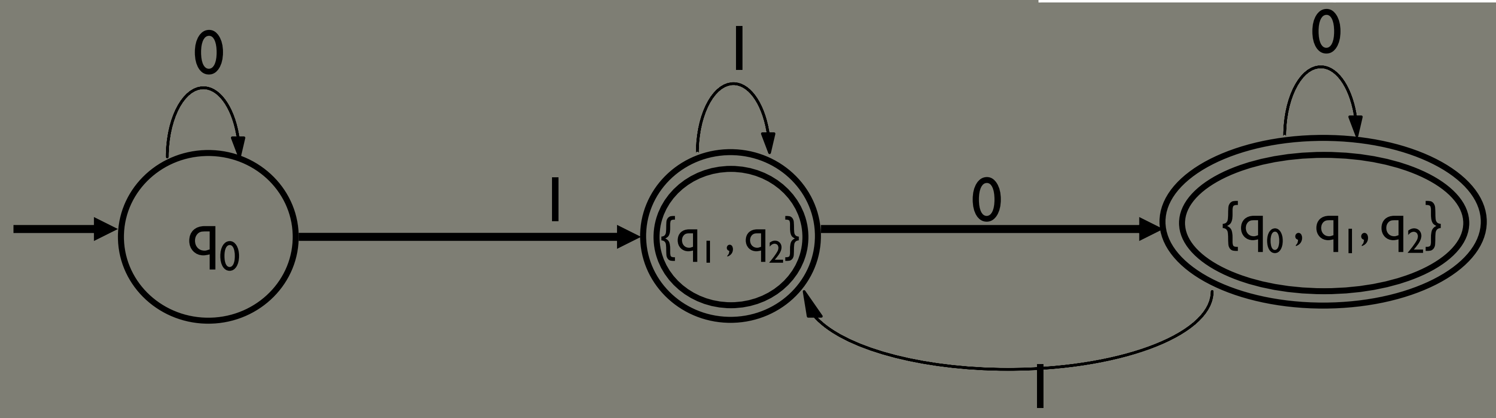
Solution:

a. Transition table for NFA:

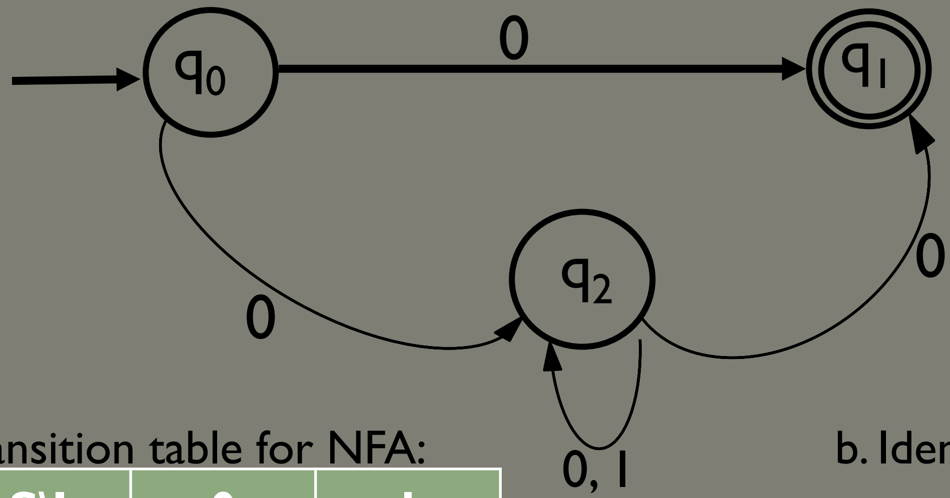
S\I	0	1
q_0	q_0	q_1 , q_2
q_1	q_1 , q_2	q_2
q_2	q_0 , q_1	q_1

b. Identifying new states and transition for DFA:

S\I	0	1
q_0	q_0	$\{q_1 , q_2\}$
$\{q_1 , q_2\}$	$\{q_0 , q_1, q_2\}$	$\{q_1, q_2\}$
$\{q_0 , q_1, q_2\}$	$\{q_0 , q_1, q_2\}$	$\{q_1, q_2\}$



3. Convert the following NFA to DFA:

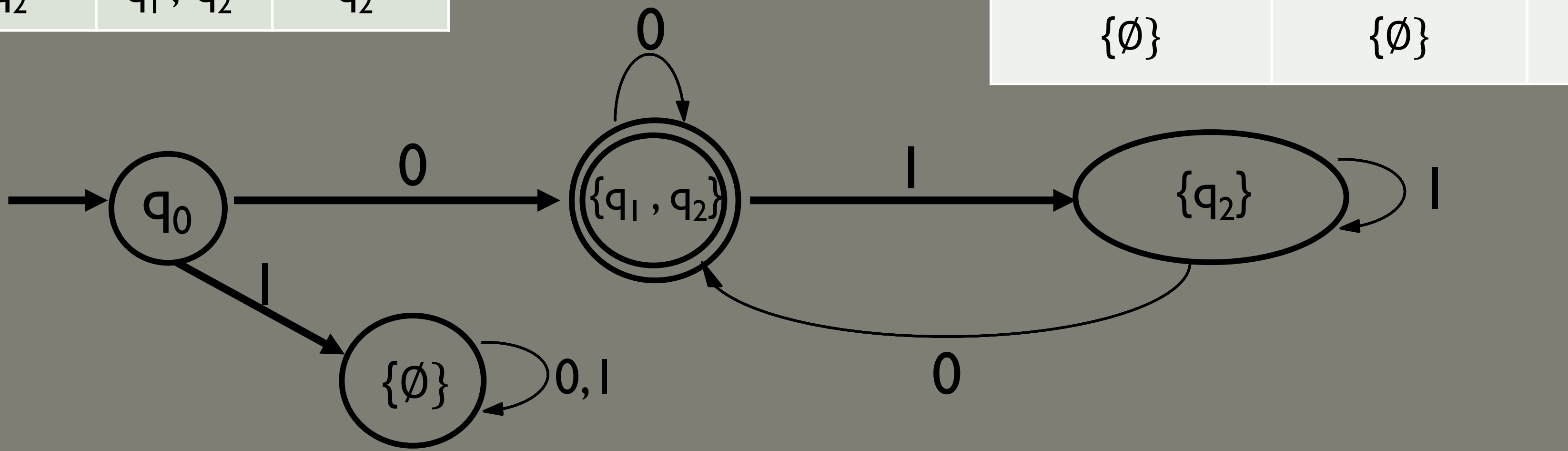


a. Transition table for NFA:

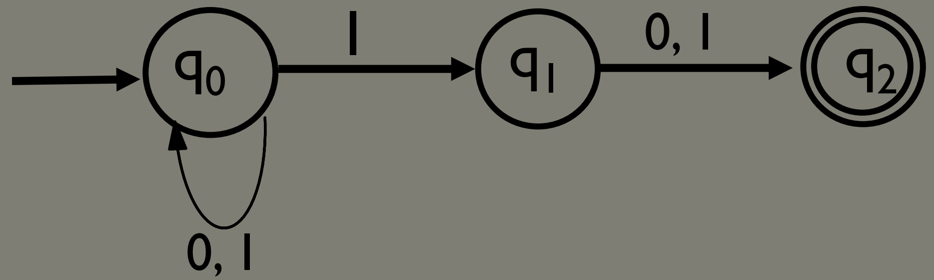
S\I	0	1
q_0	q_1 , q_2	∅
q_1	∅	∅
q_2	q_1 , q_2	q_2

b. Identifying new states and transition for DFA:

S\I	0	1
q_0	$\{q_1 , q_2\}$	{∅}
$\{q_1 , q_2\}$	$\{q_1, q_2\}$	$\{q_2\}$
$\{q_2\}$	$\{q_1, q_2\}$	$\{q_2\}$
{∅}	{∅}	{∅}



4. Convert the following NFA That accepts all string in which second last bit is 1.

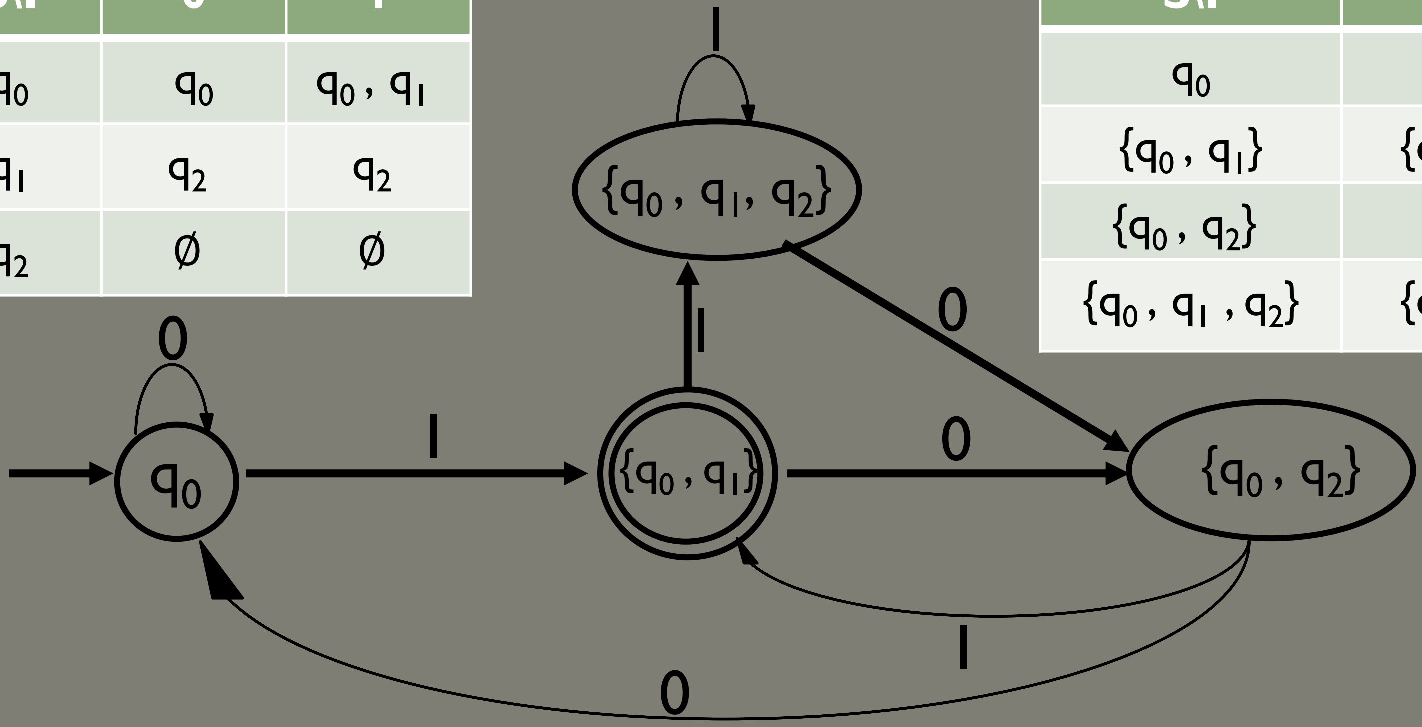


a. Transition table for NFA:

S\I	0	1
q_0	q_0	q_0 , q_1
q_1	q_2	q_2
q_2	Ø	Ø

b. Identifying new states and transition for DFA:

S\I	0	1
q_0	q_0	$\{q_0 , q_1\}$
$\{q_0 , q_1\}$	$\{q_0 , q_2\}$	$\{q_0 , q_1 , q_2\}$
$\{q_0 , q_2\}$	q_0	$\{q_0 , q_1\}$
$\{q_0 , q_1 , q_2\}$	$\{q_0 , q_2\}$	$\{q_0 , q_1 , q_2\}$



MATHEMATICAL FOUNDATION FOR COMPUTER SCIENCE

Prepared by: Er. Ankit Kharel

Nepal college of information technology

FINITE STATE AUTOMATA

- *Sequential Circuits and Finite state Machine*
- *Finite State Automata*
- *Non-deterministic Finite State Automata*
- *Language and Grammars*
- *Language and Automata*
- *Regular Expression*

REGULAR EXPRESSIONS:

RG
(Regular Grammar)
A Language is regular if there is a regular grammar to generate it.

RL
(Regular Language)

FA
(Finite Automata)
A Language is regular if there is a exist a finite automata to accept it.

RE
(Regular Expression)
A Language is regular if there is a exist a regular expression to express it.

REGULAR EXPRESSIONS:

The language accepted by finite automata can be easily described by simple **expressions** called **Regular Expressions**. A **regular expression** can also be described as a sequence of pattern that defines a string. **Regular expressions** are used to match character combinations in strings.

For instance:

In a regular expression, x^* means zero or more occurrence of x. L(R) = {e, x, xx, xxx, xxxx, ...}

In a regular expression, x^+ means one or more occurrence of x. L(R) = {x, xx, xxx, xxxx, ...}

Let 'R' be a regular expression over alphabet Σ:

a) ε is a Regular Expression denoting the set:

R= ε ; L(R) = {ε}

b) ϕ is a Regular Expression denoting the empty set:

R= ϕ ; L(R) = { }

c)For each symbol a∈ Σ , a is regular expression denoting set {a}.

R=a ; L(R) = {a}

a, b, and c are called primitive regular expression. It is the minimum language generated by RE.

REGULAR EXPRESSIONS:

(Operators used in Regular expression)

d. **Union(+)** of two RE is also Regular;

$R_1 = a$, $R_2 = b$, $R_1 \cup R_2 = a + b$ i.e. $R_1 \cup R_2 = a + b$ generates language that contains either a or b

e. **Concatenation(.)** of two RE is also Regular;

$R_1 = a$, $R_2 = b$, $R_1.R_2 = a.b$ i.e. $R_1.R_2 = a.b$ generates language that contains a and b.

f. **Kleene Closure** of RE is also regular;

$R_1 = a$, $R_1^* = a^*$

g. **Positive Closure** of RE is also regular;

$R_1 = a$, $R_1^+ = a^+$

REGULAR EXPRESSIONS:

Q. Find the regular expression for the following languages,

1. Language containing no string:

   R = ϕ

2. Language containing string of length 0:

   R = ε

3. Language accepting string of length 1 over $\Sigma = \{a, b\}$.

   L = (a, b)

   R = a+b

4. Language accepting string of length 2 over $\Sigma = \{a, b\}$..

   L= (aa , ab , ba , bb)

   R = aa + ab + ba + bb

   = a(a+b) +b(a+b)

   =(a+b)(a+b)

REGULAR EXPRESSIONS:

5. Language accepting any combination of a over $\Sigma = \{a\}$

   R = a^*

6. Language accepting any combination of a expect null over $\Sigma = \{a\}$

   R = a^+

7. Language accepting all the string containing any number of a's and b's over $\Sigma = \{a, b\}$.

   R = $(a + b)^*$

8. Language accepting string of length at most 2 over $\Sigma = \{a, b\}$..

   L= (ε , a, b, aa , ab , ba , bb)

   R = ε + a + b + aa + ab + ba + bb

REGULAR EXPRESSIONS:

9. Language accepting all string having a single b over $\Sigma = \{a, b\}$.
   R = a^*ba^*

10. Language accepting all string having at least one b over $\Sigma = \{a, b\}$.
    R = $(a+b)^*b(a+b)^*$

11. Language accepting all the string containing any number of a's and b's over $\Sigma = \{a, b\}$.
    R = $(a + b)^*$

12. Language containing string with 'bbbb' as substring over $\Sigma = \{a, b\}$.
    R = $(a+b)^*$ bbbb $(a+b)^*$

13. Language containing string that ends with 'ab' over $\Sigma = \{a, b\}$.
    R = $(a+b)^*$ ab

14. Language containing string that starts with 'ab' over $\Sigma = \{a, b\}$.
    R = ab $(a+b)^*$

REGULAR EXPRESSIONS:

15. Language accepting all string that starts with a and ends with a over $\Sigma = \{a, b\}$.

    R = a + a(a+b)*a

16. Language accepting all string that starts and ends with same symbol over $\Sigma = \{a, b\}$

    R = a(a+b)*a + b(a+b)*b + a + b

17. Language accepting all string that starts with a and ends with b over $\Sigma = \{a, b\}$.

    R = a(a+b)*b

18. Language accepting all string that starts and ends with different symbol over $\Sigma = \{a, b\}$

    R = a(a+b)*b + b(a+b)*a

19. Language accepting string that contains exactly two b's over $\Sigma = \{a, b\}$.

    R = a* b a* b a*

20. Language containing string that starts with 'ab' over $\Sigma = \{a, b\}$.

    R = ab (a+b)*

REGULAR EXPRESSIONS:

21. Language accepting all string where number of a is less than or equal to 2 over $\Sigma = \{a, b\}$.

R = $b^* + b^*a\,b^* + b^*a\,b^*a\,b^*$

22. Language accepting all string where 3rd symbol from LHS is b over $\Sigma = \{a, b\}$

R = $(a+b)(a+b)b(a+b)^*$
= $(a+b)^2b(a+b)^*$

23. Language accepting all string where every 0 is followed by immediate 11over $\Sigma = \{0, 1\}$.

R = $(1 + 011)^*$

24. Language accepting all string where 2nd symbol from RHS is b over $\Sigma = \{a, b\}$

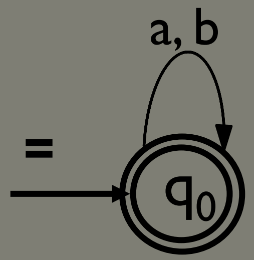
R = $(a+b)^*b(a+b)$

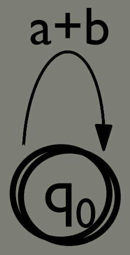
25. Second symbol is a and fourth symbols is b.

R = $(a+b)a(a+b)b\,(a+b)^*$

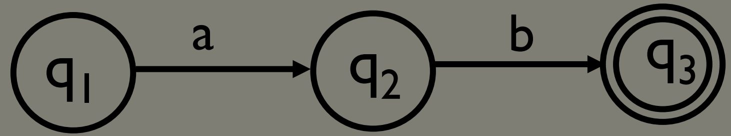
CONVERSION OF FINITE AUTOMATA TO RE:

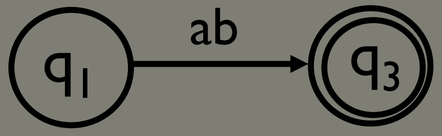
State Elimination Method:

1. = 

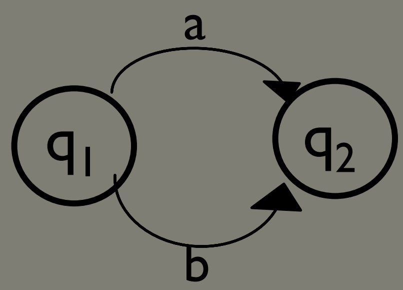
= 

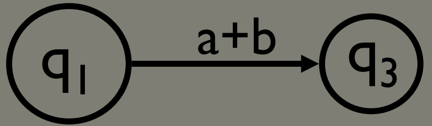
= **$(a + b)^*$**

2. = 

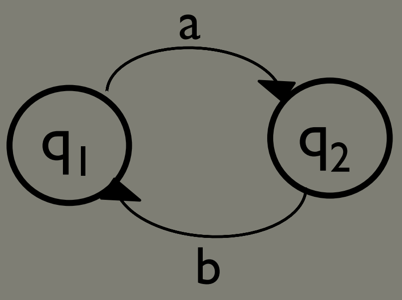
= 

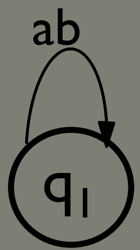
= **a.b**

3. = 

= 

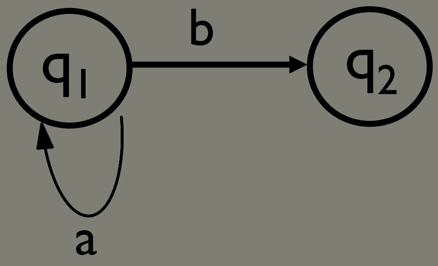
= **(a+b)**

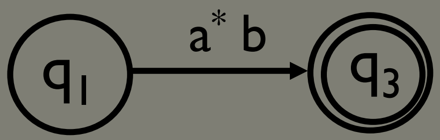
4. = 

= 

= **$(ab)^*$**

CONVERSION OF FINITE AUTOMATA TO RE:

4= 

= 

= $\mathbf{a^*.b}$

Steps to convert FA to RE:

1.If there exists any incoming edge to the initial state , create a new initial state having no incoming edge.

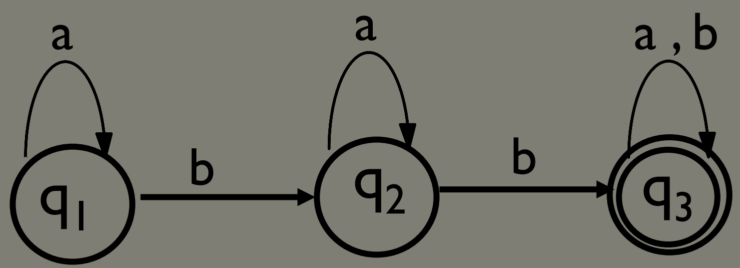
2. In case of multiple final states , convert them into non final state and create a new final state.

3.If there exist outgoing edge from final state create new final state having no outgoing edge.

4. Eliminate all intermediate state one by one . Only initial and Final state will be there. Their transition path will be our R.E.

CONVERSION OF FINITE AUTOMATA TO RE:

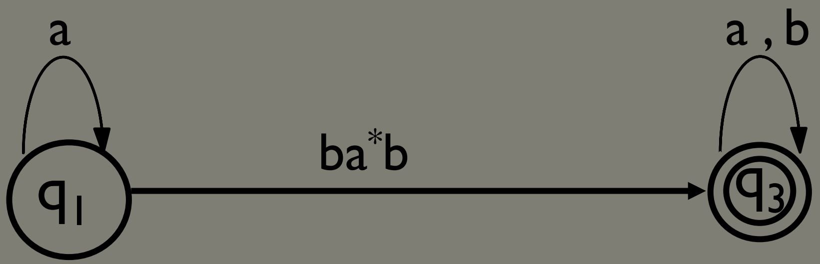
1. Convert Following to R.E



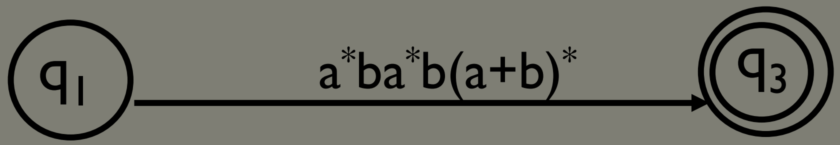
Solution:

i)Since there is no incoming edge in initial state, no outgoing edge from final state and there exist only one final state. So, start removing intermediate state .

=



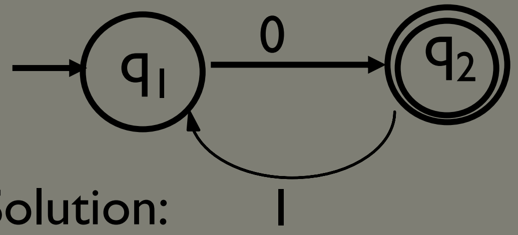
=



The Required R.E is:
$a^*ba^*b(a+b)^*$

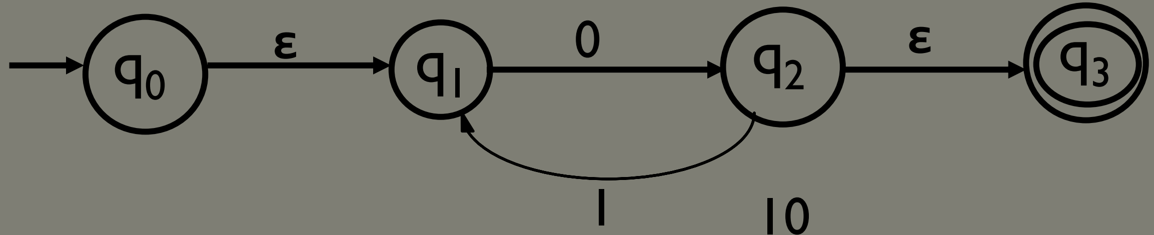
CONVERSION OF FINITE AUTOMATA TO RE:

2. Convert Following to R.E

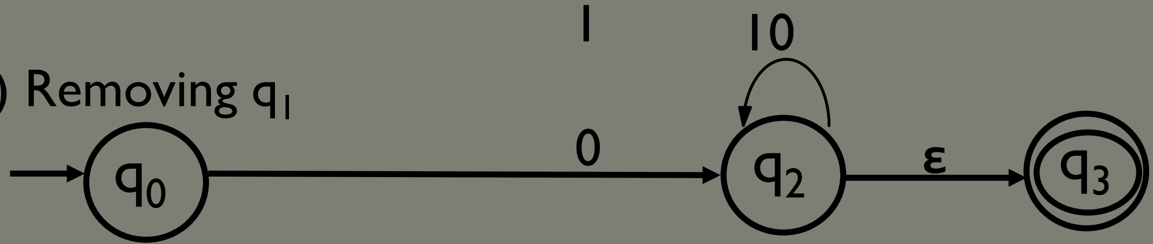


Solution:

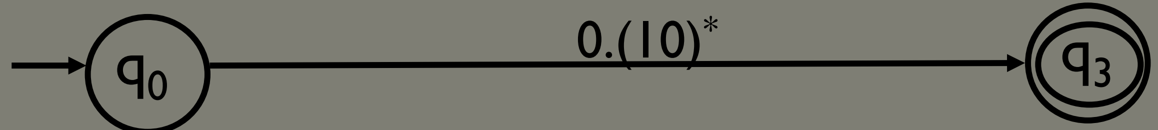
i)Since there is incoming edge in initial state, outgoing edge from final state



ii) Removing q_1



iii) Removing q_2



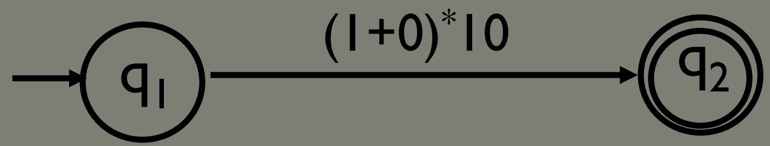
The Required R.E is:

$0.(10)^*$

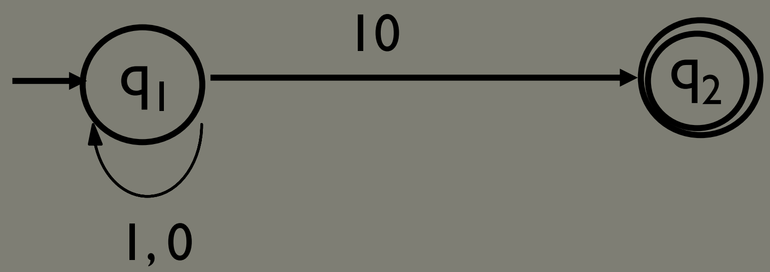
CONVERSION OF RE TO FINITE AUTOMATA :

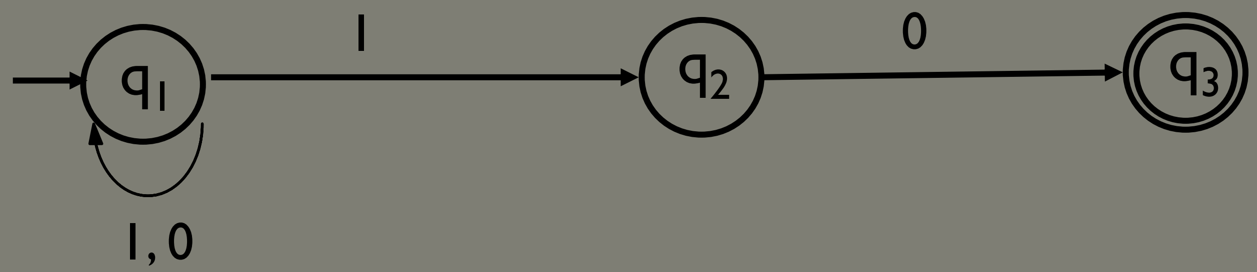
1. Convert Following to Finite Automata

Q.a $(1+0)^*10$



1





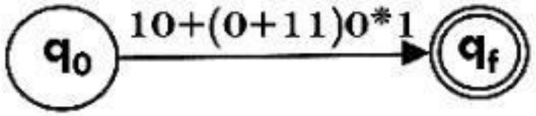
This is the required NFA

CONVERSION OF FINITE RE TO AUTOMATA :

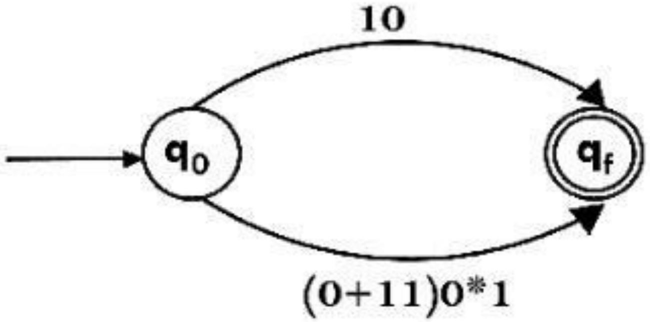
Design a FA from given regular expression 10 + (0 + 11)0* 1.

Solution: First we will construct the transition diagram for a given regular expression.

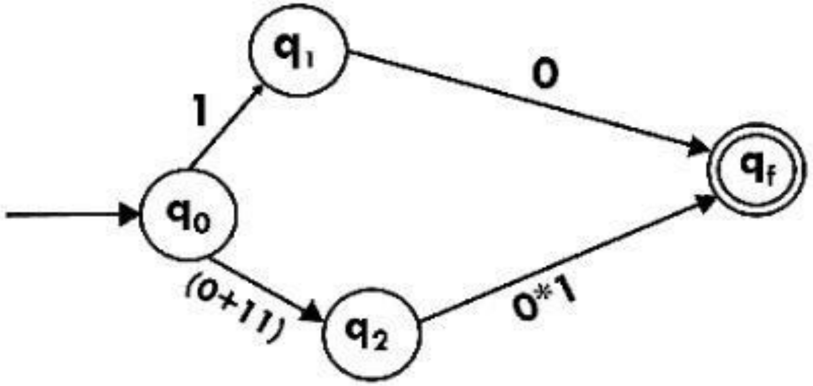
Step 1:



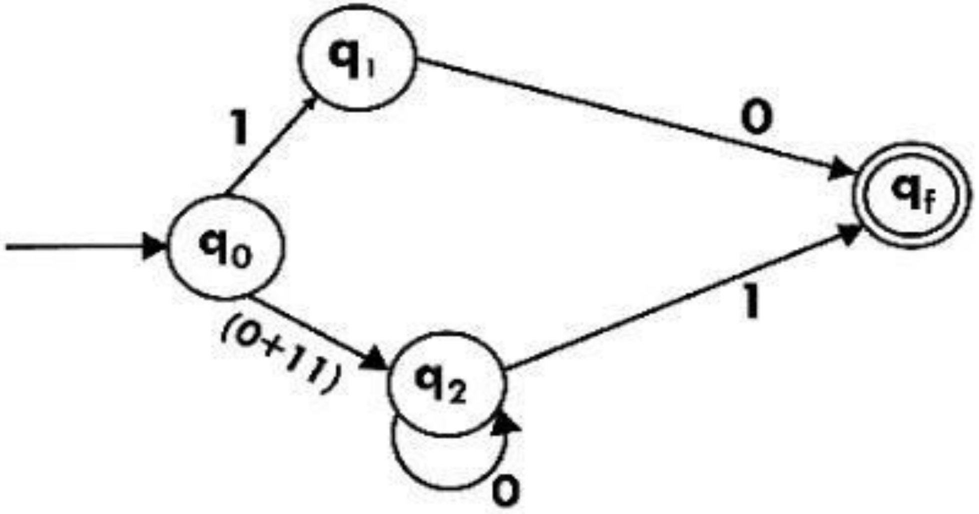
Step 2:



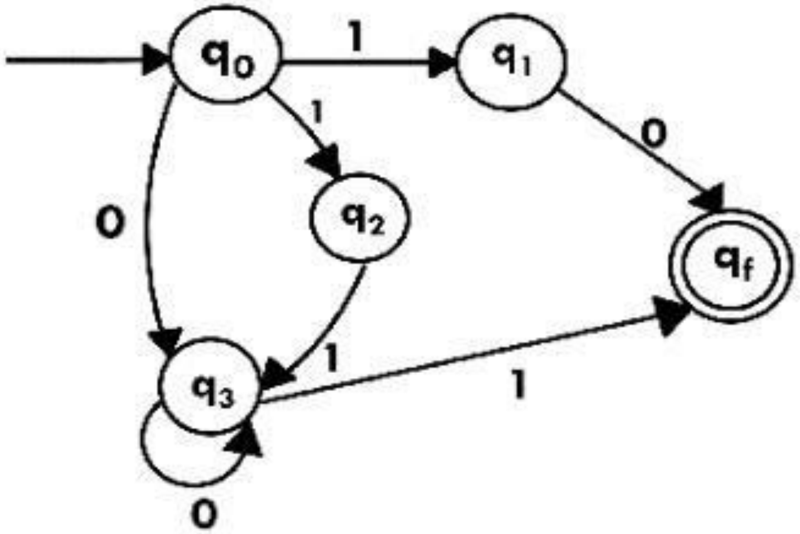
Step 3:



Step 4:



Step 5:



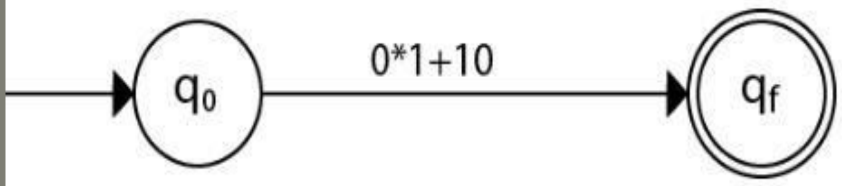
CONVERSION OF FINITE RE TO AUTOMATA :

Construct the FA for regular expression 0*1 + 10.

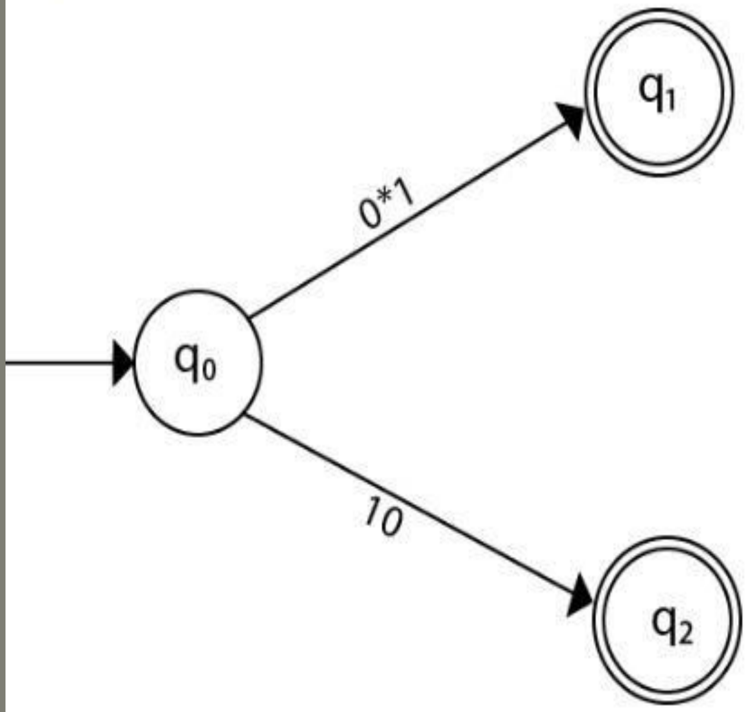
Solution:

We will first construct FA for R = 0*1 + 10 as follows:

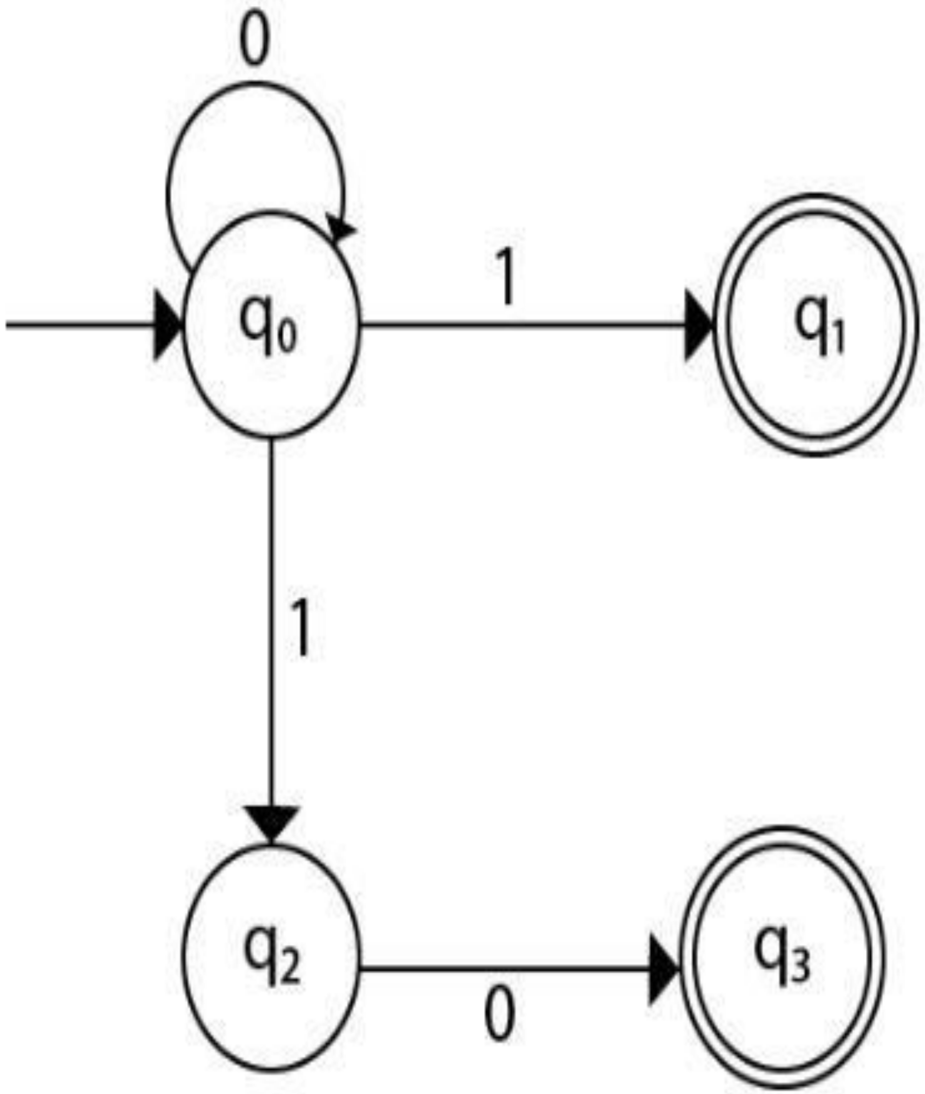
Step 1:



Step 2:



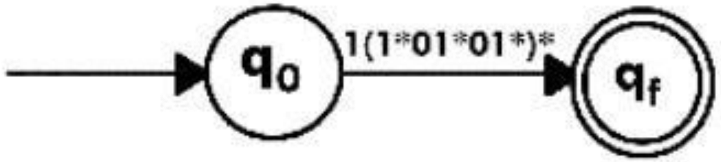
Step 4:



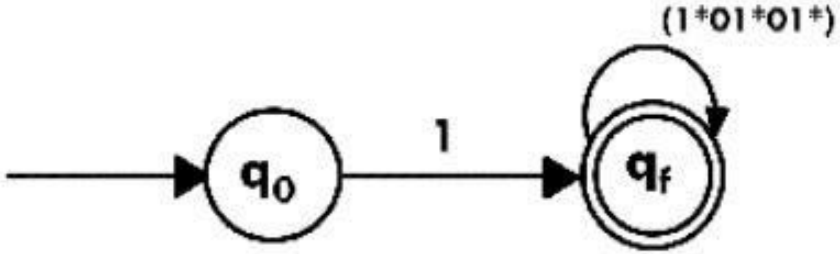
Design a NFA from given regular expression 1 (1* 01* 01*)*.

Solution: The NFA for the given regular expression is as follows:

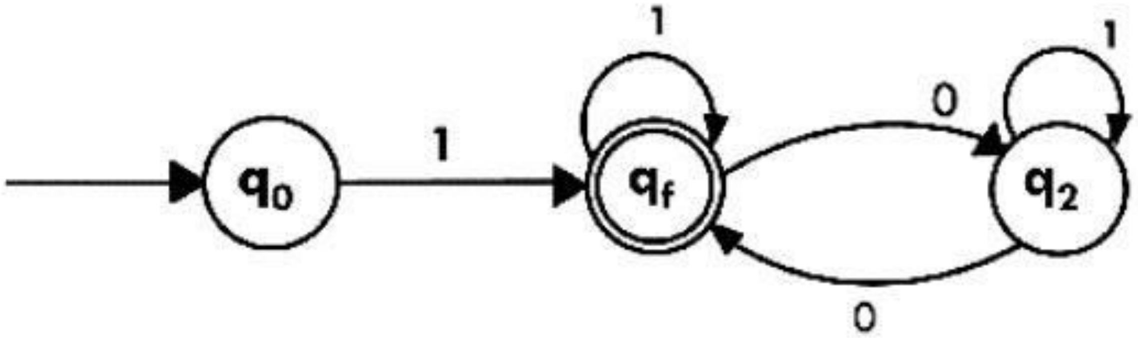
Step 1:



Step 2:



Step 3:



MATHEMATICAL FOUNDATION FOR COMPUTER SCIENCE

Prepared by: Er. Ankit Kharel

Nepal college of information technology

Finite State Automata

- *Sequential Circuits and Finite state Machine*
- *Finite State Automata*
- *Non-deterministic Finite State Automata*
- *Language and Grammars*
- *Language and Automata*
- *Regular Expression*

LANGUAGE AND GRAMMAR

- Merriam-Webster's Dictionary describes language as "the words, their pronunciation, and the methods of combining them used and understood by a community" .
- But this description of language is for natural languages • The rules of natural languages are very complex and difficult to characterize completely.
- Hence, comes the Formal language .
- Formal languages are used to model natural languages and to communicate with the computers .
- As it is possible to specify completely the rules by which certain formal languages are constructed .

LANGUAGE AND GRAMMAR

- Let A be a finite set of alphabets.
- A (formal) language L over A is a subset of A^*, the set of all strings over A.
- For example: Let A = {a, b}. The set L of all strings over A containing an odd number of a's is a language over A.
- One way to define a language is to give a list of rules that the language is assumed to obey(GRAMMAR)

GRAMMAR

A grammar is also called generator that can generate the language.

Let's consider Grammar,

$S \rightarrow aA$

$A \rightarrow aA/bA/\varepsilon$

Capital Symbols : Non- terminals

Small Symbols : Terminals

$\alpha \rightarrow \beta$ is known a production rules which means α can be written as β.

<u>Example:</u>

$S \rightarrow aA$	$S \rightarrow aA$	$S \rightarrow aA$
=a	=aA	=abA
	=abA	=ab
	=abaA	
	=aba	

If we use Grammar mentioned above we can make all string that starts with a.

$$L(G) = \{w|w \in \Sigma^*, S \xrightarrow{*} w\}$$

FORMAL DEFINITION OF GRAMMAR:

A phrase-structure grammar (or, simply, grammar) G is defined by quadruple,

$G=\{ N, T, P, \sigma \}$ where,

N = Finite set non-terminal symbols (Uppercase)

T = Finite non-empty set terminal symbols where(Lowercase)

P = Finite non-empty set of productions rules

σ = starting symbol $\sigma \in N$

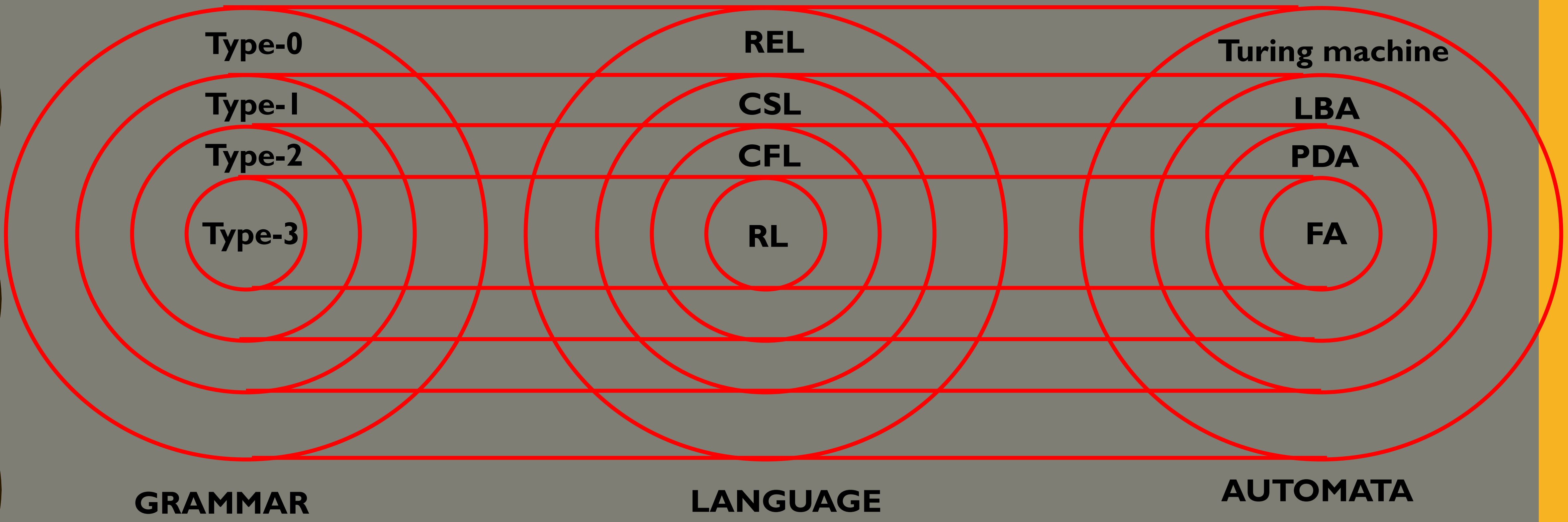
The production rule $\boldsymbol{\alpha} \rightarrow \boldsymbol{\beta}$ is valid if:

i) $\alpha \in (T \cup N)^{*} N (T \cup N)^{*}$ i.e. α must have at least one non-terminal symbol

ii) $\beta \in (T \cup N)^{*}$ i.e. β can consist of any combination of nonterminal and terminal symbols.

CHOMSKY HIERARCHY:

- Chomsky Hierarchy is a brand classification of the various types of grammar available. Grammars are classified by the form of their production category represents a class of languages that can be recognized by different automata.



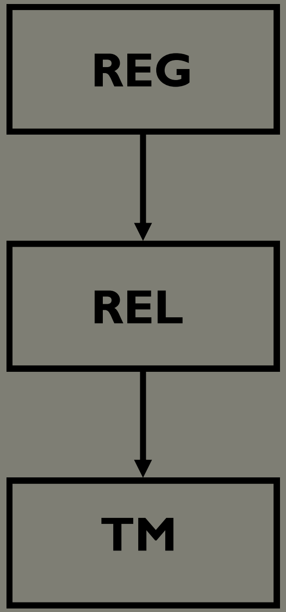
TYPE-0 (RECURSIVE ENUMERABLE GRAMMAR):

- Type – 0 Grammar(REG/Unrestricted grammar/Phase structured grammar) generates recursively enumerable language(REL). The production have no restriction. They generate the language that are recognized by a Turing Machine(TM).
- The production is in the form:

$$\alpha \rightarrow \beta \,;$$

$\alpha \in (T \cup N)^* \, N \, (T \cup N)^*$

$\beta \in (T \cup N)^*$



<u>Example:</u>

S→ACaB

Bc→acB

CB→DB

TYPE-1 (CONTEXT SENSITIVE GRAMMAR):

- Type – 1 Grammar(CSG/Length Increasing Grammar/Non-contracting grammar) generates Context Sensitive Language(CSL) which is accepted by Linearly Bounded Automata(LBA).

- The production is in the form:

$$\alpha \rightarrow \beta \,;$$

$\alpha \in (T \cup N)^{*}\, N\, (T \cup N)^{*}$

$\beta \in (T \cup N)^{*}$

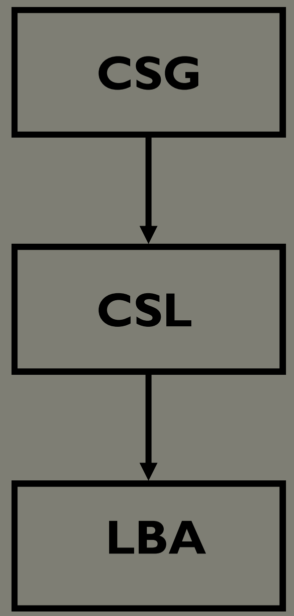
$|\alpha| \leq |\beta|$

<u>Example:</u>

AB→AbBc

A→bcA

B→a



Exception:

$S \rightarrow \varepsilon$

- S should be a start symbol but should not appear in RHS of production.

Example:

$$\left.\begin{aligned} &S \rightarrow aSb \\ &S \rightarrow \varepsilon \end{aligned}\right\} \textbf{Not allowed} \qquad \left.\begin{aligned} &S \rightarrow AB \\ &S \rightarrow \varepsilon \end{aligned}\right\} \textbf{Allowed}$$

$$\boldsymbol{\alpha A \beta} \rightarrow \boldsymbol{\alpha}\, \partial\, \boldsymbol{\beta}$$

Left Context **Right Context**

where $\boldsymbol{\alpha}$, $\boldsymbol{\beta} \in (N \cup T)^*$, $A \in N$ and $\partial \in (N \cup T)^*$ - the grammar is called context sensitive grammar.

<u>Example:</u>

$aAb \rightarrow aBb$

$cAd \rightarrow cCd$

TYPE-2 (CONTEXT FREE GRAMMAR):

- Type – 2 Grammar(CFG) generates Context Free Language(CFL) which is accepted by Push Down Automata(PDA).

- The production is in the form:

$$\alpha \to \beta \,;$$

$\alpha \in N$; $|\alpha| = 1$

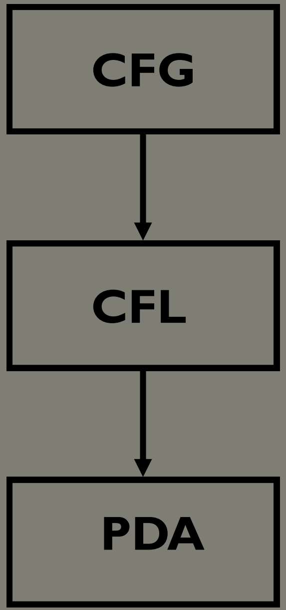
$\beta \in (T \cup N)^*$

<u>Example:</u>

S→Xa

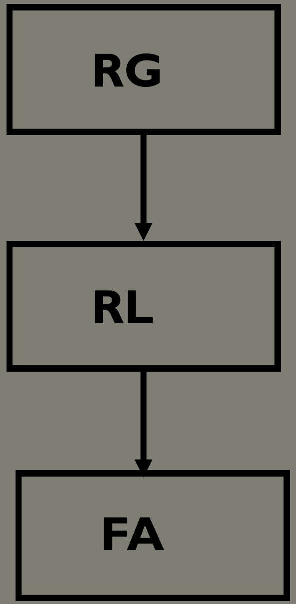
B→acB

C→a



TYPE-3 (REGULAR GRAMMAR):

- Type – 3 Grammar(RG) generates Reuglar Language(RL) which is accepted by Finite Automata(FA).



1. <u>Left Linear Grammar:</u>

$$A \rightarrow a$$
$$A \rightarrow Ba$$

- $A, B \in N$
- $|A| = |B| = 1$
- $a \in T^*$

<u>Example:</u>
A→abc
A→aBa(invalid)
A→Ca

1. <u>Right Linear Grammar:</u>

$$A \rightarrow a$$
$$A \rightarrow aB$$

- $A, B \in N$
- $|A| = |B| = 1$
- $a \in T^*$

<u>Example:</u>
A→a
A→aBa(invalid)
A→Ca(invalid)
A→aC

Q. Consider the following Grammar:

$S \rightarrow ACaB$
$Bc \rightarrow acB$
$CB \rightarrow DB$
$aD \rightarrow Db$

Determine whether the given grammar is Context-sensitive, Context-Free, Regular or None of these.

Solution:

The Given grammar is:

$S \rightarrow ACaB$
$Bc \rightarrow acB$
$CB \rightarrow DB$
$aD \rightarrow Db$

(a)Checking For Regular(Type-3)

The production rule for regular grammar is given by,

$$A \rightarrow a$$
$$A \rightarrow Ba$$
$$A, B \in N$$
$$|A| = |B| = 1$$
$$a \in T^*$$

Since the production, $S \rightarrow ACaB$ violates the rule, It is not REGULAR GRAMMAR (b)Checking For Context- Free(Type-2)

The production rule for Context-Free grammar is given by,

$$\boldsymbol{\alpha \rightarrow \beta \,;}$$

$$\boldsymbol{\alpha} \in N \,;\, \boldsymbol{|\alpha| = 1}$$

$$\boldsymbol{\beta} \in (T \cup N)^*$$

Since the production, Bc → acB violates the rule, It is not CONTEXT FREE GRAMMAR.

(c)Checking For Context- Sensitive (Type-1)

The production rule for Context-Sensitive grammar is given by,

$$\boldsymbol{\alpha \rightarrow \beta \,;}$$

$$\boldsymbol{\alpha} \in (T \cup N)^* \, N \, (T \cup N)^*$$

$$\boldsymbol{\beta} \in (T \cup N)^*$$

$$\boldsymbol{|\alpha|} \leq \boldsymbol{|\,\beta\,|}$$

Ever production given in the grammar satisfies above rule, Therefore, it is

CONTEXT SENSITIVE GRAMMAR

DERIVATION TREE FOR CFG:

- A derivation Tree or Parse Tree is an ordered rooted tree that graphically represents the semantic information of string derived from a Context Free Grammar.

  1. Root Vertex : Must be labelled by start symbol
  2. Vertex : Labelled by Non- Terminal symbols
  3. Leaves : Labelled by Terminal Symbols

- Consider the following grammar:

  G={ N ,T , P , σ } where Production rule is given by:

  S→0B

  A→1AA/ ε

  B→0AA

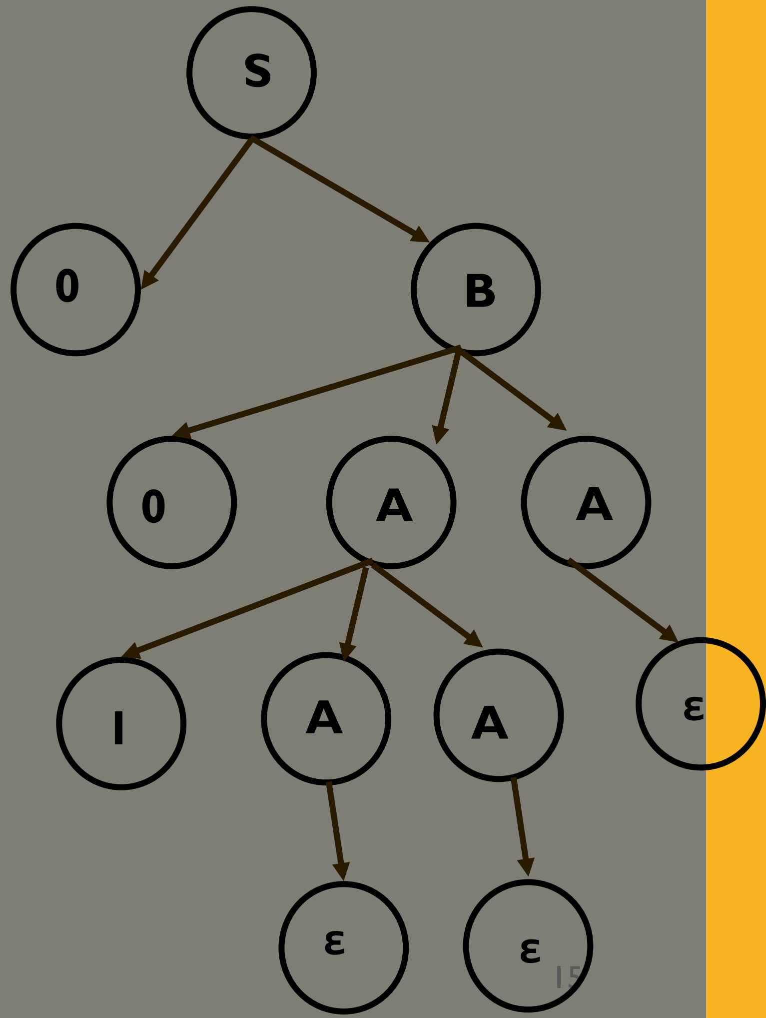
Construct Derivation Tree for the string "001"

S→0B

00AA

001AAA

001



DERIVATION TREE FOR CFG:

1. LEFTMOST DERIVATION:

A leftmost Derivation Tree is obtained by applying production function to the leftmost variable in each step.

Consider the following grammar:

G={ N ,T , P , σ } where Production rule is given by:

S→aAS/aSS/e

A→SbA/ba

Construct Derivation Tree for the string "aabaa"

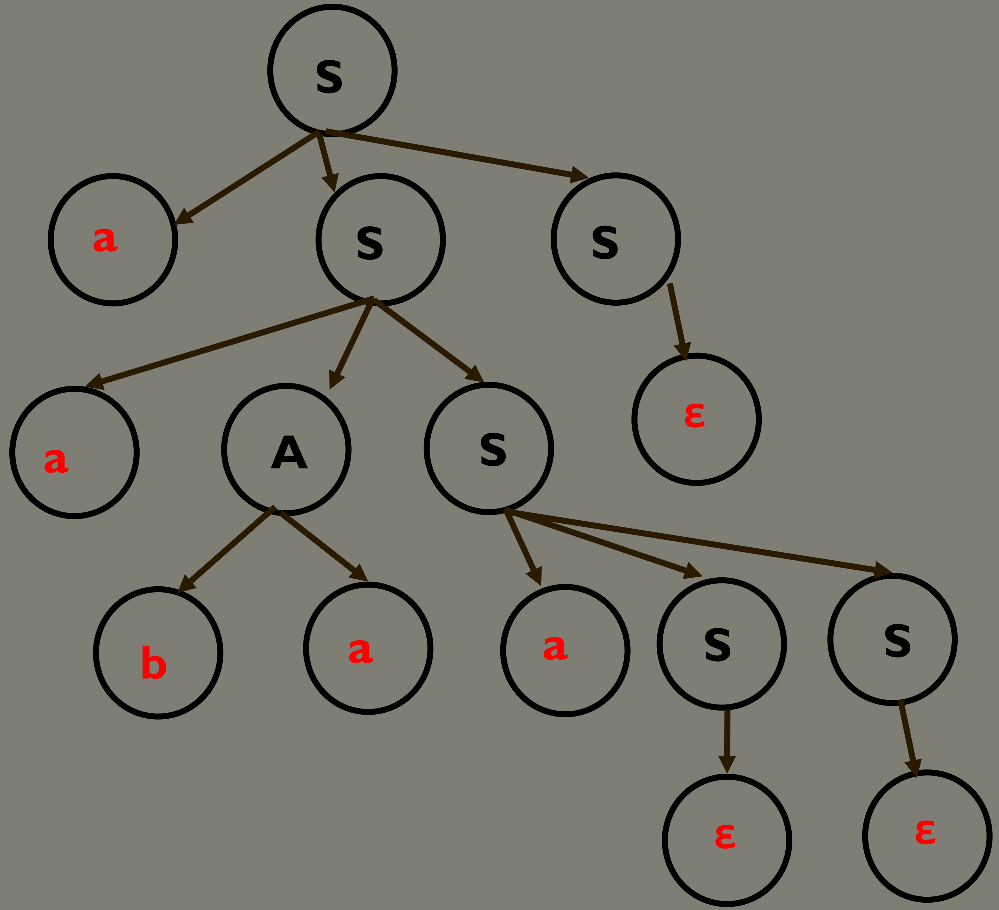
S→aSS

aaASS

aabaSS

aabaaSSS

aabaa



DERIVATION TREE FOR CFG:

2. **RIGHTMOST DERIVATION:**

   A rightmost Derivation Tree is obtained by applying production function to the rightmost variable in each step.

Consider the following grammar:

G={ N ,T , P , σ } where Production rule is given by:

S→aAS/aSS/e

A→SbA/ba

Construct Derivation Tree for the string "aabaa"

S→aSS

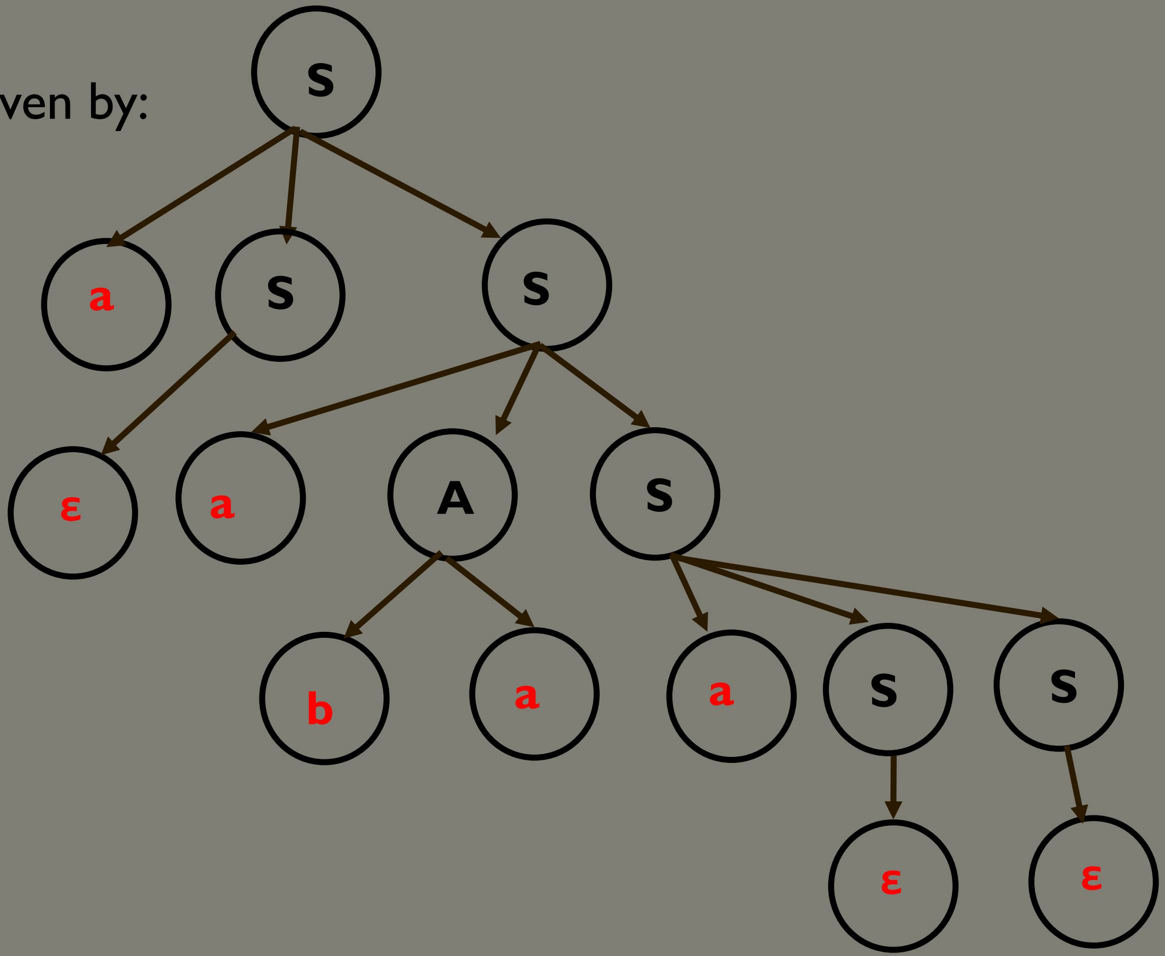
aSaAS

aSaAaSS

aSaAa

aSabaa

aabaa



DERIVATION TREE FOR CFG:

G={ N ,T , P , σ } where Production rule is given by:

S→aB/bA

A→a/aS/bAA

B→b/bS/aBB

Construct left Derivation Tree for the string "aabbabba"

S→aB

aaBB

aabSB

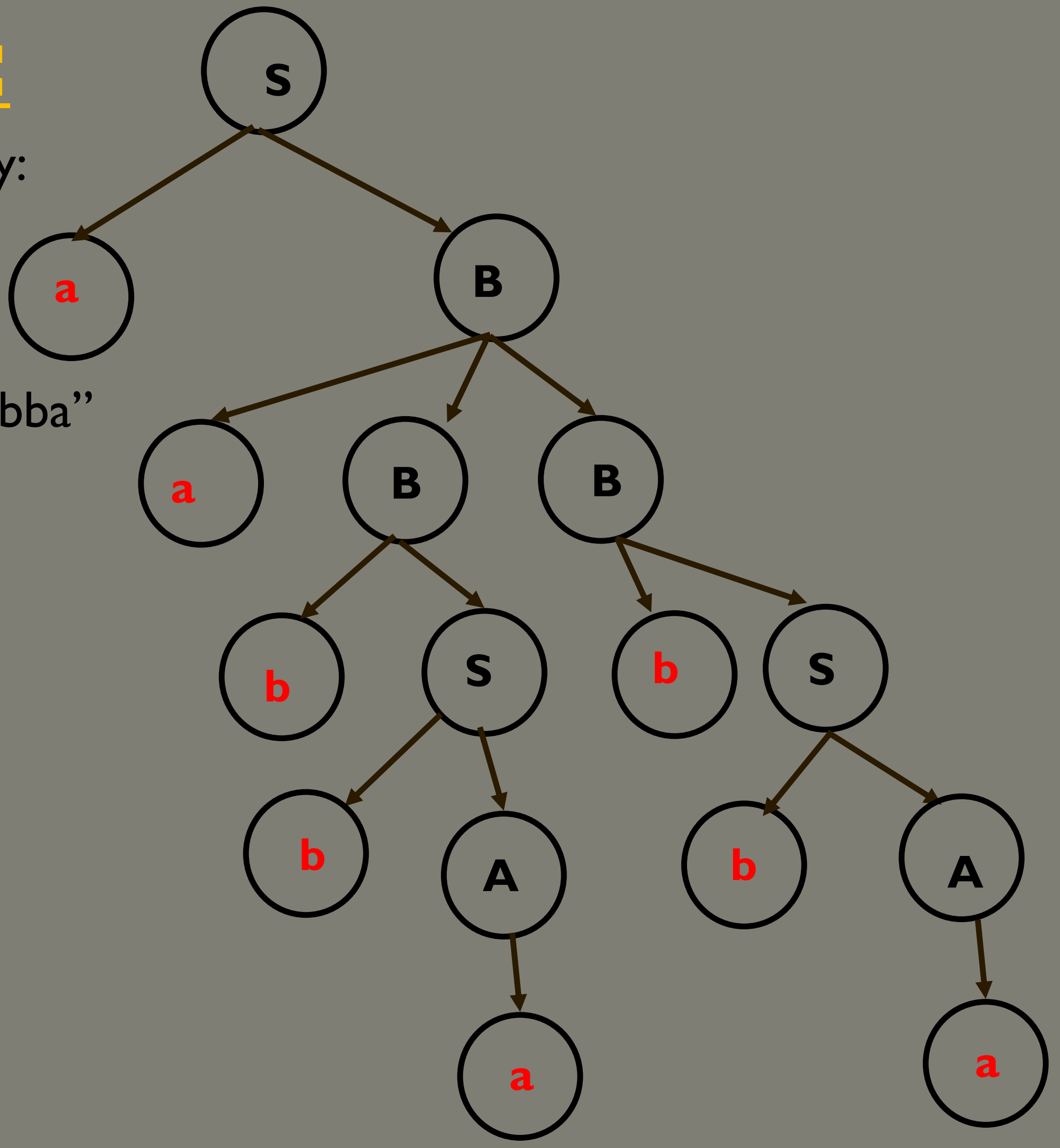
aabbAB

aabbaB

aabbabS

aabbabbA

aabbabba



DERIVATION TREE FOR CFG:

G={ N ,T , P , σ } where Production rule is given by:

S→aB/bA

A→a/aS/bAA

B→b/bS/aBB

Construct right Derivation Tree for the string "aabbabba"

S→aB

aaBB

aaBbS

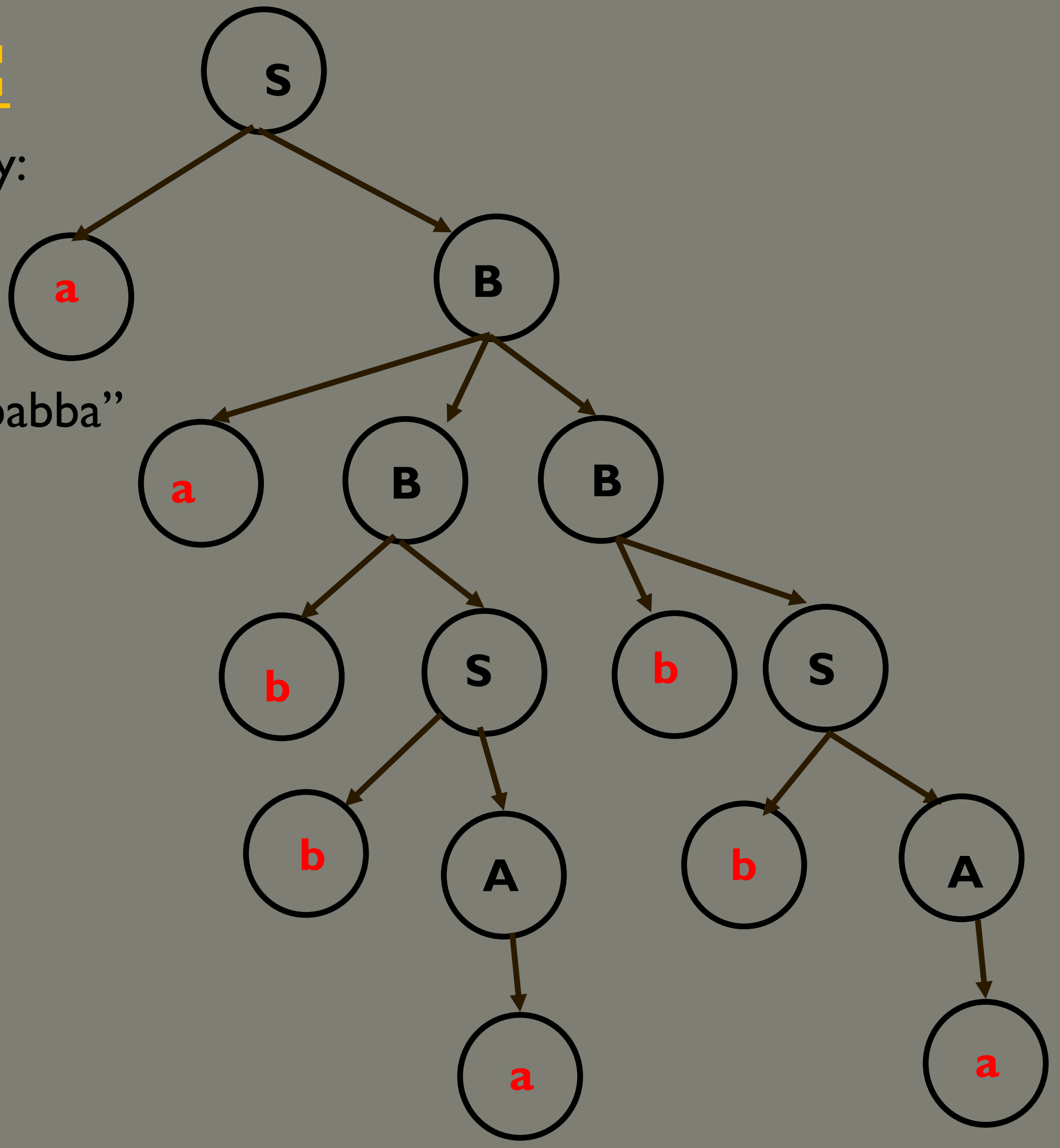
aaBbbA

aaBbba

aabSbba

aabbAbba

aabbabba



DERIVATION TREE FOR CFG:

Consider the grammar G={ N , T , P , σ } where
Production rule is given by:

E→E + E

E→E * E

E→(E)

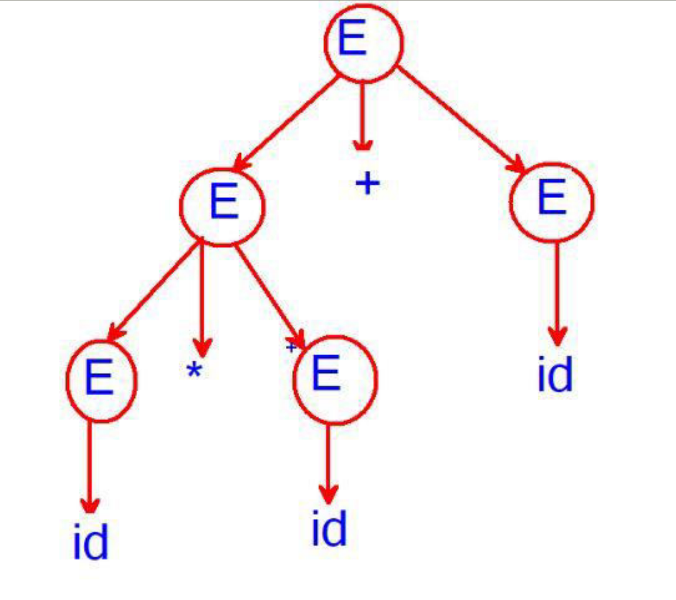
E→id

Construct Derivation Tree for the **id * id + id**

E→E + E

E * E + E

id * id + id



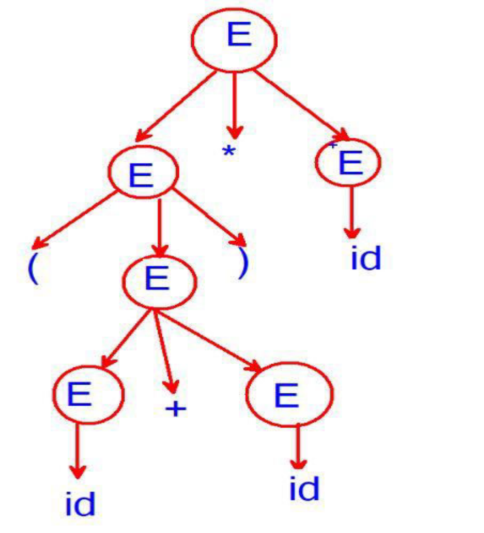
Construct Derivation Tree for the **(id + id) * id**

E→E * E

(E) * E

(E + E) * E

(id + id) * id



BNF(BACKUS NORMAL FORM):

- An alternative way to state of productions of a grammar is by using Backus Normal Form(BNF). It is meta syntax for CFG.

Syntax:

<LHS> : : = RHS

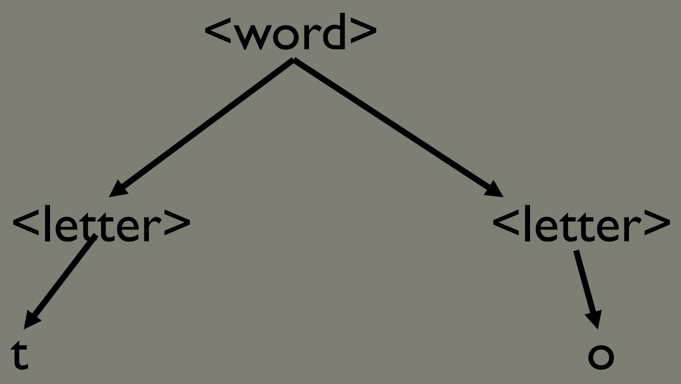
(Non – terminals) *(Terminals)*

Example:

<letter> : : = a/b/c/d/e/t/o

<word> : := <letter><letter>

(This generates word consisting of two letter)



BNF(BACKUS NORMAL FORM):

- **Grammer for integers:**

  An integer is defined as a string consisting of an optional sign(+ or -) followed by a string of digits(0 though 9)

The following Grammar generates all string:

<digit> : : = 0/1/2/3/4/5/6/7/8/9
<sign> : : = +/-
<unsigned integer> : : = <digit>/<digit><unsigned integer>
<signed integer> : : = <sign><unsigned integer>
<integer> : : = <signed integer>/<unsigned integer>

Derive integer -102 using above grammar and construct derivation tree.

BNF(BACKUS NORMAL FORM):

<digit> : : = 0/1/2/3/4/5/6/7/8/9

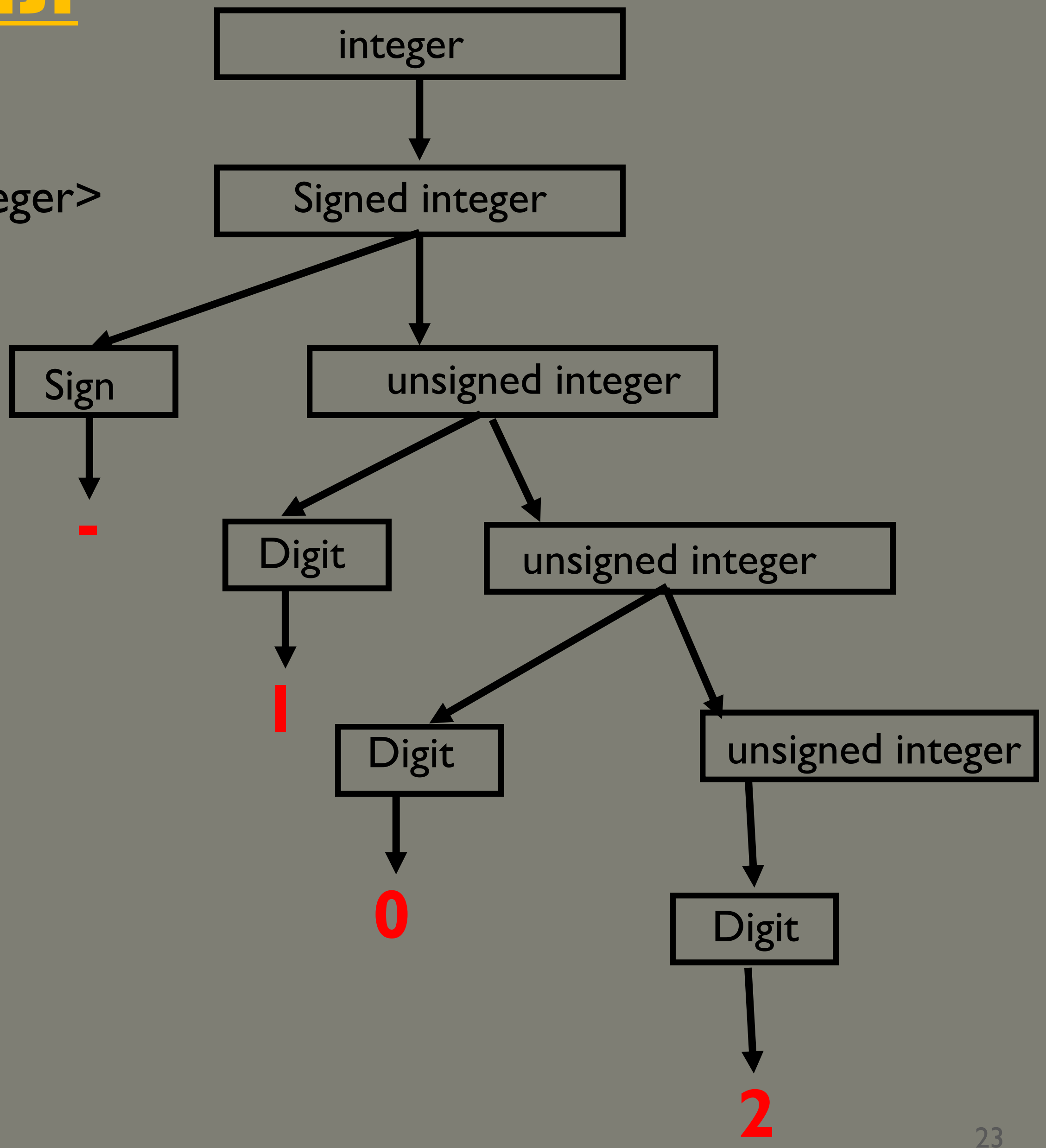
<sign> : : = +/-

<unsigned integer> : : = <digit>/<digit><unsigned integer>

<signed integer> : : = <sign><unsigned integer>

<integer> : : = <signed integer>/<unsigned integer>

<integer> : : = <signed integer>
<sign><unsigned integer>
-<digit><unsigned integer>
-1<digit><unsigned integer>
-10<digit>
-102



MATHEMATICAL FOUNDATION FOR COMPUTER SCIENCE

Prepared by: Er. Ankit Kharel

Nepal college of information technology

Finite State Automata

- *Sequential Circuits and Finite state Machine*
- *Finite State Automata*
- *Non-deterministic Finite State Automata*
- *Language and Grammars*
- *Language and Automata*
- *Regular Expression*

GRAMMAR TO FINITE AUTOMATA:

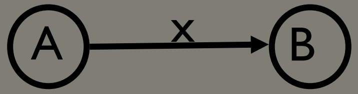
- For each Right Linear Grammar(G_r), there is one finite automata M where L(M) = L(G_R).

1. **Right Linear Grammar to NFA:**

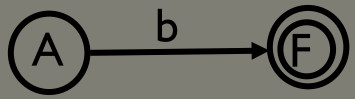
(a) The non- terminals becomes states with σ as an initial state.

(b) The terminal becomes set of alphabets(input)

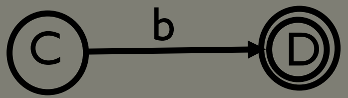
(c) The production of form, A→xB, we draw an edge from state A to B and label it with x.



(d) The production of form C→b is written as C→bF where F is final state.



(e) The production of form C→b, C→bD:



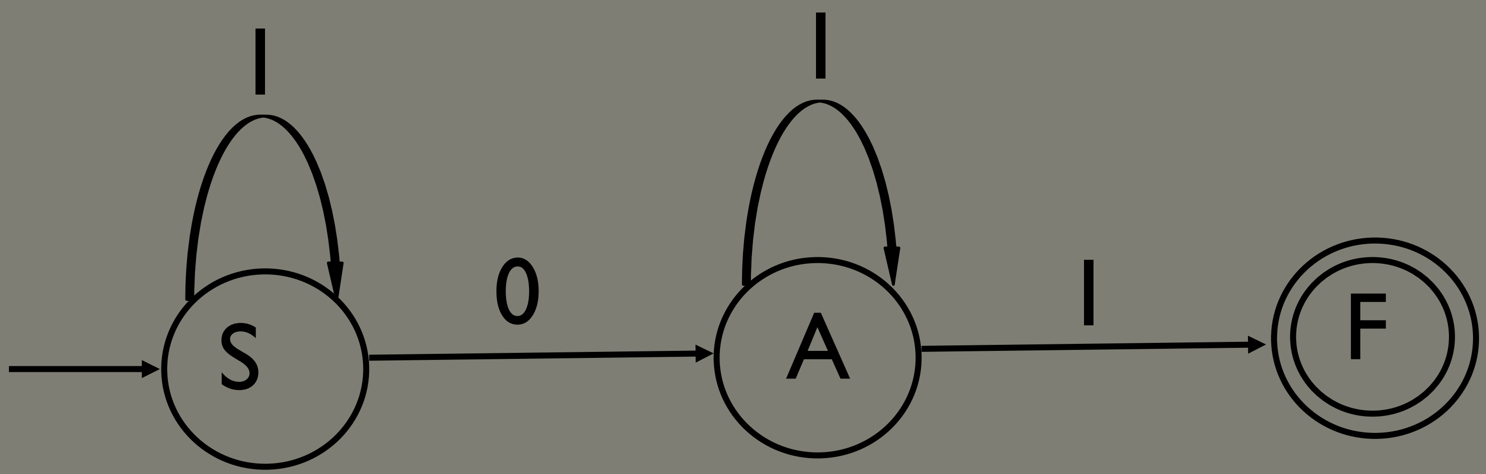
GRAMMAR TO FINITE AUTOMATA:

- Construct a non- deterministic finite automata that recognizes the language generated by the regular grammar, G={N, T, P, σ} where N={A , S} , T={0 , 1}, S is starting symbol and production P are:

  S→1S/0A

  A→1A/1

Solution:



GRAMMAR TO FINITE AUTOMATA:

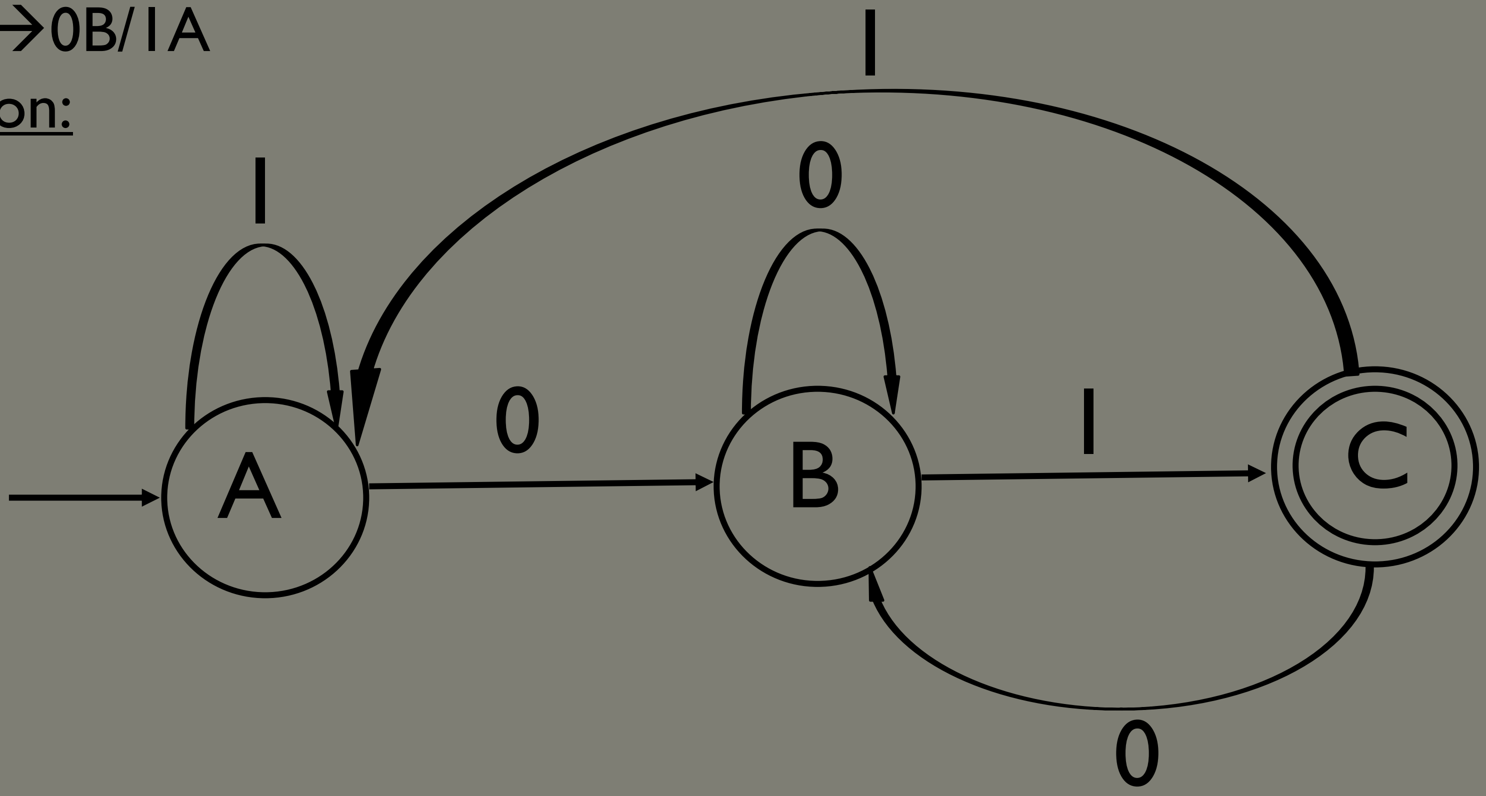
- Construct a non- deterministic finite automata that recognizes the language generated by the regular grammar, G={N, T, P, σ} where N={A , B, C} ,T={0 , 1}, A is starting symbol and production P are:

  A→0B/1A

  B→0B/1C/1

  C→0B/1A

Solution:



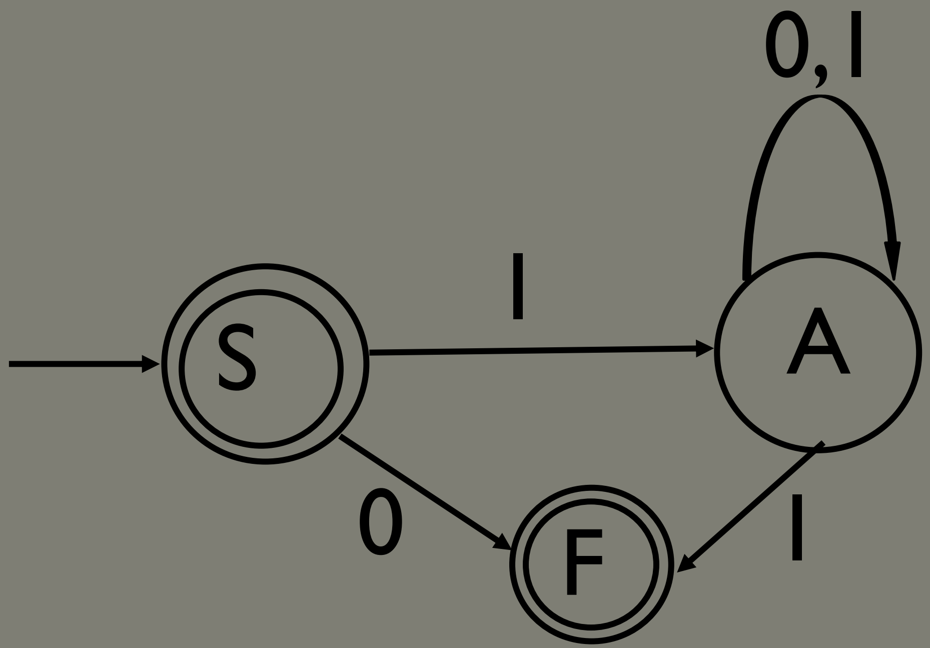
GRAMMAR TO FINITE AUTOMATA:

- Construct a non- deterministic finite automata that recognizes the language generated by the regular grammar, G={N, T, P, σ} where N={A , S} , T={0 , 1}, S is starting symbol and production P are:

  $S \rightarrow 1A/0/\epsilon$

  $A \rightarrow 0A/1A/1$

Solution:



GRAMMAR TO FINITE AUTOMATA:

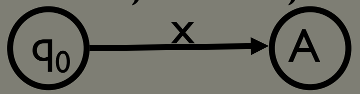
- For a given left linear grammar there is a corresponding finite automata M where L(M) = L(G_L).

2. LEFT Linear Grammar to NFA:

(a) Start symbol is the final state.

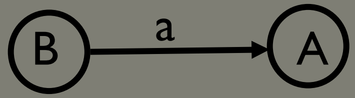
(b) The terminal becomes set of alphabets(input) and non terminal becomes states.

(c) The production of form, A→x,



where, q_0 is an initial state.

(d) The production of form A→Ba



GRAMMAR TO FINITE AUTOMATA:

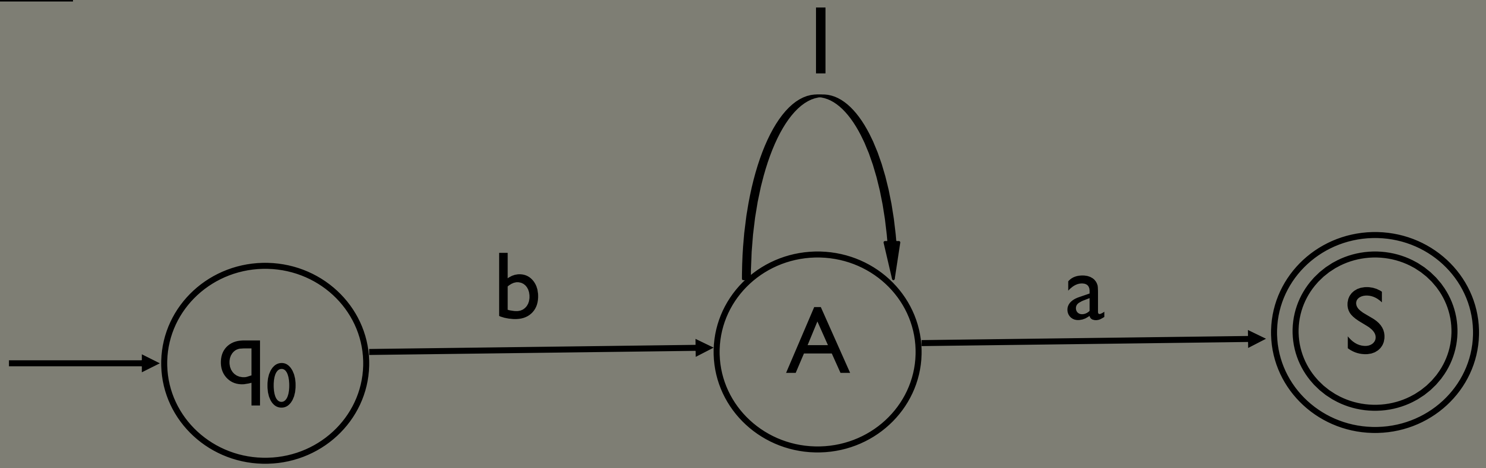
- Construct a non- deterministic finite automata that recognizes the language generated by the regular grammar, G={N, T, P, σ} where N={S , A} ,T={a , b}, S is starting symbol and production P are:

  S→Aa

  A→Aa

  A→b

Solution:



GRAMMAR TO FINITE AUTOMATA:

- Construct a non- deterministic finite automata that recognizes the language generated by the regular grammar, G={N, T, P, σ} where N={S , A, B, C} ,T={a , b}, S is starting symbol and production P are:

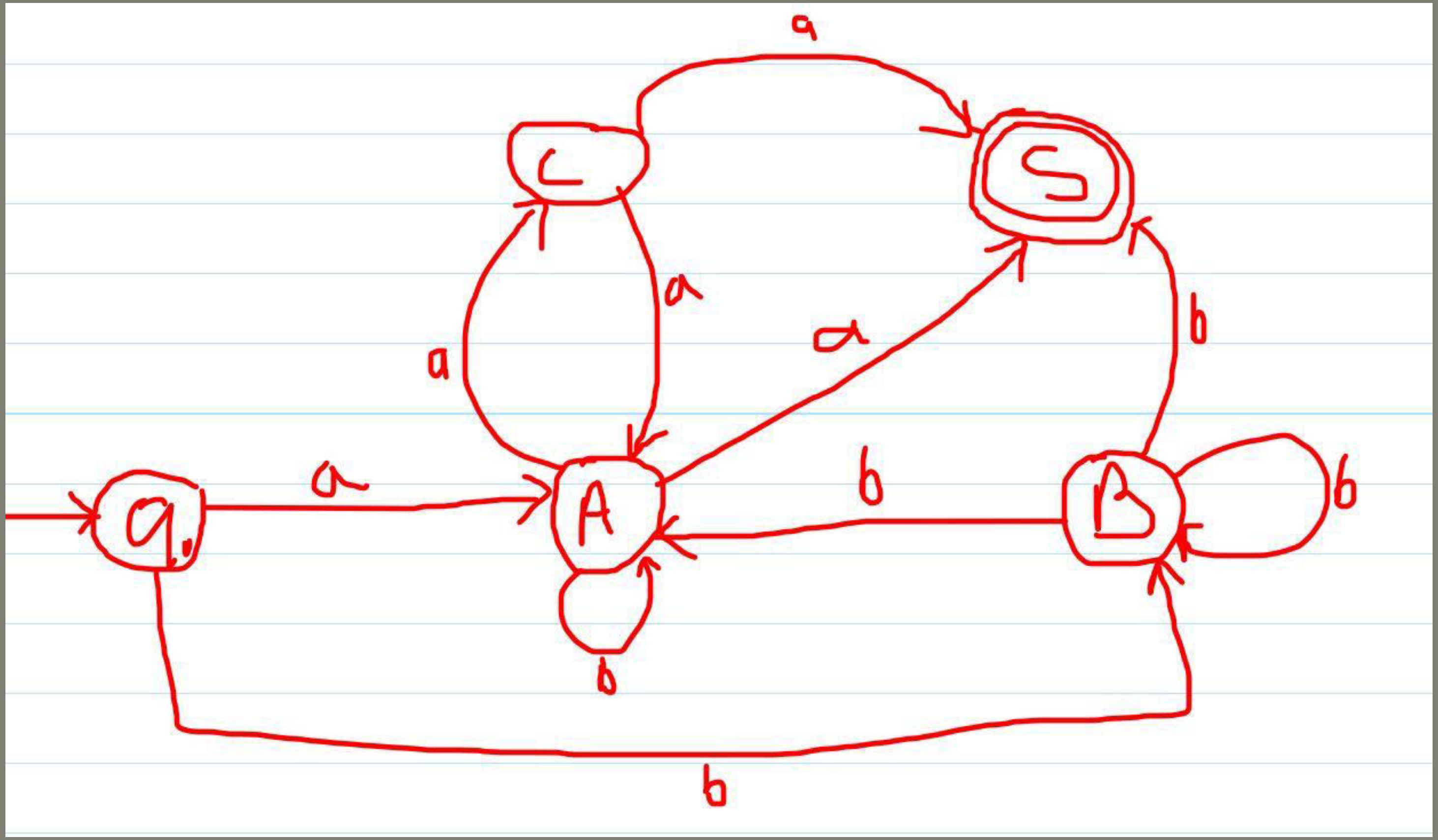
  S→Ca/Aa/Bb

  A→Aa/Ca/Bb/a

  B→Bb/b

  C→Aa

Solution:



FINITE AUTOMATA TO GRAMMAR :

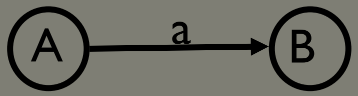
- For each Finite Automata M, there is one right linear grammar G_R where $L(G_R) = L(M)$.

1. **Finite Automata to Right Linear Grammar:**

(a) The set of states becomes non terminal symbols.

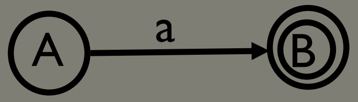
(b) The set of inputs becomes terminal symbols.

(c) Rule 1:



$= A \rightarrow aB$

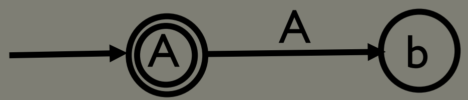
(d) Rule 2:



$= A \rightarrow aB/a$

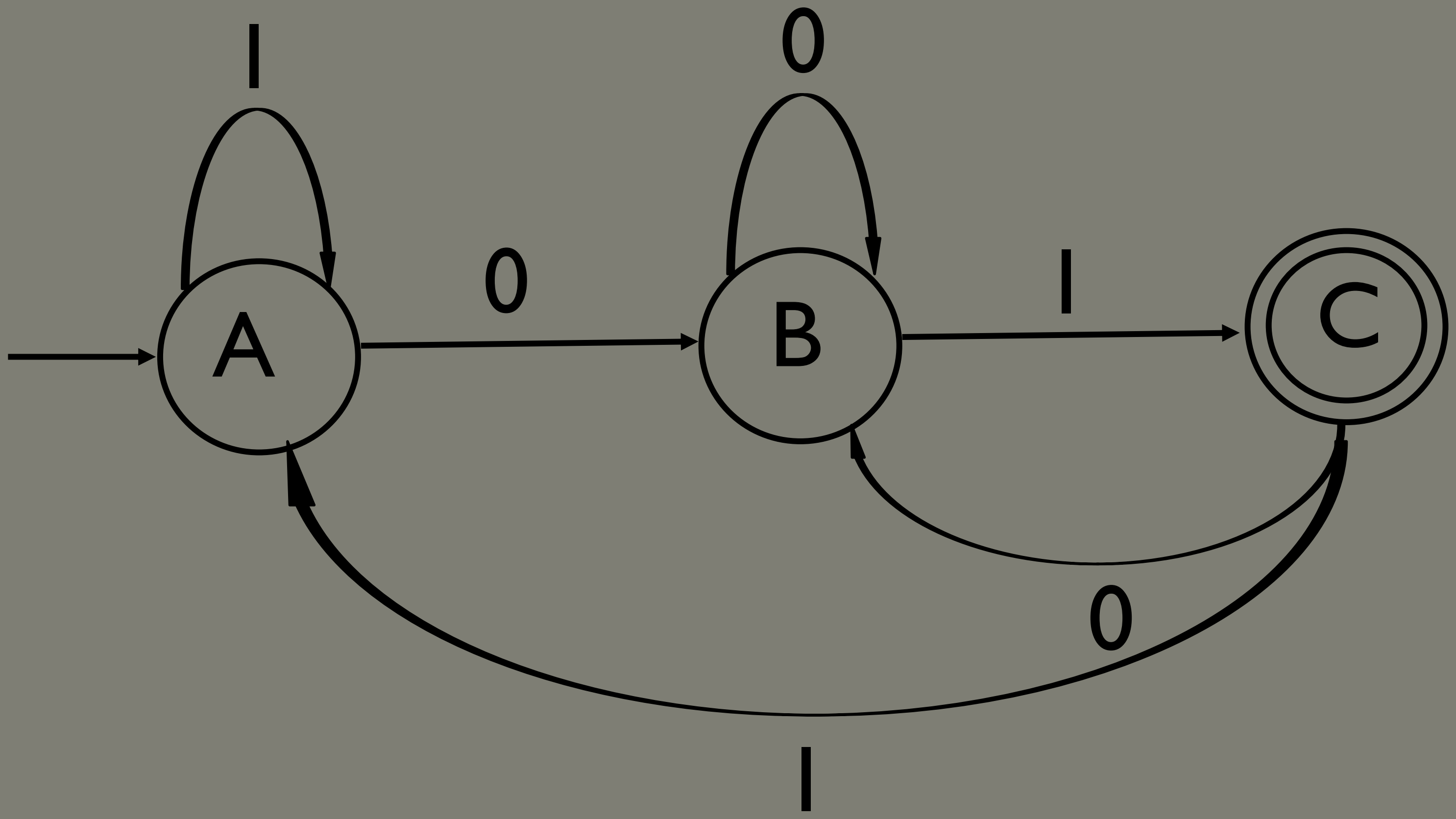
(e) Rule 3:

If initial state is final state then add eplision in a production



$= A \rightarrow aB/ \in$

GRAMMAR TO FINITE AUTOMATA:



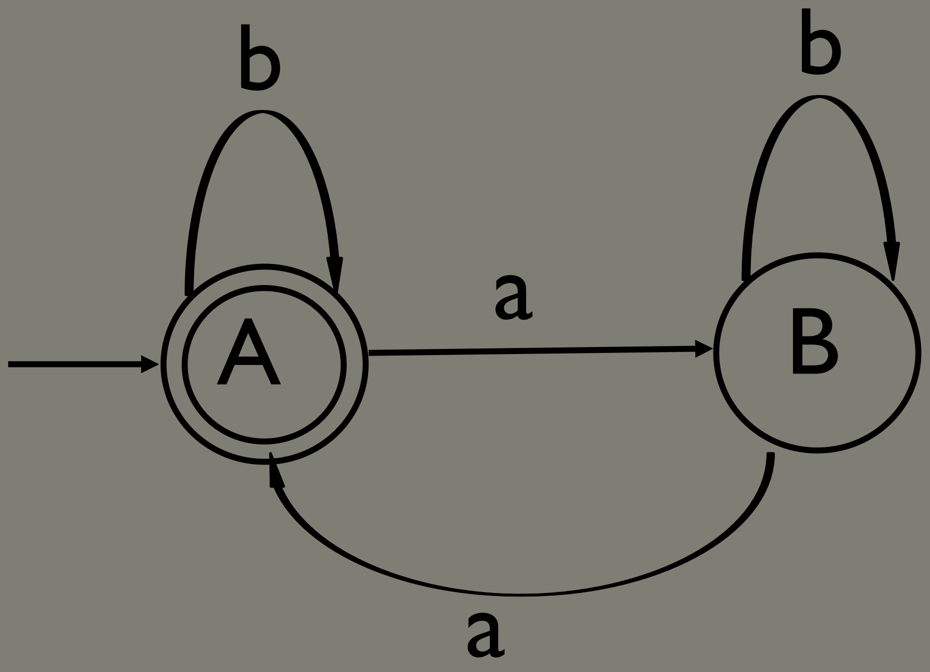
A→0B/1A

B→0B/1C/1

C→0B/1A

(A is starting symbol)

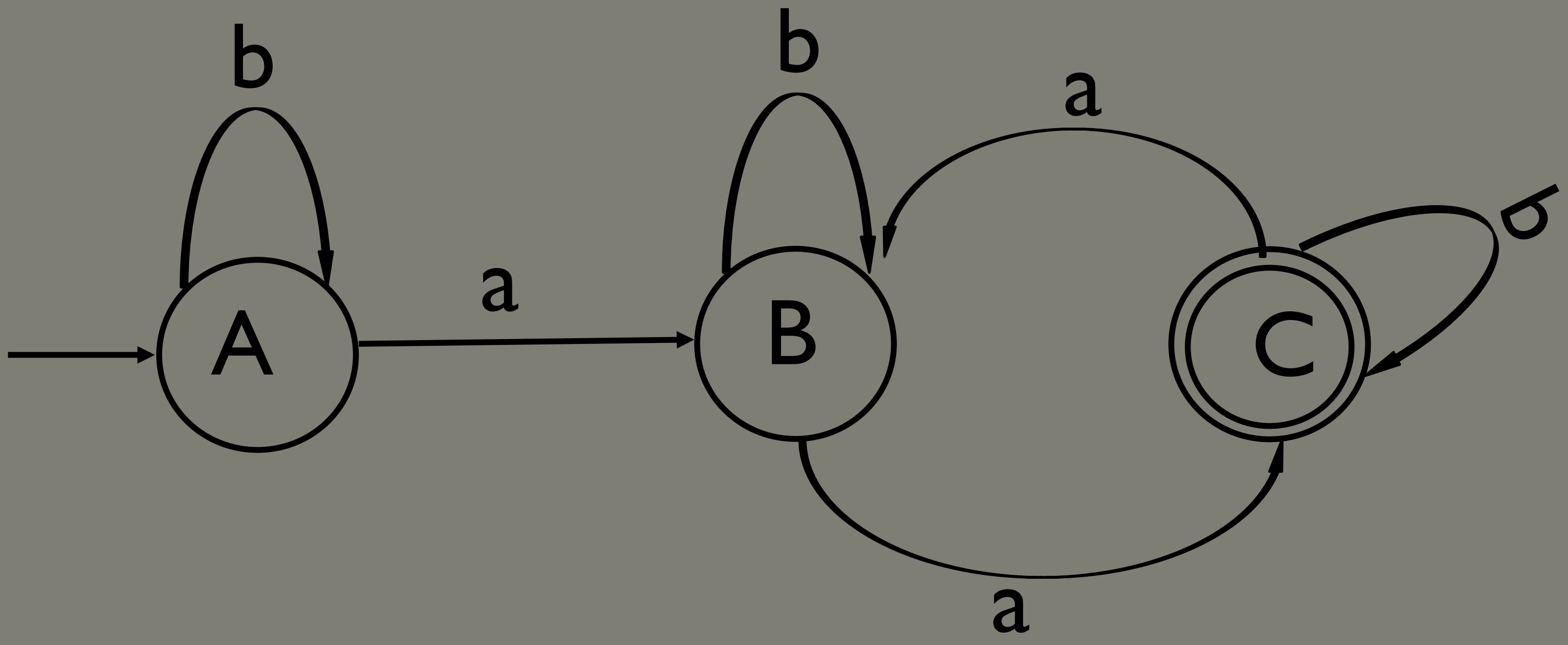
GRAMMAR TO FINITE AUTOMATA:



A→aB/bA/b/∈ (A is starting symbol)

B→bB/aA/a

GRAMMAR TO FINITE AUTOMATA:



A→aB/bA (A is starting symbol)
B→bB/aC/a
C→bC/aB/b

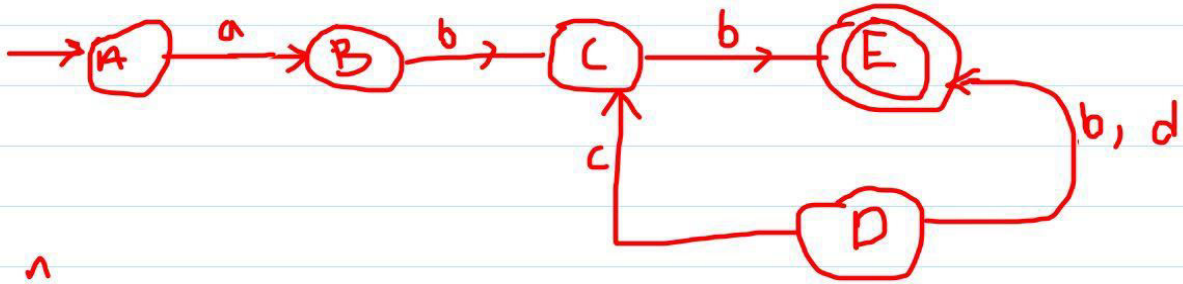
FINITE AUTOMATA TO GRAMMAR :

- For each Finite Automata M, there is one left linear grammar G_L where $L(G_L)$ = L(M).

1. **Finite Automata to Left Linear Grammar:**

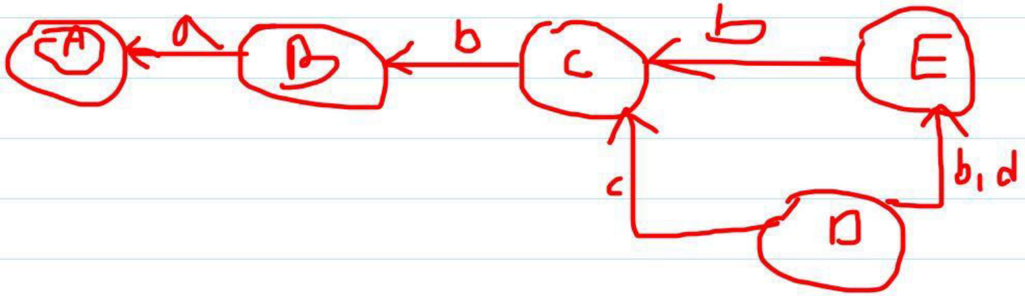
(a) The set of states becomes non terminal symbols.
(b) The set of inputs becomes terminal symbols.
(c)Reverse the edges of NFA and exchange initial and Final state.
(d)Construct Right Linear Grammar
(e)Reverse the production and obtain left linear grammar.

(B)



Sol^n

First Reverse the edges and initial and final state



Obtain right linear grammer

$E \rightarrow bC / bD / dD$
$C \rightarrow bB$
$B \rightarrow aA / a$
$D \rightarrow cC$

$\xrightarrow{\text{Reverse}}$

$E \rightarrow Cb / Db / Dd$
$C \rightarrow Bb$
$B \rightarrow Aa / a$
$D \rightarrow Cc$

MATHEMATICAL FOUNDATION FOR COMPUTER SCIENCE

Prepared by: Er. Ankit Kharel

Nepal college of information technology

FINITE STATE AUTOMATA

- *Sequential Circuits and Finite state Machine*
- *Finite State Automata*
- *Non-deterministic Finite State Automata*
- *Language and Grammars*
- *Language and Automata*
- *Regular Expression*

GENERATING GRAMMAR FOR THE LANGUAGE:

1. **First write the Regular Expression for the language.**

Some Basic Rules:

a^* $A \to aA/\epsilon$	$(a+b)^*$ $A \to aA/bA/\epsilon$	$(a+b)^+$ $A \to aA/bA/a/b$

Q. Write the Grammar that generates string having Following Properties:

a) String of exactly length two

R.E. = $(a+b)(a+b)$

$S \rightarrow AA$

$A \rightarrow a/b$

b) String of at most length 2

R.E. = $(a+b+\in)(a+b+\in)$

$S \rightarrow AA$

$A \rightarrow a/b/\in$

c) Starts with a

R.E. = $a\ (a+b)^*$

$S \rightarrow aA$

$A \rightarrow aA/bA/\in$

c) Ends with ba

R.E. = $(a+b)^*ba$

$S \rightarrow Aba$

$A \rightarrow aA/bA/\in$

d) Stats with a and ends with b

R.E. = $a(a+b)^*b$

$S \rightarrow aAb$

$A \rightarrow aA/bA/\ \in$

e) Starts and ends with same symbol

R.E. = $a(a+b)^*a + b(a+b)^*b$

$S \rightarrow aAa/bAb/a/b$

$A \rightarrow aA/bA/\in$

f) Starts and ends with different symbol

R.E. = $a(a+b)^*b + b(a+b)^*a$

$S \rightarrow aAb/bAa$

$A \rightarrow aA/bA/\in$

c) Ends with ba

R.E. = $(a+b)^*ba$

$S \rightarrow Aba$

$A \rightarrow aA/bA/\in$

Q. Write the Context Free Grammar that generates Palindrome strong over $\sum(a, b)$

S→∈/a/b
S→aSa/bSb

Properties of regular language;

1. If L and M are regular languages, then L ∪ M(**UNION)** is a regular language.
   Let L and M be the languages of regular expressions R and S, respectively. Then R+S is a regular expression whose language is L U M.

2. If L is regular languages, then L^* **(Kleen Closure)**is a regular language.
   Let L the languages of regular expressions R. Then R^* is a regular expression whose language is L^*

3. If L and M are regular languages, then L.M**(Concatenation)** is a regular language.
   Let L and M be the languages of regular expressions R and S, respectively. Then R.S is a regular expression whose language is L.M.

4. If L is a regular language over Σ, then $\bar{L}$ **(Complement)** is also a regular language.
   Construct a DFA for $L.$ This can be transformed into a DFA for $\bar{L}$ by making all accepting states non-accepting and vice versa.

5. If L and M are regular languages, then L ∩ M**(Intersection)** is a regular language.
   RE_1 = a(a*) and RE_2 = (aa)*
   So, L_1 = { a,aa, aaa, aaaa, ...} (Strings of all possible lengths excluding Null)
   L_2 = { ε, aa, aaaa, aaaaaa,.......} (Strings of even length including Null)
   $L_1 \cap L_2$ = { aa, aaaa, aaaaaa,.......} (Strings of even length excluding Null)
   RE ($L_1 \cap L_2$) = aa(aa)* which is a regular expression itself.

Properties of regular language;

6. If L and M are regular, then so is L − M**(Difference)**.

   Proof: $M - N = M \cap \overline{N}$

7. If L is regular, then so is L^R**(Reversal)**.

   Let, L = {01, 10, 11, 10}

   RE (L) = 01 + 10 + 11 + 10

   L^R = {10, 01, 11, 01}

   RE (L^R) = 01 + 10 + 11 + 10 which is regular

MATHEMATICAL FOUNDATION FOR COMPUTER SCIENCE

Prepared by: Er. Ankit Kharel

Nepal college of information technology

GRAPH THEORY

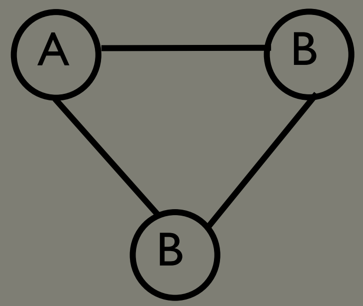
GRAPHS:

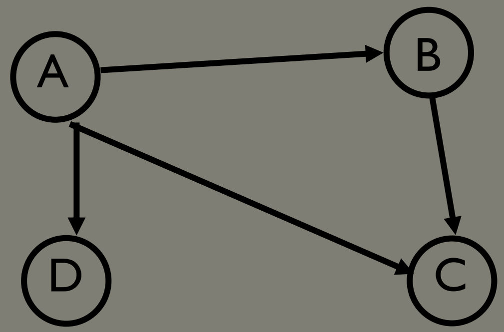
- Graphs are discrete structures consisting of vertices and edges that connect these vertices.

- There are different kinds of graphs, depending on whether edges have directions, whether multiple edges can connect the same pair of vertices, and whether loops are allowed.

- **A graph G = (V , E) consists of V , a nonempty set of vertices (or nodes) and E, a set of edges. Each edge has either one or two vertices associated with it, called its endpoints. An edge is said to connect its endpoints.**

- The set of vertices V of a graph G may be infinite. A graph with an infinite vertex set or an infinite number of edges is called an infinite graph, and in comparison, a graph with a finite vertex set and a finite edge set is called a finite graph. I

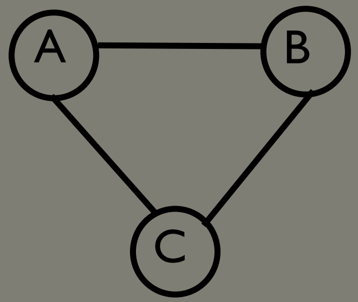




TYPES OF GRAPHS:

1. **Undirected Graph:** A Graph whose edges are undirected is called undirected Graph

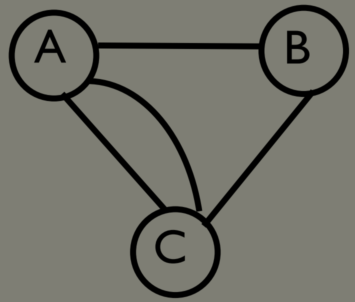
   a. *Simple Graph:* A graph in which each edge connects to the two different vertices and no two edges connect same pair of vertices is called a Simple Graph[no- parallel edges and no loops]



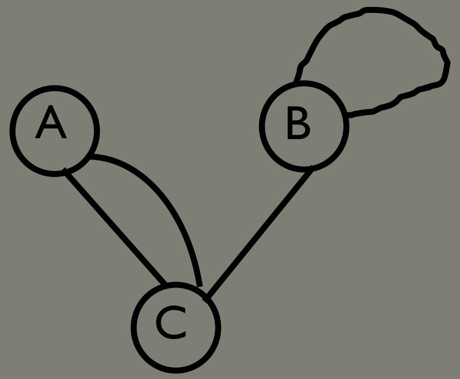
Maximun number of edges possible in a simple graph with n vertices is

$$\frac{n(n-1)}{2}$$

   b. *Multi Graph:* If in a graph multiple edges between the same set of vertices are allowed, it is called multigraph.



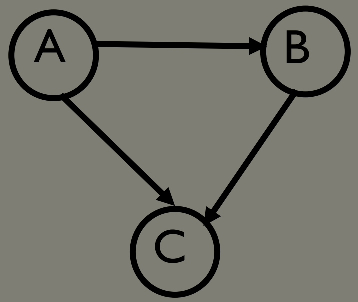
   c. *Pseudo Graph* : It is a multigraph with loops.



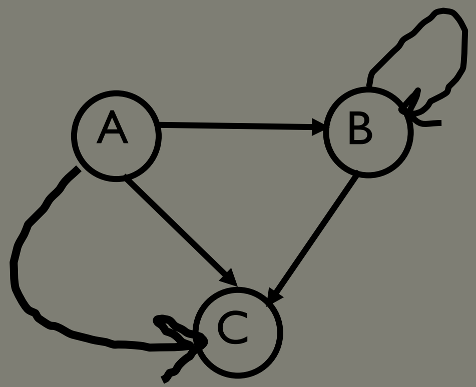
TYPES OF GRAPHS:

1. **Directed Graph:** A Graph whose edges are directed is called directed Graph

   a. *Simple Directed Graph*: A directed graph in which each edge connects to the two different vertices and no two edges connect same pair of vertices is called a Simple Graph[no- parallel edges and no loops]



   b. *Multiple Directed Graph*: If in a Directed graph multiple edges between the same set of vertices and loops are allowed, it is called multigraph.

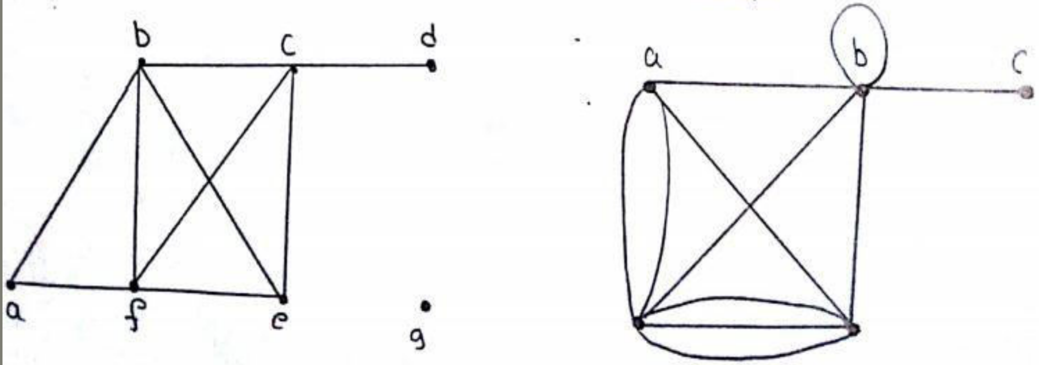


2. **Mixed Graph:** A graph with both directed and undirected edges is called a mixed graph.

GRAPH TERMINOLOGIES:

- First, we give some terminology that describes the vertices and edges of undirected graphs
- Two vertices u and v in an undirected graph G are called adjacent (or neighbors) in G if u and v are endpoints of an edge e of G. Such an edge e is called incident with the vertices u and v and e is said to connect u and v.
- The **degree** of a vertex in an undirected graph is the number of edges incident with it, except that a loop at a vertex contributes twice to the degree of that vertex. The degree of the vertex v is denoted by deg(v).

- Example: What are the degree of the vertices in the graphs G and H? ~~graph~~.



[G] [H]

Solution: In G.

deg(a) = 2
deg(b) = 4
deg(c) = 4
deg(d) = 1
deg(e) = 3
deg(f) = 4
deg(g) = 0

In H

deg(a) = 4
deg(b) = 6
deg(e) = 6
deg(d) = 5
deg(c) = 1

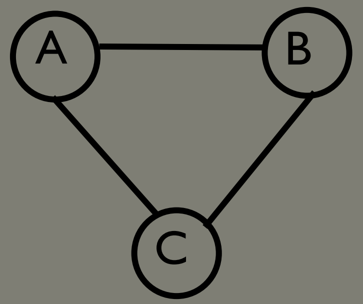
GRAPH TERMINOLOGIES:

- A vertex of degree zero is called **isolated**. It follows that an isolated vertex is not adjacent to any vertex.

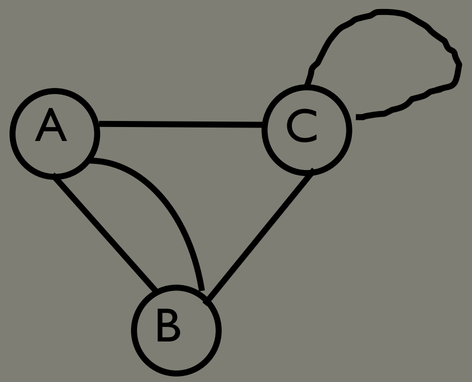
- A vertex is **pendant** if and only if it has degree one. Consequently, a pendant vertex is adjacent to exactly one other vertex.

Theorem 1: THE HANDSKING THEOREM:

Let G = (V, E) be an undirected graph with m edges. Then $2m = \sum_{v \in V} deg(v)$



2*3 = 2+2+2
6=6



2*5 = 3+3+4
10=10

Q. How many edges are there in a graph with 10 vertices each of degree 6?

Because the sum of the degrees of the vertices is $6 \cdot 10 = 60$, it follows that $2m = 60$ where m is the number of edges. Therefore, $m = 30$.

Theorem 1 shows that the sum of the degrees of the vertices of an undirected graph is even

GRAPH TERMINOLOGIES:

***Theorem 2:** An undirected graph has an even number of vertices of odd degree:*

Proof:

Take two sets of vertices, V_1, a set of vertices with even degree, and V_2, a set of vertices with odd degree in an undirected graph G =(V, E) with m edges. Then,

$$2e = \sum_{v \in V} \deg(v) = \sum_{v_1 \in V} \deg(v) + \sum_{v_2 \in V} \deg(v)$$

Because deg(v) is even for $v \in V_1$, the first term in the right-hand side of the last equality is even. Furthermore, the sum of the two terms on the right-hand side of the last equality is even, because this sum is 2m. Hence, the second term in the sum is also even. Because all the terms in this sum are odd, there must be an even number of such terms. Thus, there are an even number of vertices of odd degree

GRAPH TERMINOLOGIES:

- Terminology for graphs with directed edges reflects the fact that edges in directed graphs have directions.

- Let (u, v) be an edge representing edge of a directed graph G. u is called adjacent to v and v is called adjacent from u. The vertex u is called initial vertex and the vertex v is called terminal or end vertex. Loop has same initial and terminal vertex.

- In directed graph the in-degree of a vertex v, denoted by deg^- (v) , is the number of edges that have v as their terminal vertex(incoming edges). The out-degree of a vertex v, denoted by deg^+ (v), is the number of edges that have v as their initial vertex(outgoing edges). (Note that a loop at a vertex contributes 1 to both the in-degree and the out-degree of this vertex.)

GRAPH TERMINOLOGIES:

- Example: Find the in-degree and out-degree of each vertex in the graph G with directed edges shown in the following figure:

Solution:

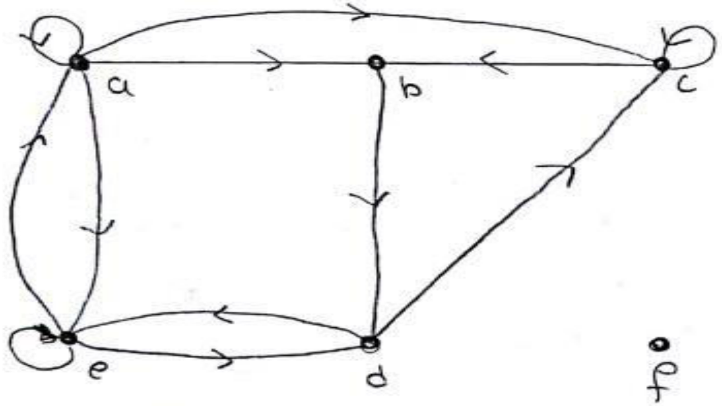


Fig: The directed graph G.

The in degrees in G are

$\deg^-(a) = 2$

$\deg^-(b) = 2$

$\deg^-(c) = 3$

$\deg^-(d) = 2$

$\deg^-(e) = 3$

$\deg^-(f) = 0$

The out degrees in G are

$\deg^+(a) = 4$

$\deg^+(b) = 1$

$\deg^+(c) = 2$

$\deg^+(d) = 2$

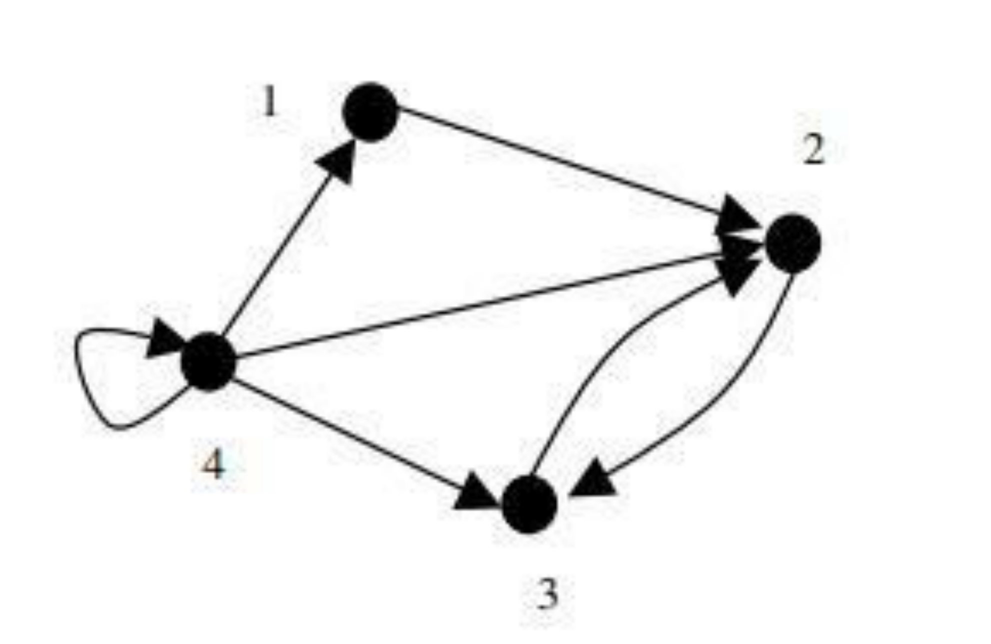
$\deg^+(e) = 3$

$\deg^+(f) = 0$

GRAPH TERMINOLOGIES:

Theorem 3: ***Let G(V, E) be a graph with directed edges. Then,***

$$\sum_{v \in V} deg^{-}(v) = \sum_{v \in V} deg^{+}(v) = |E|.$$

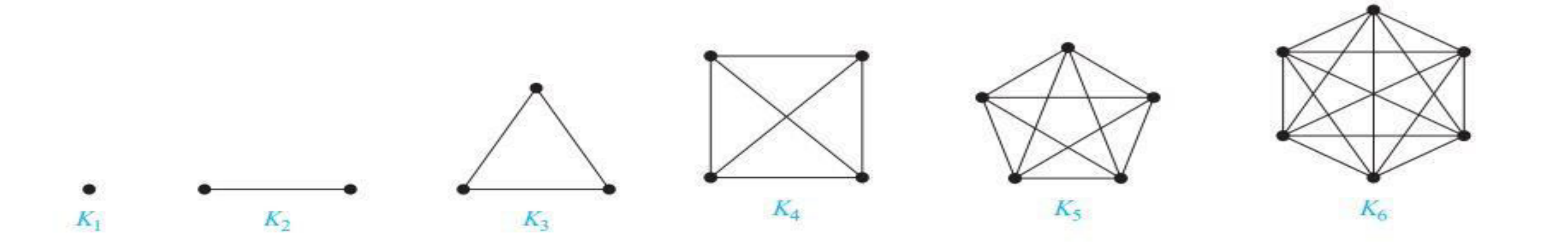


Sum of in-degree =1+1+3+2 =7
Sum of out-degree= 1+1+1+4 =7
Total edges = 7

In-degrees of a graph are $deg^{-}(1) = deg^{-}(4) = 1$; $deg^{-}(2) = 3$; $deg^{-}(3) = 2$ and the out-degrees of a graph are $deg^{+}(1) = deg^{+}(2) = deg^{+}(3) = 1$; $deg^{+}(4) = 4$.

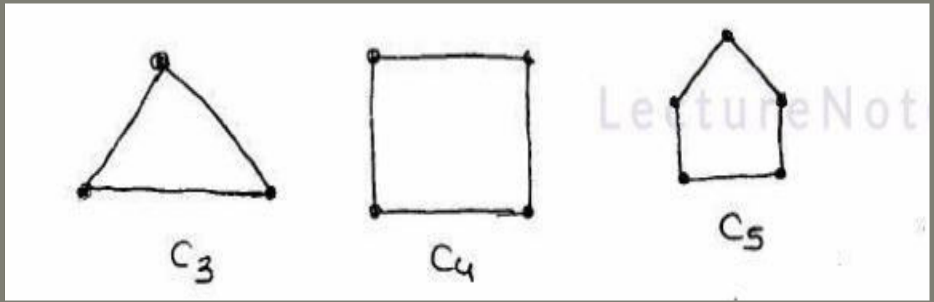
SOME SPECIAL TYPES OF GRAPHS:

1. **Complete Graph:** A complete graph on n vertices, denoted by K_n, is a simple graph that contains exactly one edge between each pair of distinct vertices. The graphs K_n, for n = 1, 2, 3, 4, 5, 6, are displayed in below Figures.



- **A complete graph is a regular graph but every regular graph is not complete graph**
- **A complete graph is a simple graph with max number of edges**
- **Number of edges in K_n= $\frac{n(n-1)}{2}$**
- **Degree of each vertex = (n-1)**

2. **Cycle Graph:** A graph G with n vertices (n>=3) and n edges , is called a cycle graph(C_n) , if all edges form cycle of length n.



1. **Prove that a complete graph with n vertices contains [n(n − 1)]/2 edges**

Solution:

This is easy to prove by induction.

(a) Base Case:

If n = 1,

K_1= [1(1 − 1)]/2 = 0(which is true)

(b) Induction Hypothesis:

We assume that it is true for some arbitrary value k(k>=2) i.e. [k(k-1)]/2

(c) Induction Step:

We have to prove it for (k+1) .ie. [(k+1)(k)]/2

When we add the $(k + 1)^{st}$ vertex, we need to connect it to the k original vertices, requiring k additional edges.

We will then have $\frac{k(k-1)}{2}$+ k

$= \frac{k(k-1)}{2} + k$

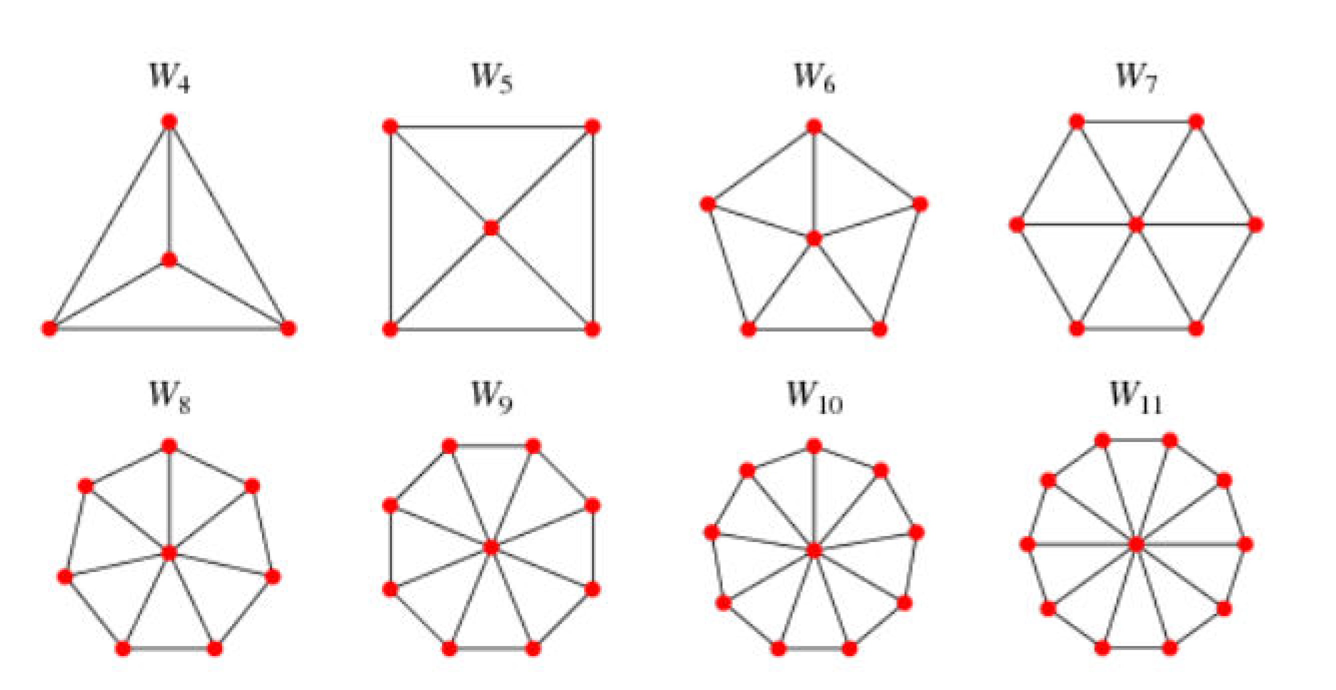
$= ^{-}\frac{k(k-1)+2k}{2}$

$= \frac{k^2-k+2k}{2} \quad = \frac{k^2+k}{2}$

$= \frac{k(k+1)}{2}$

SOME SPECIAL TYPES OF GRAPHS:

3. **Wheel Graph:** A wheel graph W_n of n vertices(n>=4) can be formed from a cycle graph C_{n-1} by adding a new vertex(hub) which is adjacent to all vertices of C_{n-1}. . The wheel graphs are displayed below.

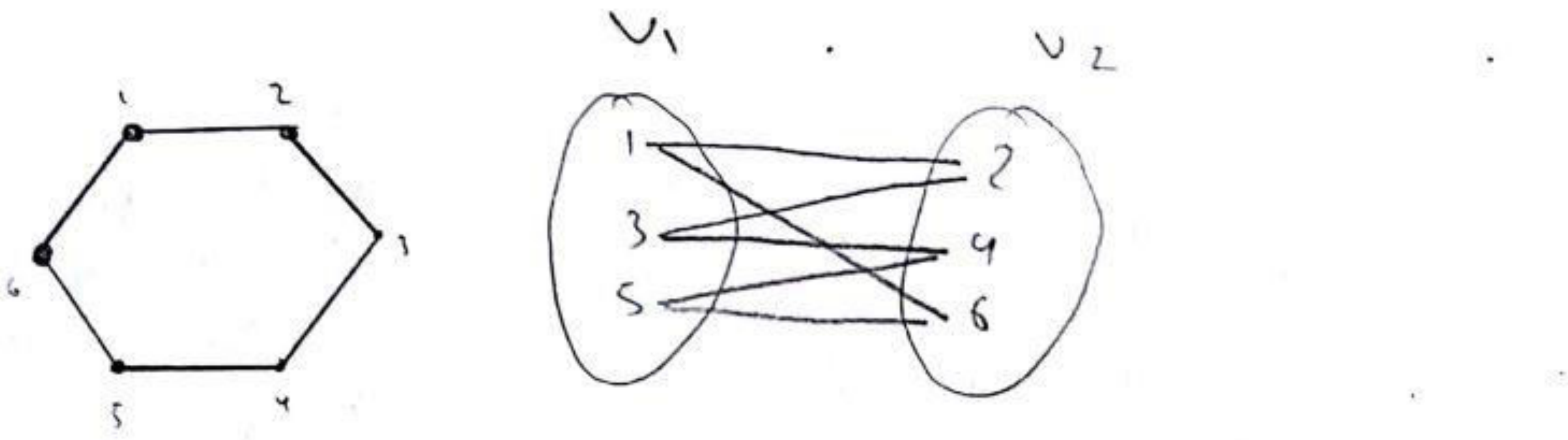


- Number of edges in W_n = 2(n-1)

SOME SPECIAL TYPES OF GRAPHS:

4. Bipartite Graphs: A simple graph G is called bipartite if its vertex set V can be partitioned into two disjoint sets V_1 and V_2 such that every edge in the graph connects a vertex in V_1 and a vertex in V_2 (so that no edge in G connects either two vertices in V_1 or two vertices in V_2). When this condition holds, we call the pair (V_1, V_2) a bipartition of the vertex set V of G.

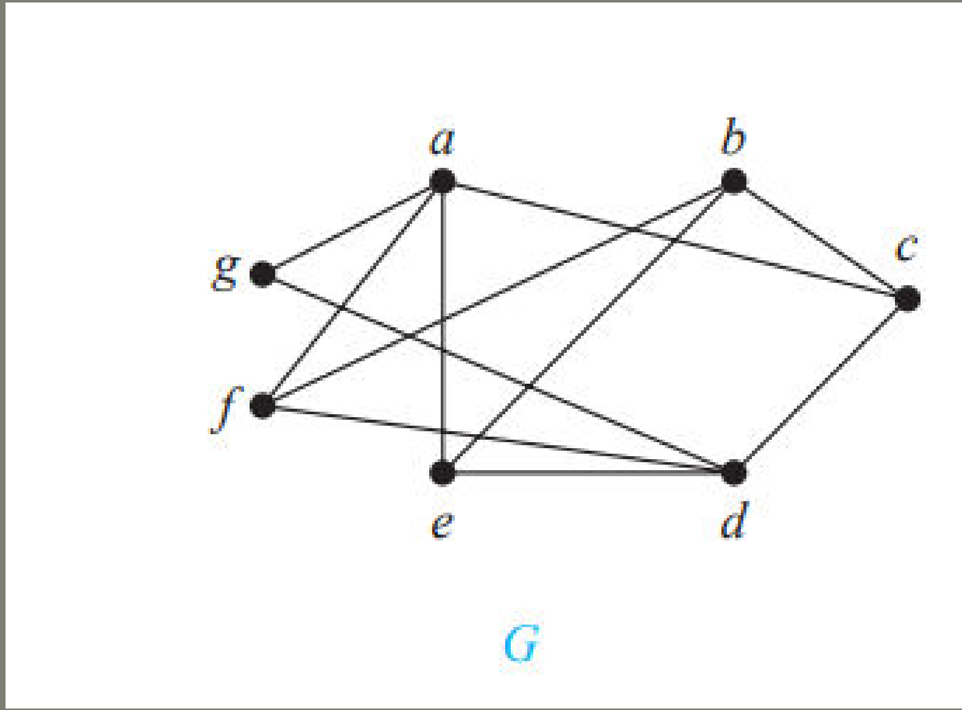
<u>Ex</u> C_6 is bipartite, because its vertex set can be partitioned into the two sets $V_1 = \{1, 3, 5\}$ and $V_2 = \{2, 4, 6\}$ and every edge of C_6 connects a vertex in V_1 and a vertex in V_2.

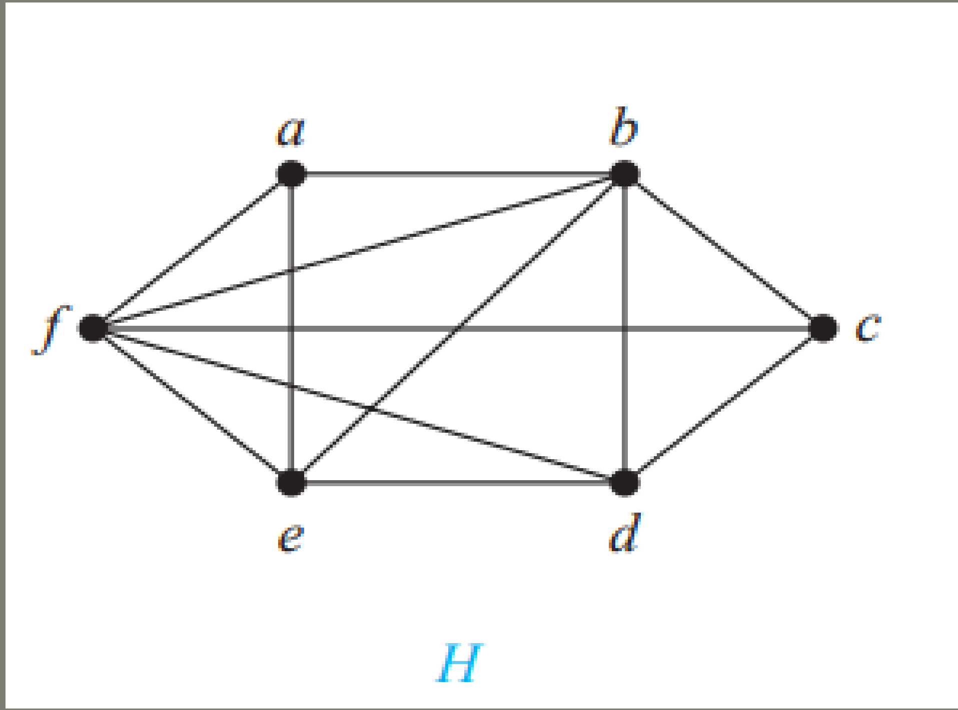


SOME SPECIAL TYPES OF GRAPHS:

Theorem 4 : *A simple graph is bipartite if and only if it is possible to assign one of two different colors to each vertex of the graph so that no two adjacent vertices are assigned the same color.*

Is G and H bipartite graph??





SOME SPECIAL TYPES OF GRAPHS:

5. Complete Bipartite Graphs: A complete bipartite $K_{m,n}$ graph is a special type of bipartite graph where every vertex of one set is connected to every vertex of another set.

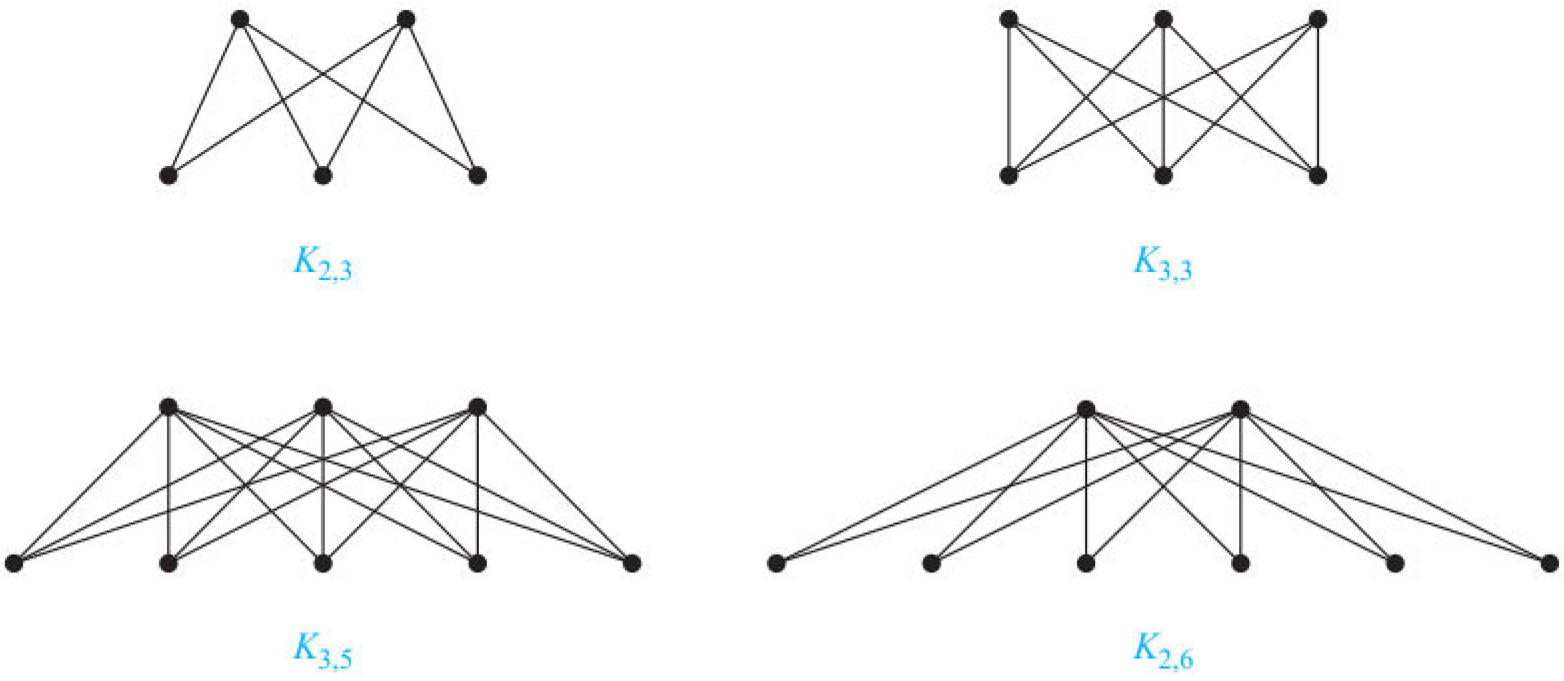
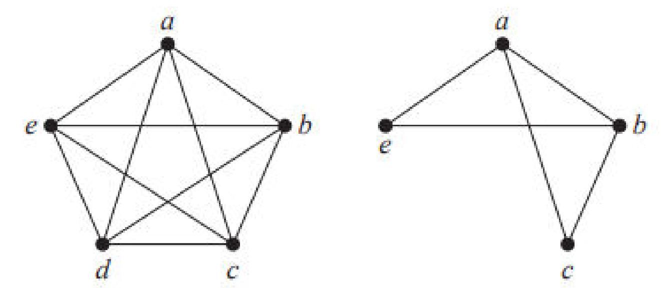


FIGURE 9 **Some Complete Bipartite Graphs.**

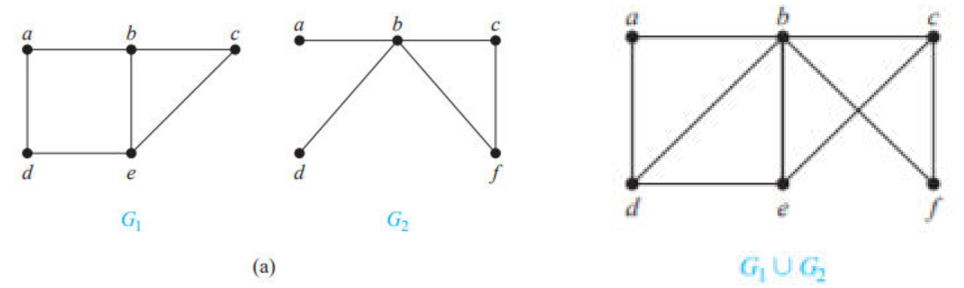
SOME SPECIAL TYPES OF GRAPHS:

5. **Subgraph Graphs:** A subgraph of a graph $G = (V, E)$ is a graph $H = (W, F)$, where $W \subseteq V$ and $F \subseteq E$. A subgraph H of G is a proper subgraph of G if $H \neq G$



A Subgraph of K_5.

GRAPH UNIONS: Two or more graphs can be combined in various ways. The new graph that contains all the vertices and edges of these graphs is called the union of the graphs. 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 $G_1 \cup G_2$.



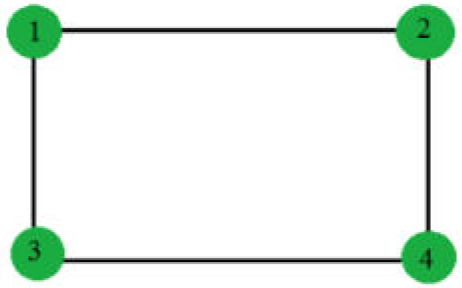
(a)

(a) The Simple Graphs G_1 and G_2;

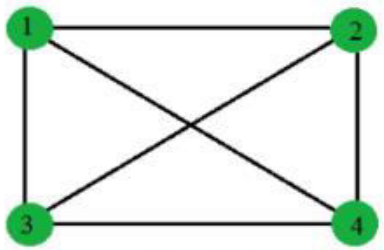
(b)

SOME SPECIAL TYPES OF GRAPHS:

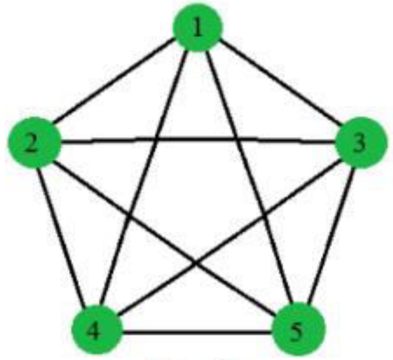
6. **Regular Graphs:** A graph is called regular graph if degree of each vertex is equal. A graph is called **K regular** if degree of each vertex in the graph is K.
   - A complete graph with N vertices is *(N-1)* regular.
   - Cycle(C_n) is always 2 Regular.



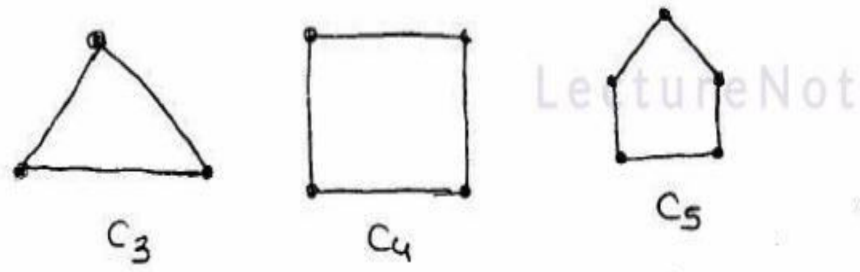
2 Regular



3 Regular



4 Regular



2 Regular

1. **Prove that a maximum number of edges possible in a simple graph with n vertices is [n(n − 1)]/2**

Solution:

By Handshakin Therorem , We have

$$2m = \sum_{v \in V} deg(v)$$ where m= number of edges with n vertices in the Graph G

$deg(v_1) + dwg(v_2) + deg(v_3) + \ldots\ldots\ldots deg(v_n) = 2m$-------------------------(i)

We know that the maxium degree of a vertices in a Simple grah can be(n-1),

We can write equation (i) as,

(n-1) + (n-1) + (n-1) + …………up to n vertices = 2m

n(n-1) = 2m

m= **[n(n − 1)]/2**

MATHEMATICAL FOUNDATION FOR COMPUTER SCIENCE

Prepared by: Er. Ankit Kharel

Nepal college of information technology

GRAPH THEORY

REPRESENTING GRAPHS:

1. **Adjacency List:**

   One way to represent a graph is to use adjacency lists, which specify the vertices that are adjacent to each vertex of the graph

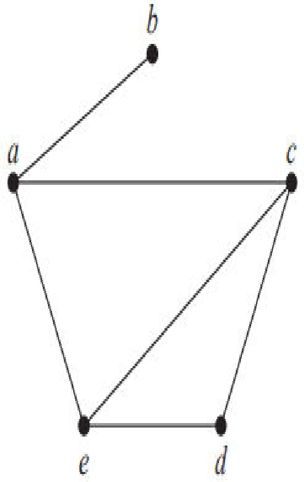


FIGURE 1 A Simple Graph.

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*

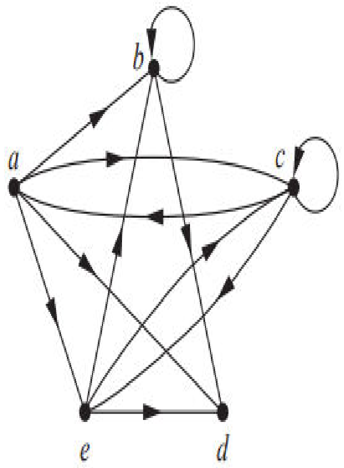


FIGURE 2 A Directed Graph.

TABLE 2 An Adjacency List for a Directed Graph.

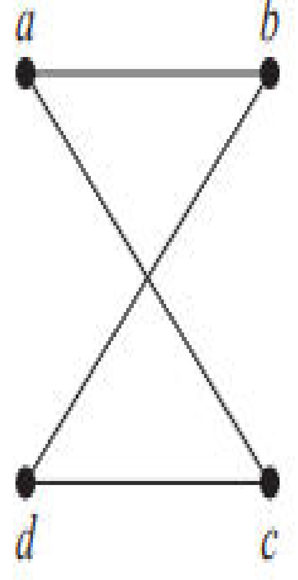
Initial Vertex	*Terminal Vertices*
a	*b, c, d, e*
b	*b, d*
c	*a, c, e*
d	
e	*b, c, d*

REPRESENTING GRAPHS:

2. Adjacency Matrix:

Suppose that G = (V , E) is a simple graph where |V| = n. Suppose that the vertices of G are listed arbitrarily as $v_1, v_2, \ldots, v_n$. The **adjacency matrix** A (or AG) of G, with respect to this listing of the vertices, is the n x n zero–one matrix with 1 as its (i, j)th entry when v_i and v_j are adjacent, and 0 as its (i, j)th entry when they are not adjacent. In other words, if its adjacency matrix is $A = [a_{ij}]$, then

$$a_{ij} = \begin{cases} 1 \text{ if } \{vi, vj\} \text{ is an edge of G,} \\ 0 \text{ otherwise.} \end{cases}$$



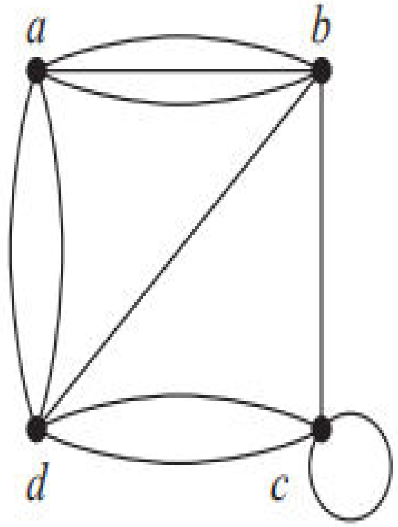
$$\begin{bmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix}$$

with respect to the ordering of vertices *a*, *b*, *c*, *d*.

REPRESENTING GRAPHS:

Adjacency matrices can also be used to represent undirected graphs with loops and with multiple edges. When multiple edges connecting the same pair of vertices vi and vj , or multiple loops at the same vertex, are present, the adjacency matrix is no longer a zero–one matrix, because the (i, j)th entry of this matrix equals the number of edges that are associated to {vi, vj }. All undirected graphs, including multigraphs and pseudo graphs, have symmetric adjacency matrices.

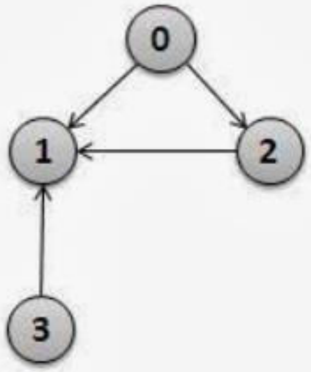
EXAMPLE 5 Use an adjacency matrix to represent the pseudograph shown in Figure



Solution: The adjacency matrix using the ordering of vertices a, b, c, d is

$$\begin{bmatrix} 0 & 3 & 0 & 2 \\ 3 & 0 & 1 & 1 \\ 0 & 1 & 1 & 2 \\ 2 & 1 & 2 & 0 \end{bmatrix}.$$

REPRESENTING GRAPHS:



	0	1	2	3
0	0	1	1	0
1	0	0	0	0
2	0	1	0	0
3	0	1	0	0

Adjacency Matrix Representation of Directed Graph

The matrix for a directed graph G = (V , E) has a 1 in its (i, j)th position if there is an edge from vi to vj , where v1, v2,..., vn is an arbitrary listing of the vertices of the directed graph. In other words, if A = [aij] is the adjacency matrix for the directed graph with respect to this listing of the vertices, then

$$a_{ij} = \begin{cases} 1 & \text{if } (v_i, v_j) \text{ is an edge of } G, \\ 0 & \text{otherwise.} \end{cases}$$

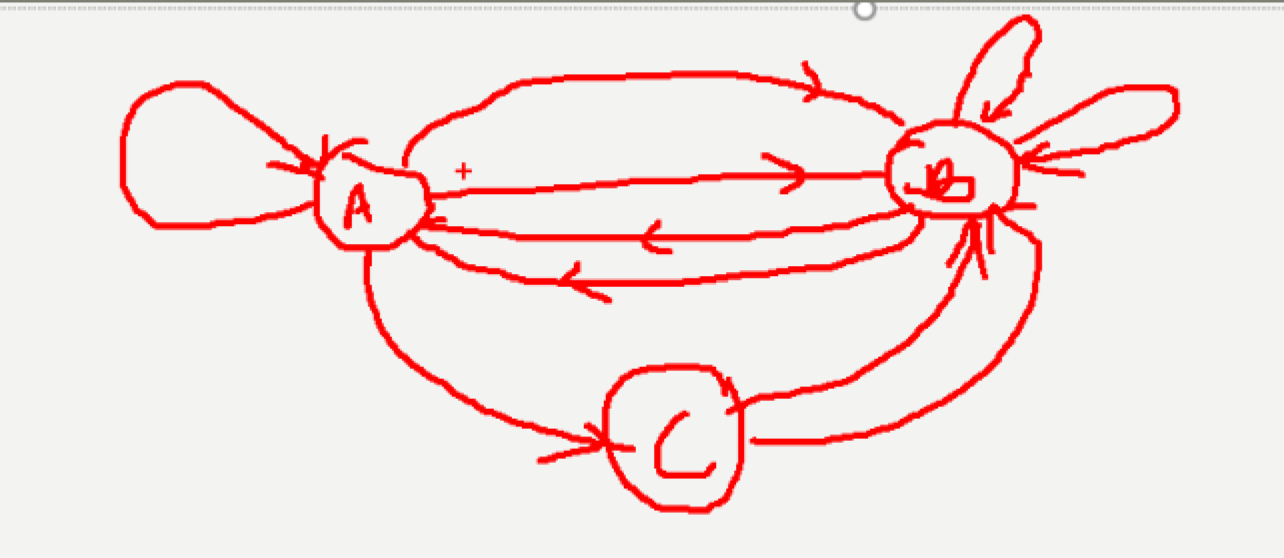
The adjacency matrix for a directed graph does not have to be symmetric, because there may not be an edge from vj to vi when there is an edge from vi to vj .

REPRESENTING GRAPHS:

Draw the graph represented by the given adjacency matrix.

$$\begin{bmatrix} 1 & 2 & 1 \\ 2 & 0 & 0 \\ 0 & 2 & 2 \end{bmatrix}$$

Since the matrix is not symmetric, we need directed edges.



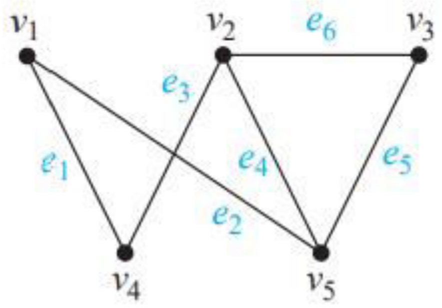
REPRESENTING GRAPHS:

3. Incidence Matrices:

Another common way to represent graphs is to use **incidence matrices**. Let $G = (V, E)$ be an undirected graph. Suppose that $v_1, v_2, \ldots, v_n$ are the vertices and $e_1, e_2, \ldots, e_m$ are the edges of G. Then the incidence matrix with respect to this ordering of V and E is the $n \times m$ matrix $\mathbf{M} = [m_{ij}]$, where

$$m_{ij} = \begin{cases} 1 & \text{when edge } e_j \text{ is incident with } v_i, \\ 0 & \text{otherwise.} \end{cases}$$

Represent the graph shown in Figure 6 with an incidence matrix.

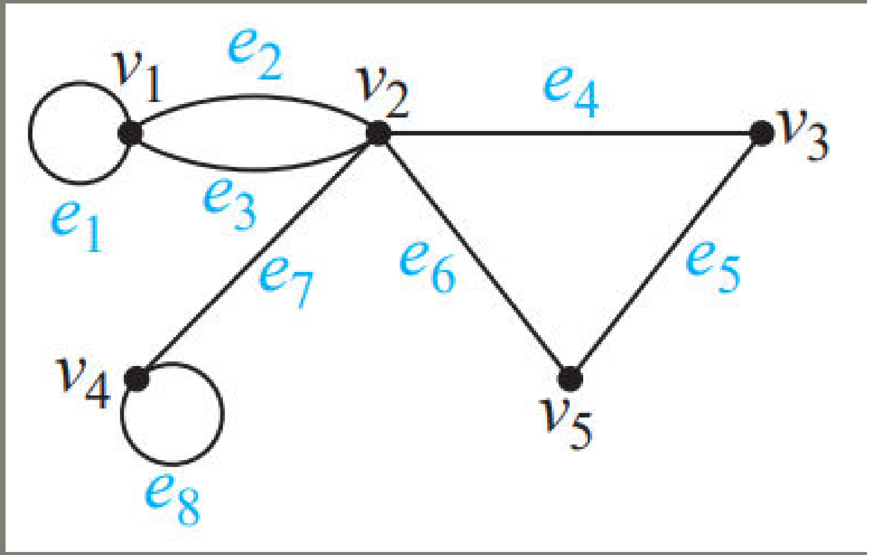


Solution: The incidence matrix is

$$\begin{array}{c} \\ v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \end{array} \begin{array}{c} \begin{array}{cccccc} e_1 & e_2 & e_3 & e_4 & e_5 & e_6 \end{array} \\ \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 & 0 \end{bmatrix} \end{array}.$$

REPRESENTING GRAPHS:

3. **Incidence Matrices:**



Solution: The incidence matrix for this graph is

$$
\begin{array}{c}
\\ v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5
\end{array}
\begin{array}{c}
\begin{array}{cccccccc} e_1 & e_2 & e_3 & e_4 & e_5 & e_6 & e_7 & e_8 \end{array} \\
\begin{bmatrix}
1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 1 & 1 & 0 & 1 & 1 & 0 \\
0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\
0 & 0 & 0 & 0 & 1 & 1 & 0 & 0
\end{bmatrix}
\end{array}.
$$

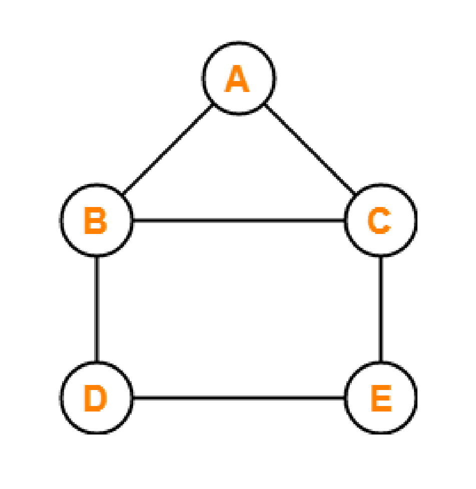
CONNECTIVITY:

1. **WALK:** A walk is defined as a finite alternating sequence of vertices and edges, beginning and ending with vertices such that each edge is incident with the vertices preceding and following it.

   Open Walk: a walk is called as an Open walk if Length of the walk is greater than zero and the vertices at which the walk starts and ends are different.

   Closed Walk: a walk is called as an Closed walk if Length of the walk is greater than zero and the vertices at which the walk starts and ends are same.

- If length of walk i= 0, then it is called as a Trivial Walk
- Both vertices and edges can repeat in a walk whether it is an open walk or a closed walk.



In this graph, few examples of walk are-

- a , b , c , e , d (Length = 4)
- d , b , a , c , e , d , e , c (Length = 7)
- e , c , b , a , c , e , d (Length = 6)

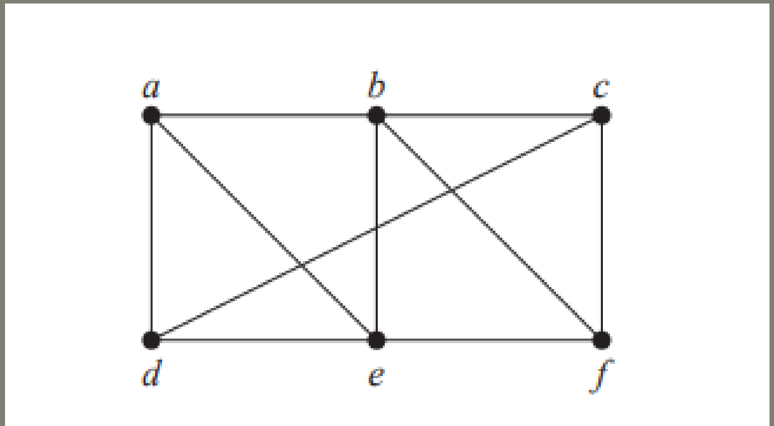
CONNECTIVITY:

2. **TRIAL:** A trail is defined as an open walk in which Vertices may repeat but edges are not allowed to repeat.

3. **CIRCUIT:** A circuit is defined as a closed walk in which Vertices may repeat but edges are not allowed to repeat.(Closed trial)

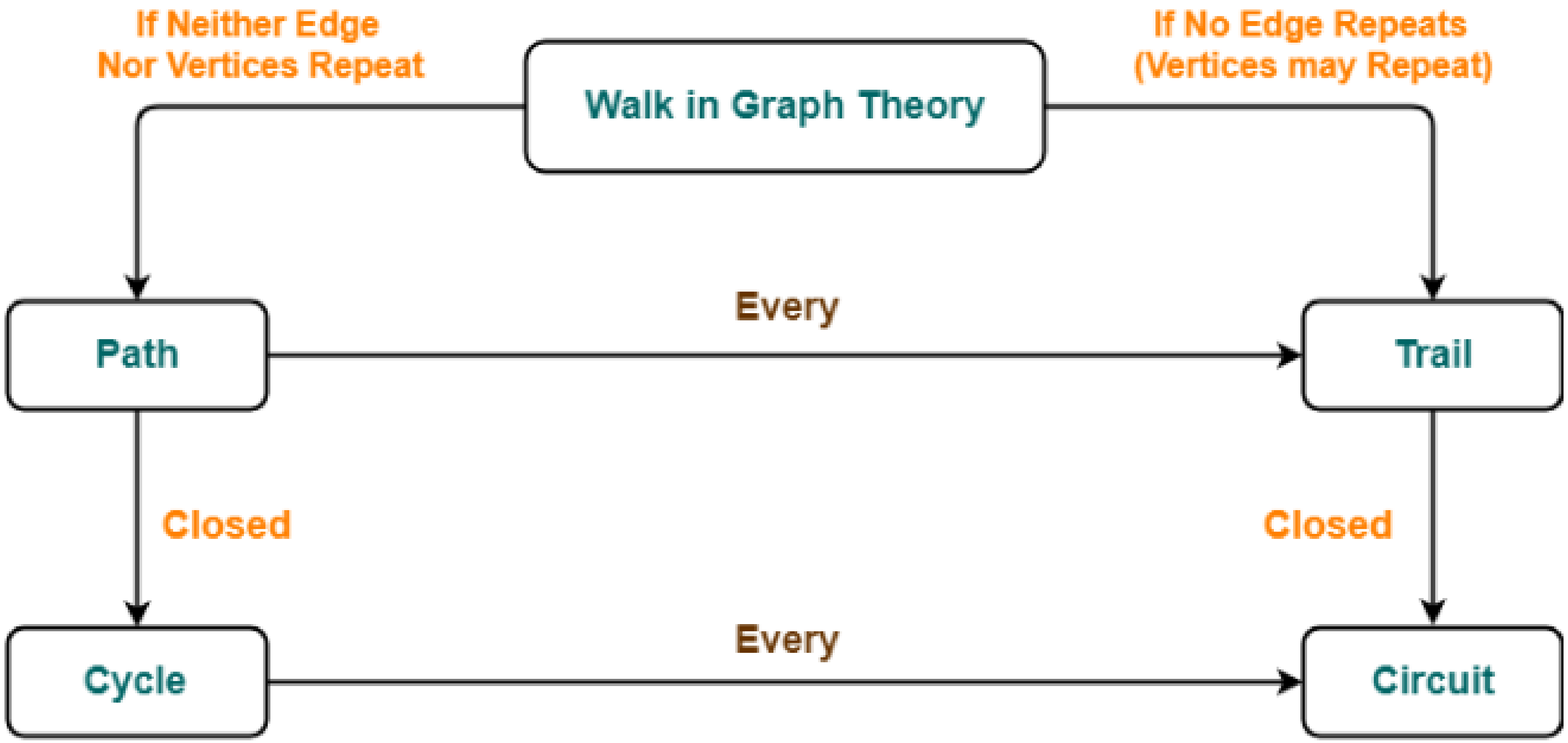
4. **PATH:** A path is defined as an open walk in which neither vertices are allowed to repeat nor edges are allowed to repeat.

5. **CYCLE:** A cycle is defined as a closed walk in which neither vertices (except possibly the starting and ending vertices) are allowed to repeat nor edges are allowed to repeat.(Closed Path)



Important Chart-

The following chart summarizes the above definitions and is helpful in remembering them-



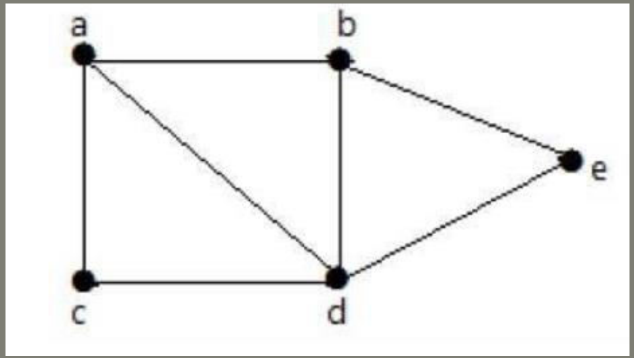
Important Chart to Remember

CONNECTEDNESS IN UNDIRECTED GRAPHS:

- A graph is said to be **connected if there is a path between every pair of vertex**. From every vertex to any other vertex, there should be some path to traverse. That is called the connectivity of a graph

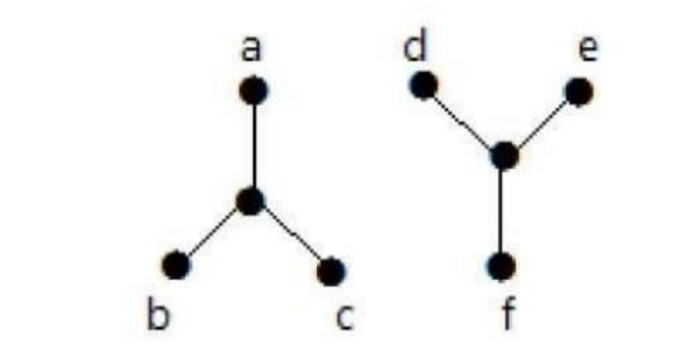
Example 1

In the following graph, it is possible to travel from one vertex to any other vertex. For example, one can traverse from vertex 'a' to vertex 'e' using the path 'a-b-e'.



Example 2

In the following example, traversing from vertex 'a' to vertex 'f' is not possible because there is no path between them directly or indirectly. Hence it is a disconnected graph.



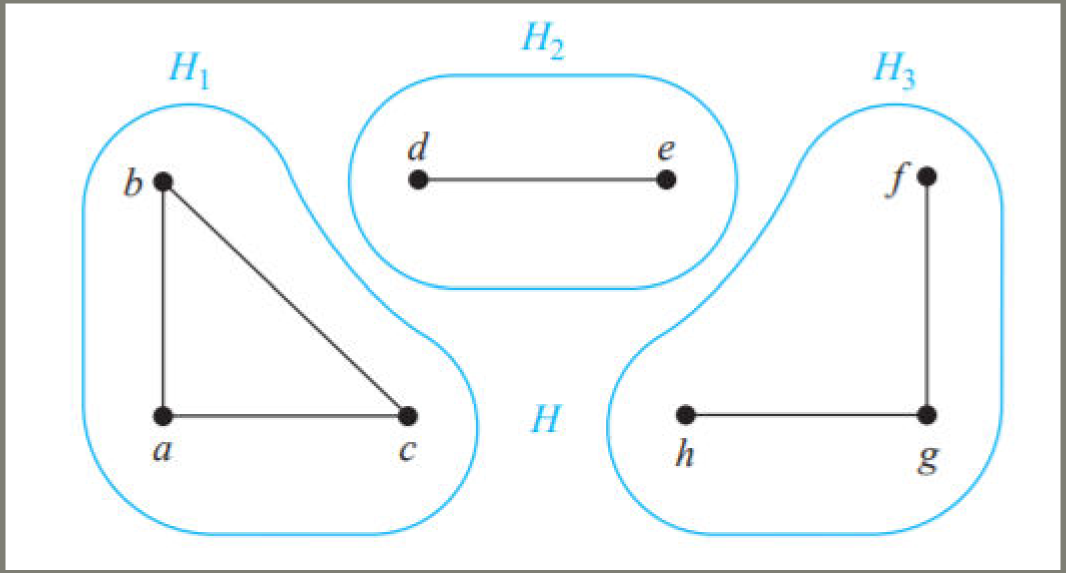
CONNECTEDNESS IN UNDIRECTED GRAPHS:

- **CONNECTED COMPONENTS:** A connected component of a graph G is a maximal connected subgraph of G.

- What are the connected components of the graph H shown in below figure?

  Solution:

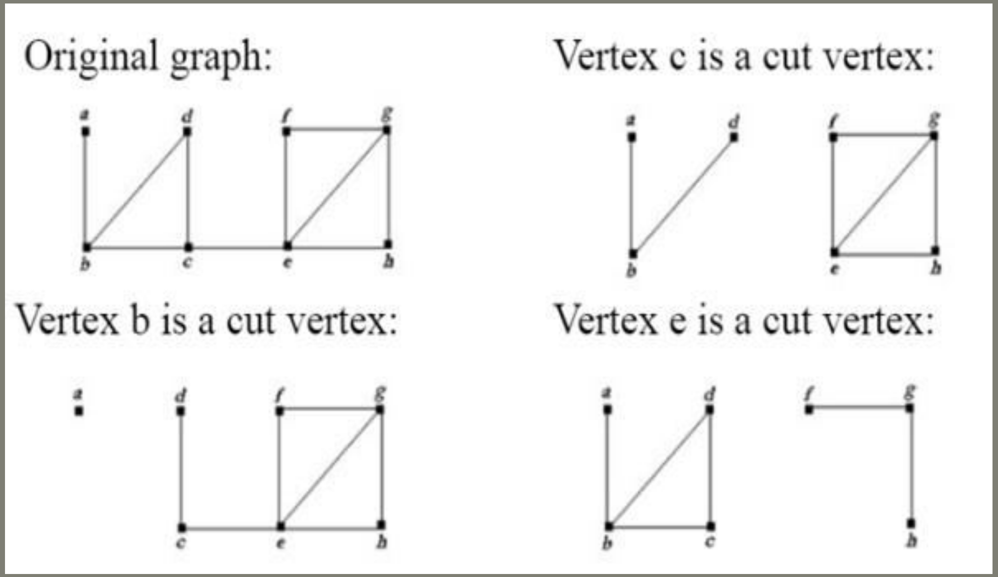
  The graph H is the union of three disjoint connected subgraphs H1, H2, and H3, shown in Figure. These three subgraphs are the connected components of H.



CONNECTEDNESS IN UNDIRECTED GRAPHS:

1. **CUT VERTICES:** Sometimes the removal from a graph of a vertex and all incident edges produces a subgraph with more connected components disconnects the Graph. Such vertices are called cut vertices(or articulation points).

2. **CUT EDGES:** Analogously, an edge whose removal produces a graph with more connected components than in the original graph is called a cut edge or bridge.



- Find the cut vertices and cut edges in the graph G shown in above Figure.

  <u>Solution:</u>

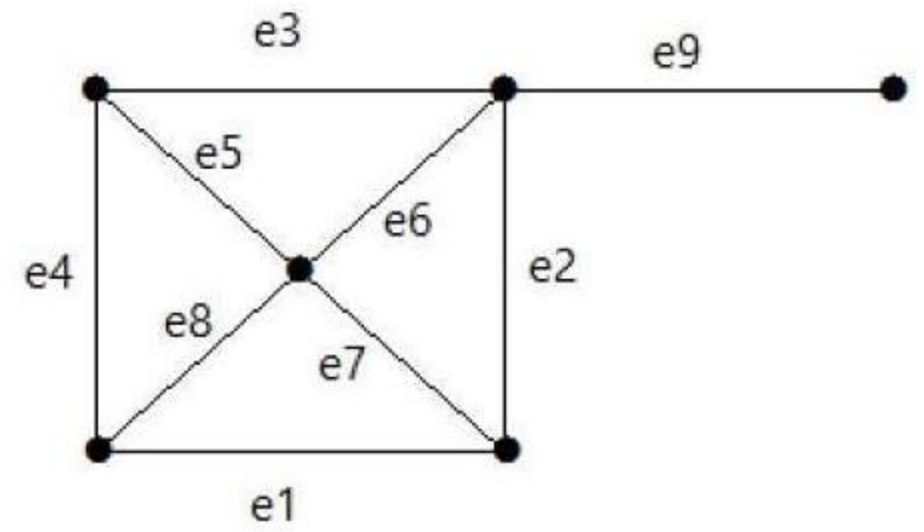
  The cut vertices of G are **b**, **c**, and **e**. The removal of one of these vertices (and its adjacent edges) disconnects the graph.

  The cut edges are {a, b} and {c, e}. Removing either one of these edges disconnects G.

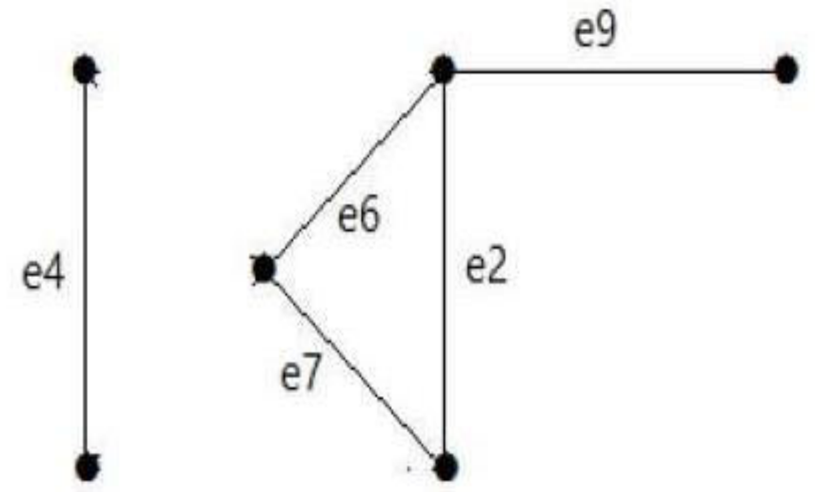
CONNECTEDNESS IN UNDIRECTED GRAPHS:

3. **Cut Set of a Graph:** Let 'G'= (V, E) be a connected graph. A subset E' of E is called a cut set of G if deletion of all the edges of E' from G makes G disconnect.

Take a look at the following graph. Its cut set is E1 = {e1, e3, e5, e8}.



After removing the cut set E1 from the graph, it would appear as follows –



Similarly there are other cut sets that can disconnect the graph-

E3 = {e9} – Smallest cut set of the graph.

E4 = {e3, e4, e5}

E5 = {e1, e7, e2}

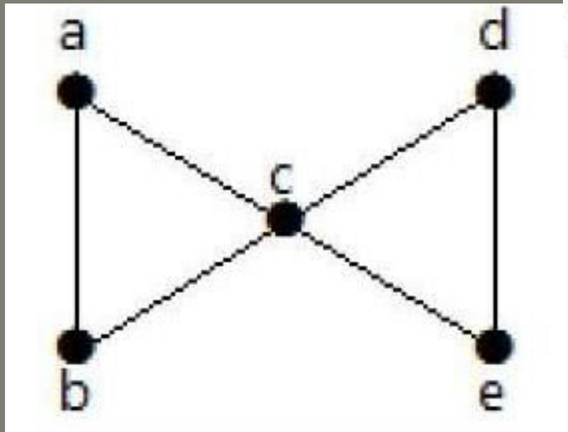
CONNECTEDNESS IN UNDIRECTED GRAPHS:

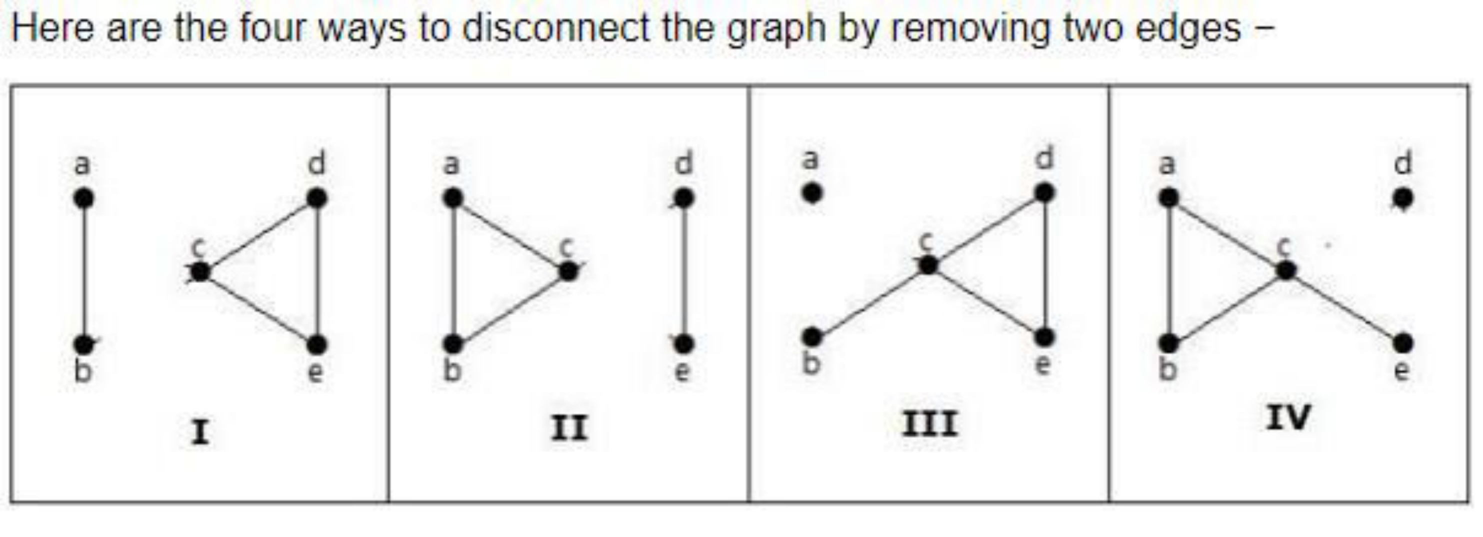
- Not all graphs have cut vertices. For example, the complete graph K_n, where $n \geq 3$, has no cut vertices. When you remove a vertex from K_n and all edges incident to it, the resulting subgraph is the complete graph K_{n-1}, a connected graph. Connected graphs without cut vertices are called **non separable graphs.**

- **Edge Connectivity:** Let 'G' be a connected graph. The minimum number of edges whose removal makes 'G' disconnected is called edge connectivity of G.

  Notation − λ(G)

  Take a look at the following graph. By removing two minimum edges, the connected graph becomes disconnected. Hence, its edge connectivity (λ(G)) is 2. Therefore the above graph is a **2-edge-connected graph**.

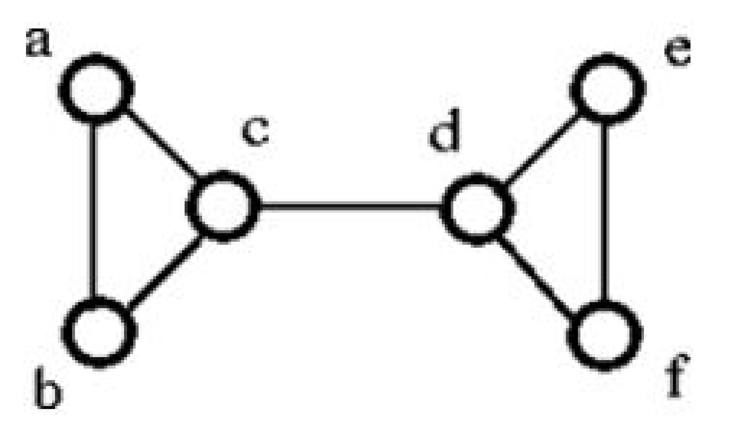




CONNECTEDNESS IN UNDIRECTED GRAPHS:

- **Vertex Connectivity:** The connectivity (or vertex connectivity) of a connected graph G is the minimum number of vertices whose removal makes G disconnects or reduces to a trivial graph. To remove a vertex we must also remove the edges incident to it.

  Notation − K(G)



The above graph G can be disconnected by removal of the single vertex either 'c' or 'd'. Hence, its vertex connectivity, K(G) is 1. Therefore, it is a 1-connected graph.

AN INEQUALITY FOR VERTEX CONNECTIVITY AND EDGE CONNECTIVITY

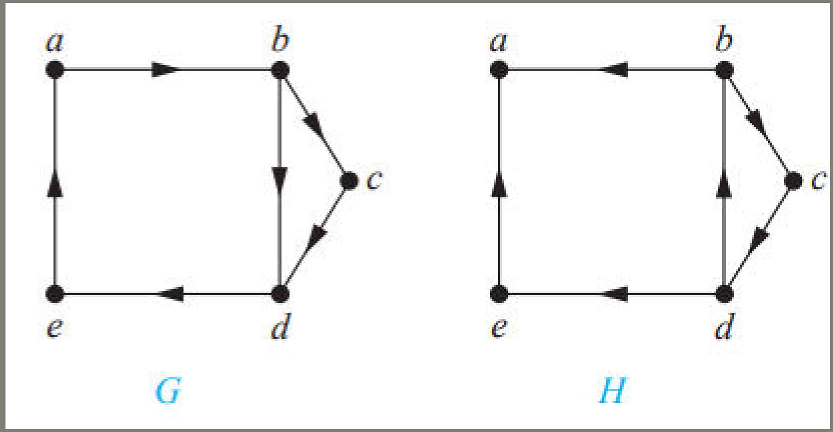
$$\kappa(G) \leq \lambda(G) \leq \min_{v \in V} \deg(v).$$

CONNECTEDNESS IN DIRECTED GRAPHS:

- There are two notions of connectedness in directed graphs, depending on whether the directions of the edges are considered.

- A directed graph is **strongly connected** if there is a path from a to b and from b to a whenever a and b are vertices in the graph.

- A directed graph is **weakly connected** if there is a path between every two vertices in the underlying undirected graph

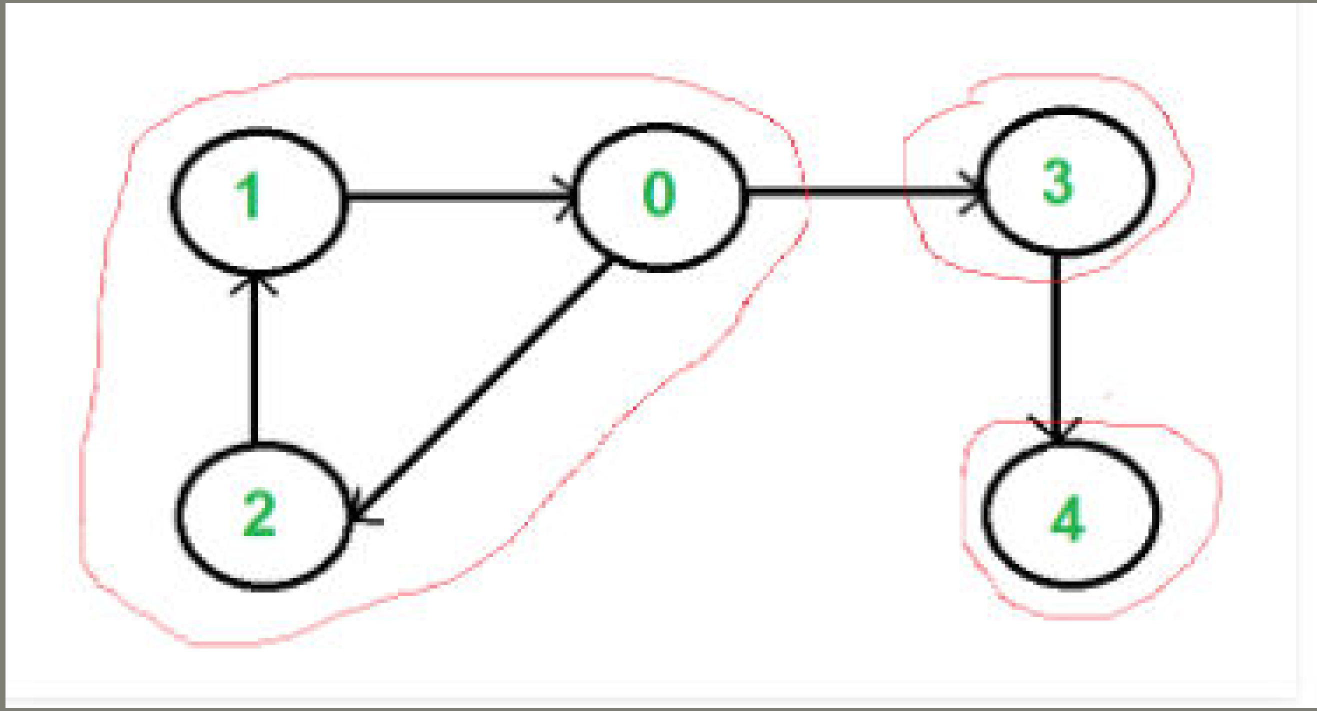


- Graph G is strongly connected because there is a path between any two vertices in this directed graph.

- The graph H is not strongly connected. There is no directed path from a to b in this graph. However, H is weakly connected, because there is a path between any two vertices in the underlying undirected graph of H

CONNECTEDNESS IN DIRECTED GRAPHS:

- A **strongly connected component** (**SCC**) of a directed graph is a maximal strongly connected subgraph. For example, there are 3 SCCs in the following graph.



MATHEMATICAL FOUNDATION FOR COMPUTER SCIENCE

Prepared by: Er. Ankit Kharel

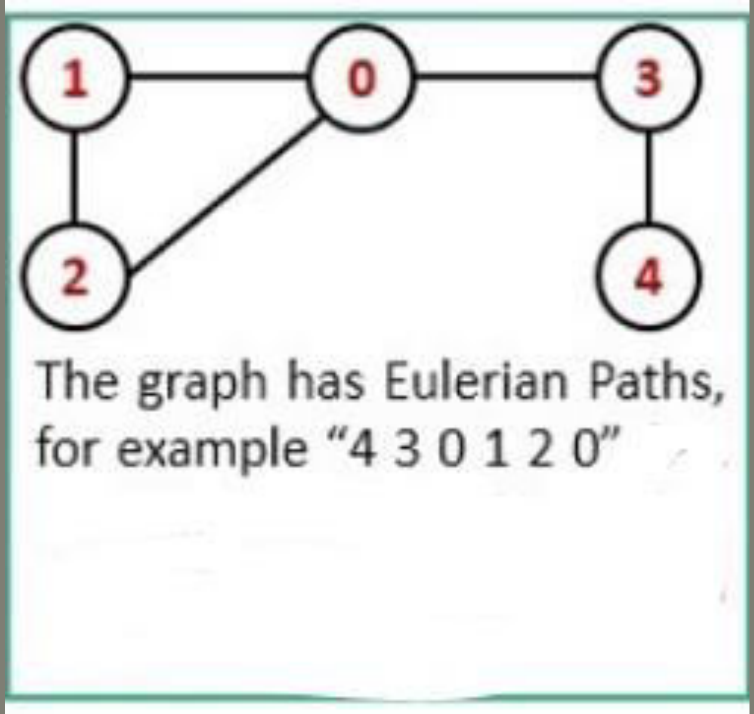
Nepal college of information technology

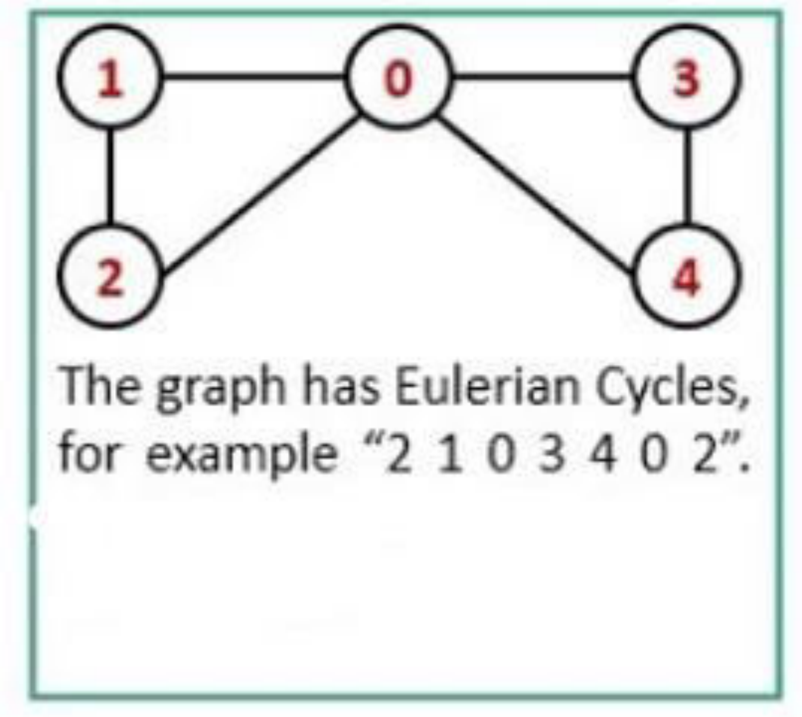
GRAPH THEORY

EULER GRAPH:

1. The **Euler path** is a trial, by which we can visit every edge of a Graph exactly once. We can use the same vertices for multiple times. The Euler Circuit is a special type of Euler path. When the starting vertex of the Euler path is also connected with the ending vertex of that path, then it is called the **Euler Circuit**.

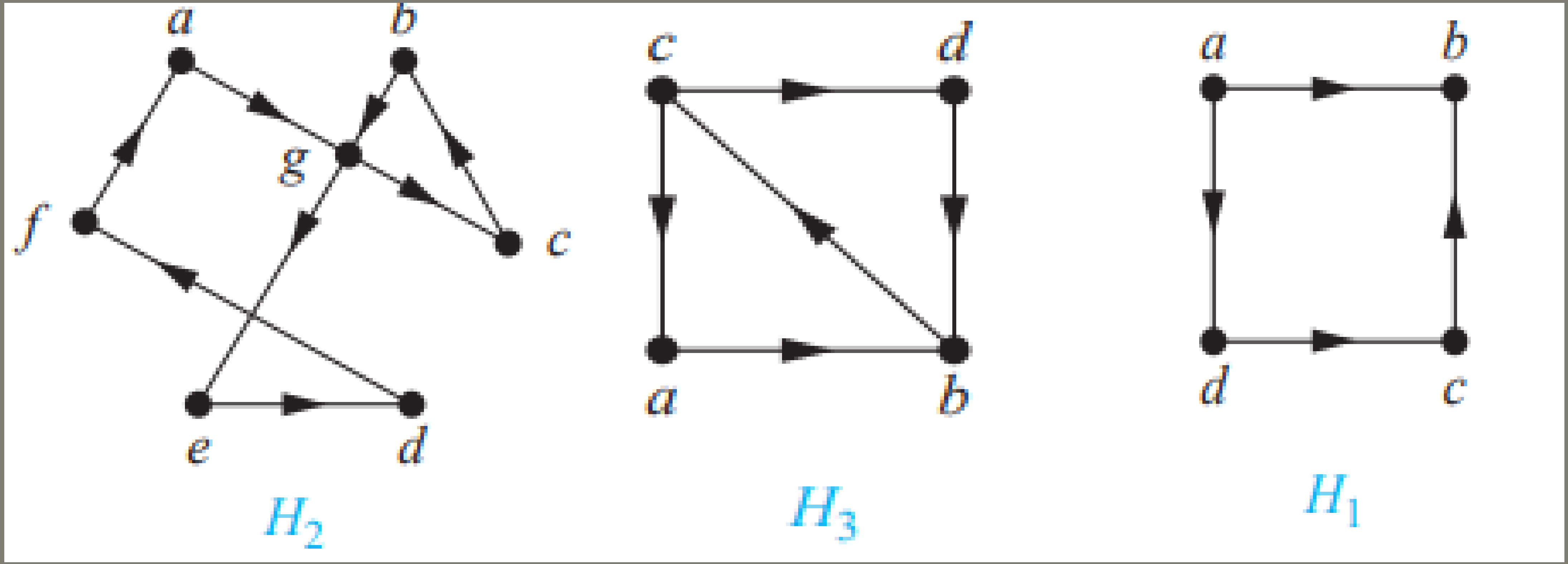
2. A graph containing Euler Path is called **Euler Graph**.





EULER GRAPH:

1. The graph H2 has an Euler circuit, for example, a, g, c, b, g, e, d, f, a. Neither H1 nor H3 has an Euler circuit (as the reader should verify). H3 has an Euler path, namely, c, a, b, c, d, b, but H1 does not (as the reader should verify).

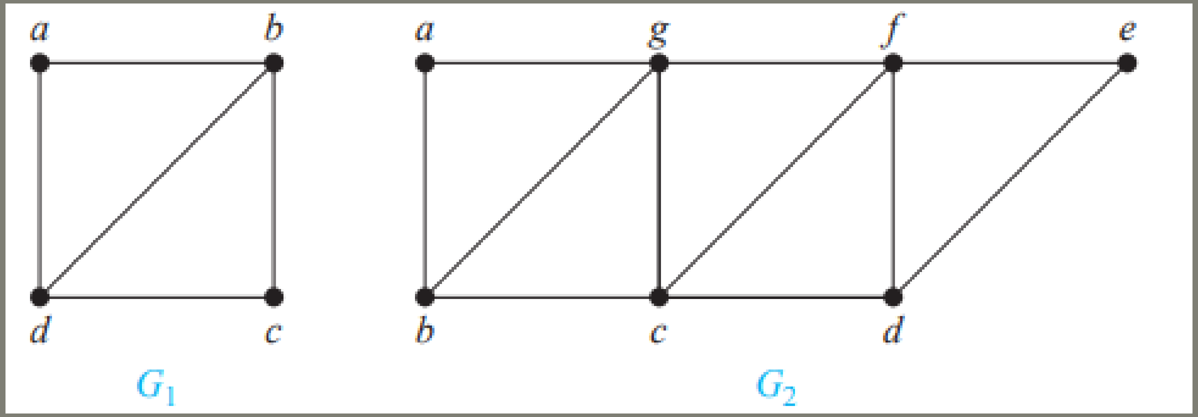


EULER GRAPH:

1. NECESSARY AND SUFFICIENT CONDITIONS FOR EULER CIRCUITS AND PATHS:

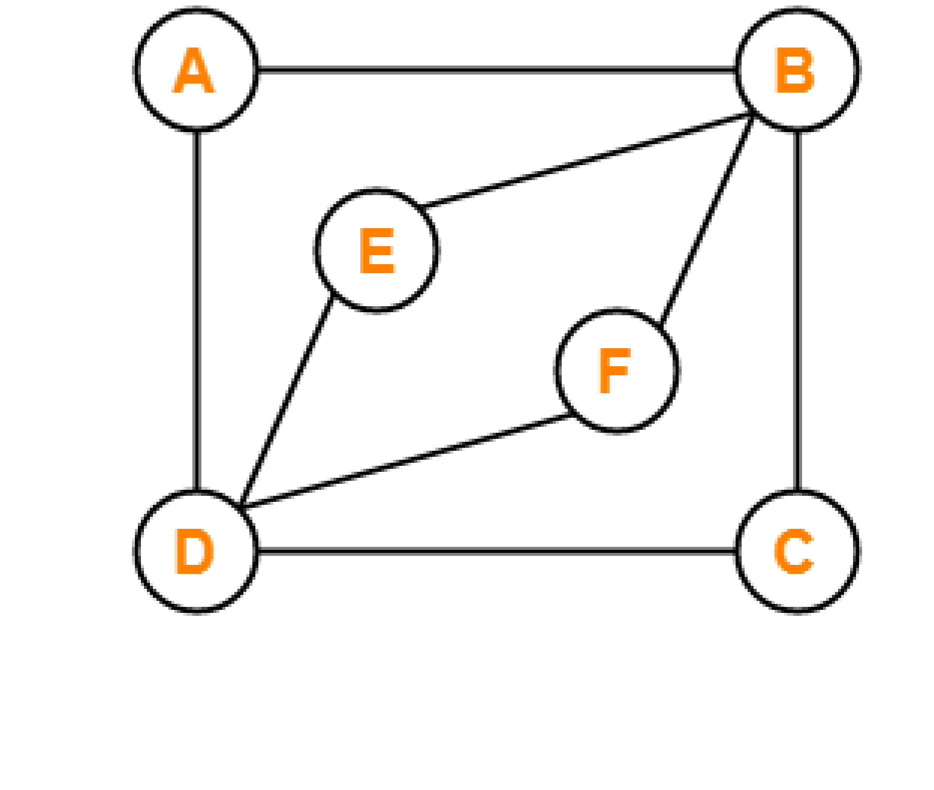
- *A connected multigraph with at least two vertices has an Euler circuit if and only if each of its vertices has even degree.*

- *A connected multigraph has an Euler path but not an Euler circuit if and only if it has exactly two vertices of odd degree and they are starting and ending point.*



G1 contains exactly two vertices of odd degree, namely, b and d. Hence, it has an Euler path that must have b and d as its endpoints. One such Euler path is **d, a, b, c, d, b.** Similarly, G2 has exactly two vertices of odd degree, namely, b and d. So it has an Euler path that must have b and d as endpoints. One such Euler path is **b, a, g, f, e, d, c, g, b, c, f, d**

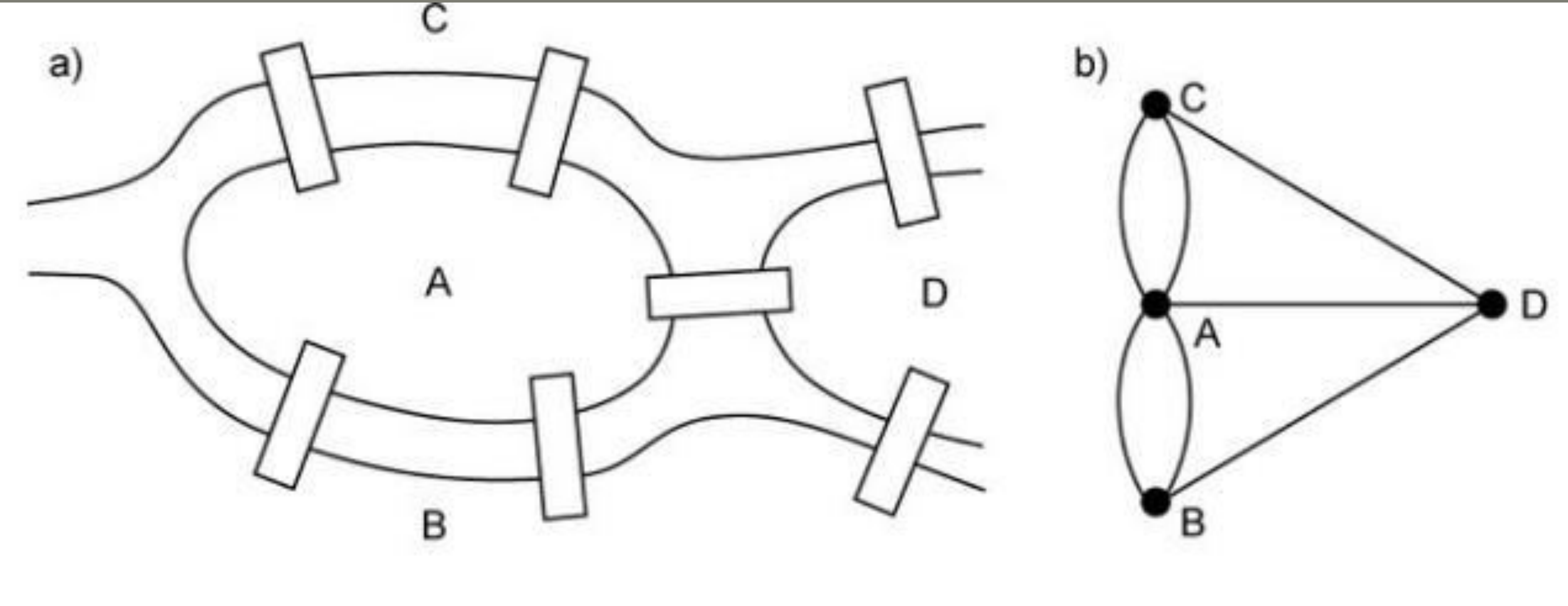
EULER GRAPH:



Since all vertices of above graph has EVEN degree. There is Euler Circuit.

A-B-C-D-F-B-E-D-A

7 BRIDGE OF KÖNIGSBERG:

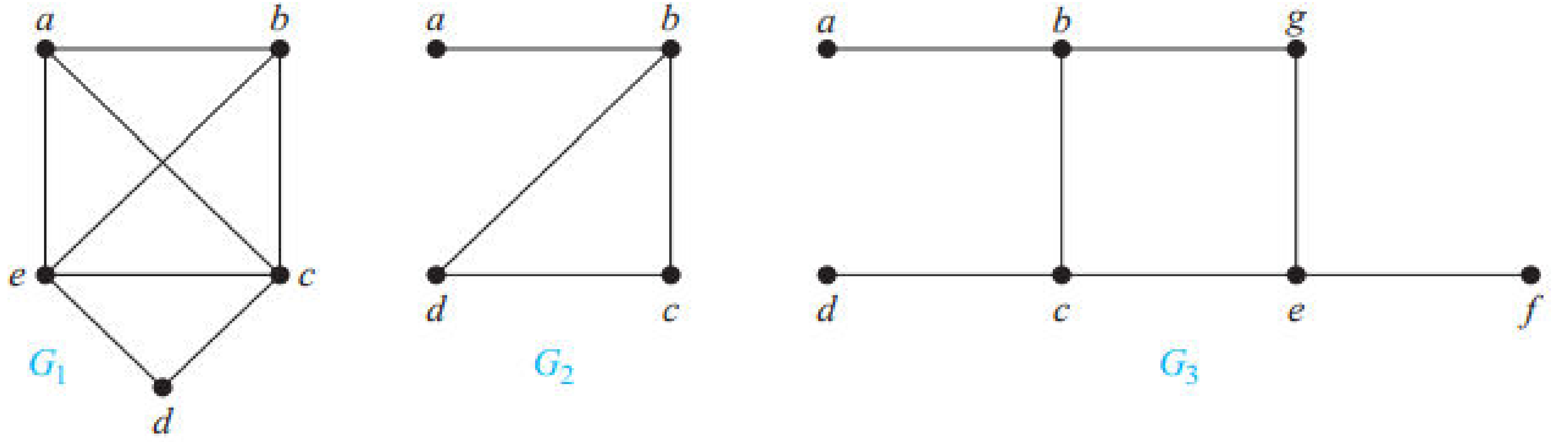


HAMILTON GRAPHS:

- A path in a graph G that passes through every vertex exactly once is called a **Hamilton path**. The Hamilton Circuit is a special type of Hamilton path. When the starting vertex of the Hamilton path is also connected with the ending vertex of that path, then it is called the **Hamilton Circuit**.

Which of the simple graphs in Figure 10 have a Hamilton circuit or, if not, a Hamilton path?

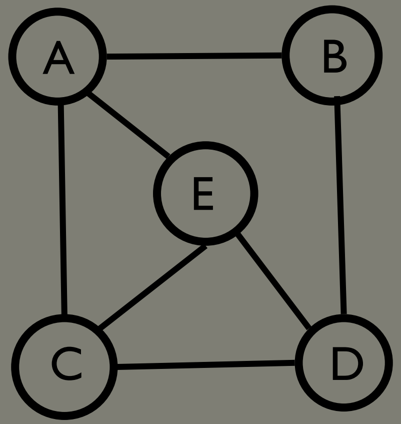
Solution: G_1 has a Hamilton circuit: a, b, c, d, e, a. There is no Hamilton circuit in G_2 (this can be seen by noting that any circuit containing every vertex must contain the edge $\{a, b\}$ twice), but G_2 does have a Hamilton path, namely, a, b, c, d. G_3 has neither a Hamilton circuit nor a Hamilton path, because any path containing all vertices must contain one of the edges $\{a, b\}$, $\{e, f\}$, and $\{c, d\}$ more than once. ◀



HAMILTON GRAPHS:

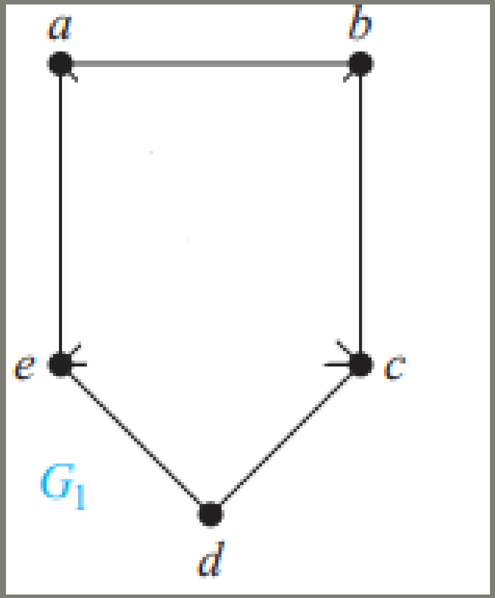
:SUFFICIENT CONDITON FOR HAMILTON GRAPH:

- **ORE'S THEOREM :** If G is a simple graph with n vertices with n ≥ 3 such that the sum of degrees of every pair of non adjacent vertices is greater or equal to n, then G has Hamilton circuit.



total vertices(n) = 5

- AD = 3+2 = 5
- BC = 2+3 = 5
- BE = 2+3 = 5
- Hence it is Hamiltonian.

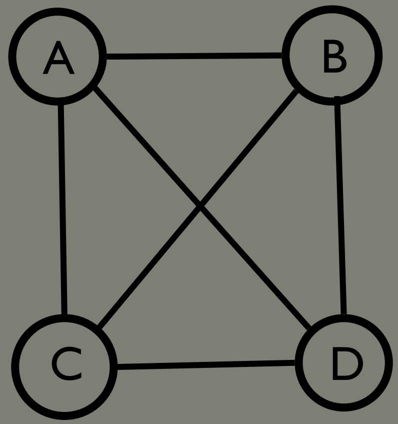


ORE's Condition does not apply to this graph but this graph is still Hamiltonian.

HAMILTON GRAPHS:

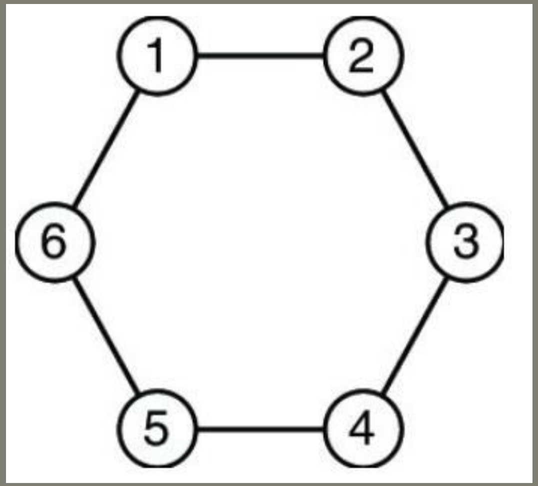
:SUFFICIENT CONDITON FOR HAMILTON GRAPH:

- **DIRAC'S THEOREM :** If G is a simple graph with n vertices with n ≥ 3 such that the degree of every vertex in G is at least n/2, then G has a Hamilton circuit.



total vertices(n) = 4

- Degree of A = 3
- Degree of B = 3
- Degree of C = 3
- Degree of D = 3
- Hence it is Hamiltonian.



DIRAC's Condition does not apply to this graph but this graph is still Hamiltonian.

MATHEMATICAL FOUNDATION FOR COMPUTER SCIENCE

Prepared by: Er. Ankit Kharel

Nepal college of information technology

GRAPH THEORY

PLANAR GRAPHS:

1. A graph is called **planar** if it can be drawn in the plane without any edges crossing (where a crossing of edges is the intersection of the lines or arcs representing them at a point other than their common endpoint). Such a drawing is called a planar representation of the graph.

2. A graph may be planar even if it is usually drawn with crossings, because it may be possible to draw it in a different way without crossings.

3. A planar representation of a graph splits the plane into regions, including an unbounded region.

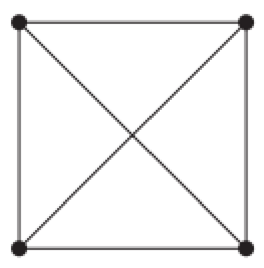


FIGURE 2 The Graph K_4.

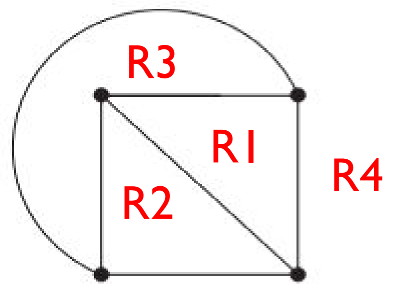


FIGURE 3 K_4 Drawn with No Crossings.

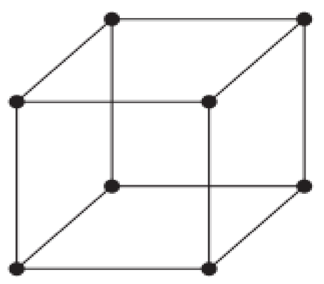


FIGURE 4 The Graph Q_3.

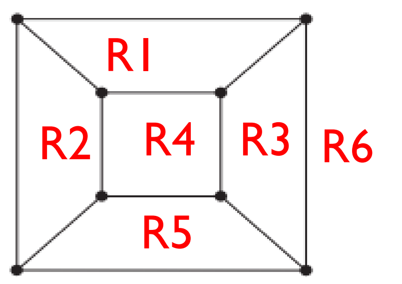


FIGURE 5 A Planar Representation of Q_3.

5. A complete graph of five vertices is non planar.

PLANAR GRAPHS:

1. **EULER'S FORMULA :** Let G be a connected planar simple graph with e edges and v vertices. Let r be the number of regions in a planar representation of G. Then $r = e - v + 2$.

Poof:

We will use Mathematical induction

(a) Base Case:

For n=1 i.e. For number of edge = 1.

$r_1 = e_1 - v_1 + 2$

$r_1 = 1-2+2$

$r_1 = 1$ (TRUE)

(b) Induction Hypothesis:

Assume the equation is true for n=k i.e. for number of edges = k

$r_k = e_k - v_k + 2$ is true for G_k

(c) Induction Step:

Let $\{a_{k+1}, b_{k+1}\}$ be the edge that is added to G_k to obtain G_{k+1}.

PLANAR GRAPHS:

(c) Induction Step:

Let $\{a_{k+1}, b_{k+1}\}$ be the edge that is added to G_k to obtain G_{k+1}.

There are two case to be considered:

i. Both a_{k+1} and b_{k+1} are already in G_k

$r_{k+1} = r_k+1$

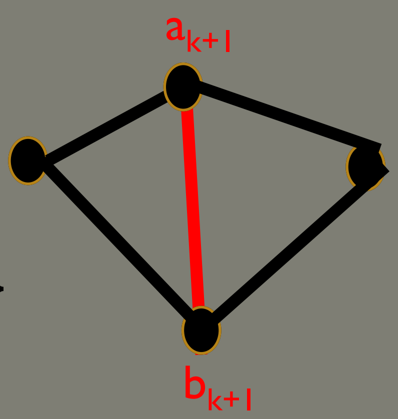
$e_{k+1} = e_k + 1$

$v_{k+1} = v_k$ {Because both vertices are in G_k}

$r_{k+1} = e_{k+1} - v_{k+1} + 2$

$r_k+1 = e_k + 1 - v_k + 2$

$\mathbf{r_k = e_k - v_k + 2}$ **which is true**



ii. One vertices is added to G_k

$r_{k+1} = r_k$

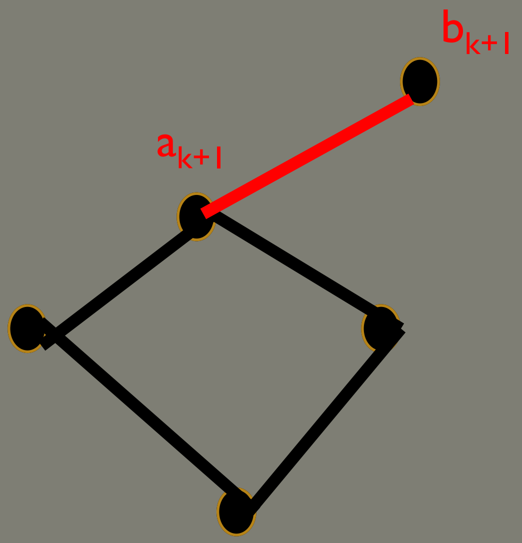
$e_{k+1} = e_k + 1$

$v_{k+1} = v_k +1$ {Because one vertices is added in G_k}

$r_{k+1} = e_{k+1} - v_{k+1} + 2$

$r_k = e_k + 1 - (v_k + 1) + 2$

$\mathbf{r_k = e_k - v_k + 2}$ **which is true**



Hence by induction method Euler's Formula is proved

PLANAR GRAPHS:

- 4 Suppose that a connected planar simple graph has 20 vertices, each of degree 3. Into how many regions does a representation of this planar graph split the plane?
  Solution:

This graph has 20 vertices, each of degree 3, so $v = 20$
Now using Handshaking theorem ,

$$2e = (20*3)$$
$$e = 30$$

Consequently, from Euler's formula, the number of regions is

$$r = e - v + 2$$
$$r = 30 - 20 + 2$$
$$r = 12.$$

PLANAR GRAPHS:

Corollary of Euler's Theorem:

1. **If G is a connected planar simple graph with e edges and v vertices, where $v \geq 3$, then $e \leq 3v - 6$**

Proof:

Let G(v, e) be the connected planar simple graph. A connected planar simple graph drawn in the plane divides the plane into regions, say **r** of them.

The degree of each region is at least three. (Because the graphs discussed here are simple graphs, no multiple edges that could produce regions of degree two, or loops that could produce regions of degree one, are permitted.) Degree of region is defined to be the number of edges on the boundary of this region.

Sum of the degrees of the regions is exactly twice the number of edges in the graph, because each edge occurs on the boundary of a region exactly twice . Because each region has degree greater than or equal to three, it follows that

$$\mathbf{2e} = \sum_{all\ regions\ R} \deg(R) \geq \mathbf{3r}$$

$$(2/3)e \geq r$$

Using $r = e - v + 2$ (Euler's formula),

we obtain $v - e + r = 2$

$$v - e + (2/3)e \geq 2$$

$$\mathbf{e \leq 3v - 6.}$$ **Hence proved**

PLANAR GRAPHS:

- Show that K_5 is nonplanar using Corollary 1.

Solution:

The graph K_5 has five vertices and 10 edges.

However, the inequality $e \leq 3v - 6$ is not satisfied for this graph because

$10 \leq 3*5 - 6$

$10 \leq 9$ which is false. Therefore, K5 is not planar.

2. If G is a connected planar simple graph, then G has a vertex of degree not exceeding five

Poof:

If G has at least three vertices, by Corollary 1 we know that

$e \leq 3v - 6,$

$2e \leq 6v - 12.$

$2e + 12 \leq 6v$

If the degree of every vertex were at least six, then because

$2e = \sum_{v \in V} \deg(v)$ (by the handshaking theorem), we would have

$2e + 12 \geq 6v + 12.$

But this contradicts the inequality $2e + 12 \leq 6v$. It follows that there must be a vertex with degree no greater than five.

PLANAR GRAPHS:

Mathematical Foundation for Computer Science

Prepared by: Er. Ankit Kharel

Nepal college of information technology

GRAPH THEORY

SHORTEST PATH ALGORITHM:

1. The shortest paths is defined as the smallest weighted path from the starting vertex to the goal vertex out of all other paths in the weighted graph. Here, you can think "weighted" in the weighted path means the reaching cost to the goal vertex (some vertex).

2. One algorithm for finding the shortest path from a starting node to a target node in a weighted graph is **Dijkstra's algorithm.**

3. The graph can either be directed or undirected.

4. One stipulation to using the algorithm is that the graph needs to have a nonnegative weight on every edge.

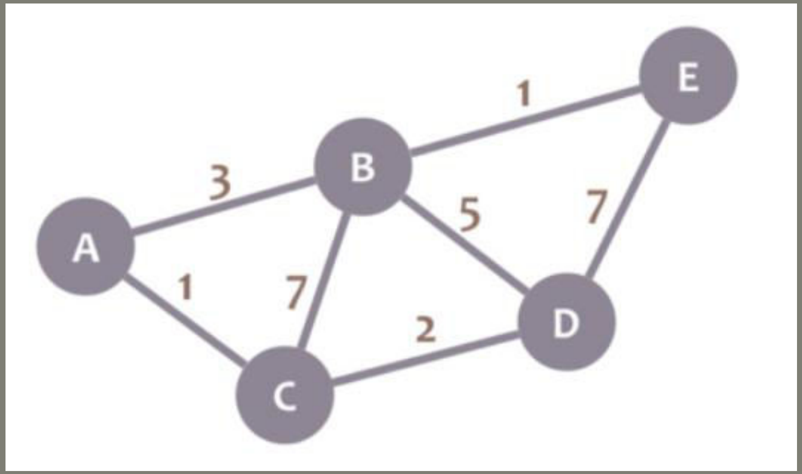
Edge Relaxation:

For the edge from the vertex u to the vertex v,

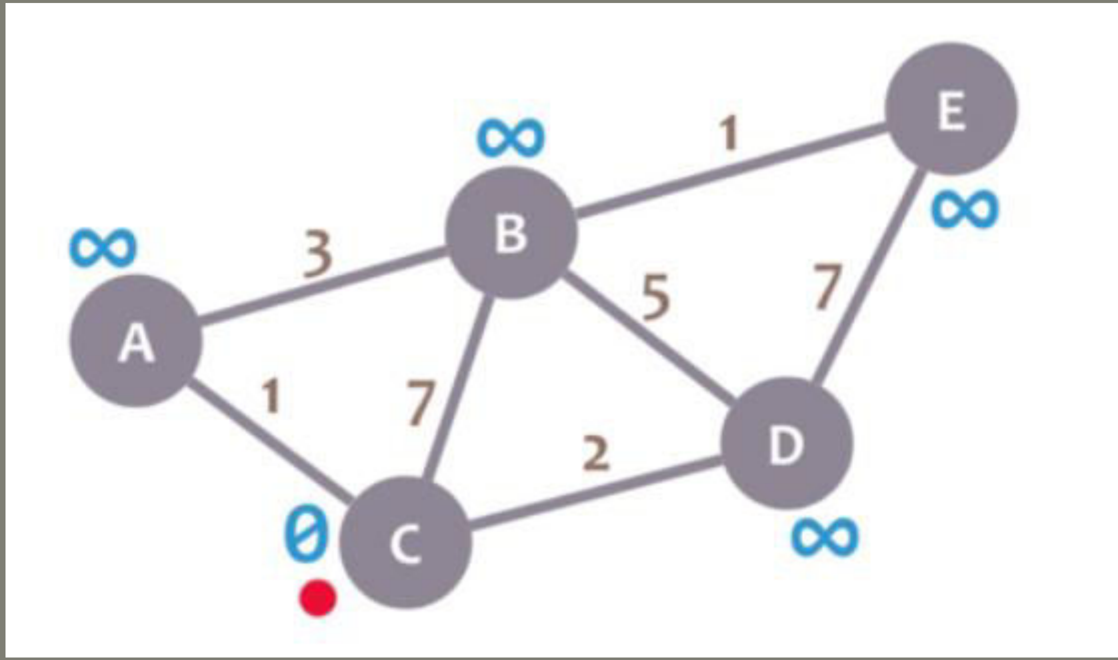
if ($d[u]+w(u,v)<d[v]$) is satisfied,

update $d[v] = d[u]+w(u,v)$

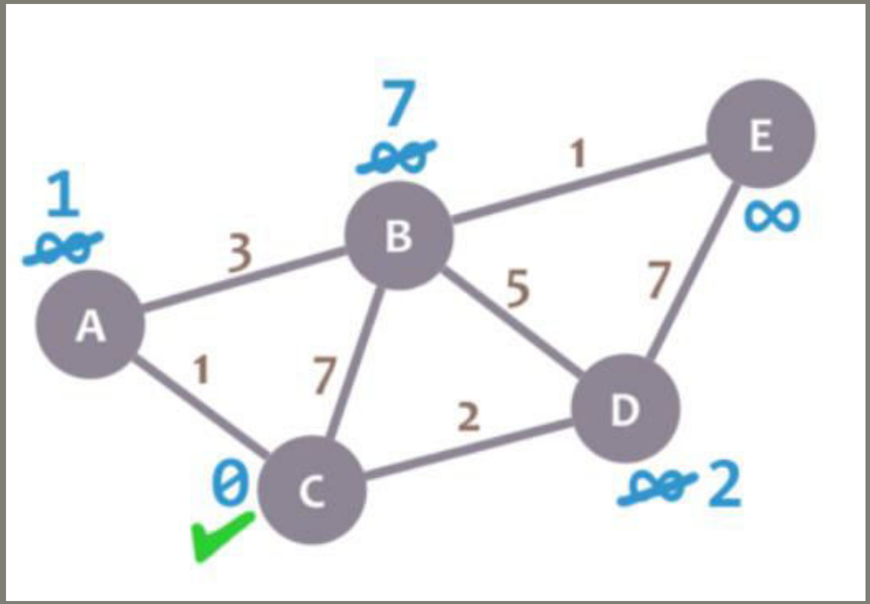
Let's calculate the shortest path between node C and the other nodes in our graph:



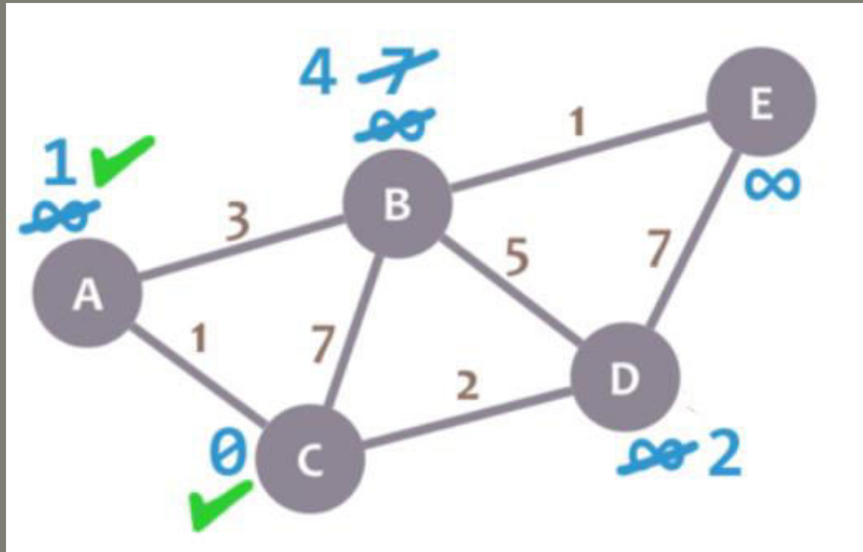
1. For node C, this distance is 0. For the rest of nodes, as we still don't know that minimum distance, it starts being infinity (∞):



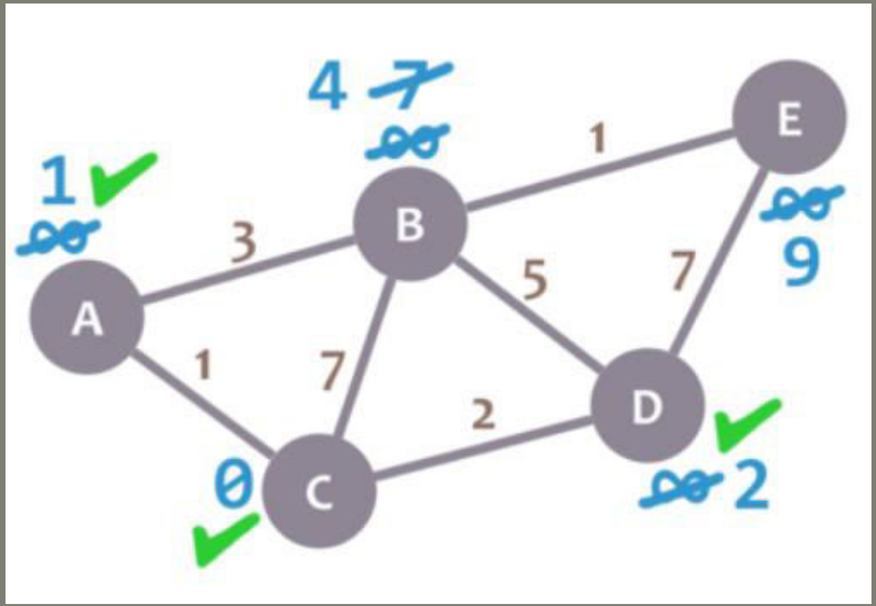
2. Now, we check the neighbors of our current node (A, B and D) in no specific order and perform relaxation if required.



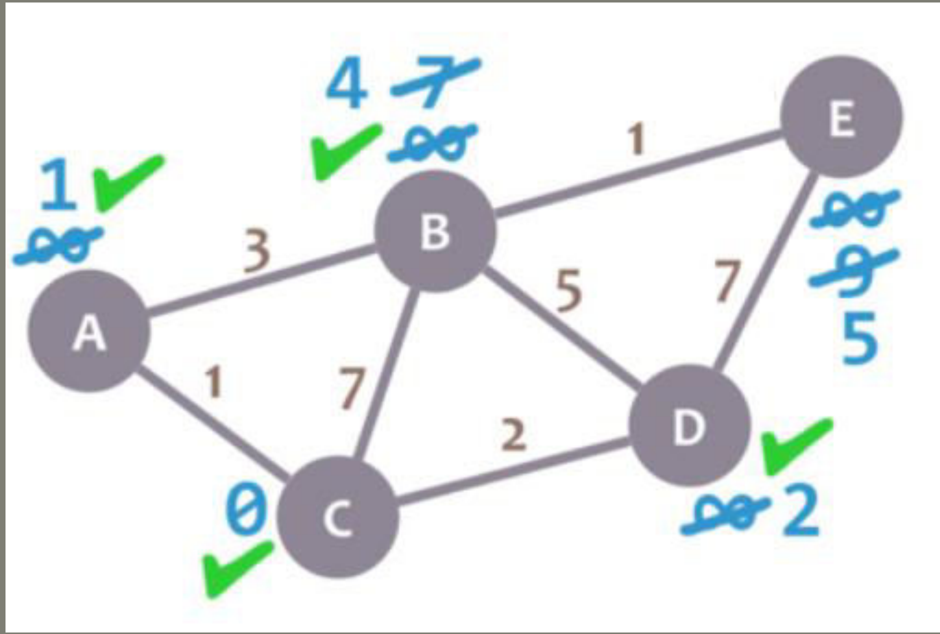
3. We now need to pick a new *current node*. That node must be the unvisited node with the smallest minimum distance .That's A and now we repeat the algorithm. We check the neighbors of our current node(which is B), ignoring the visited nodes.



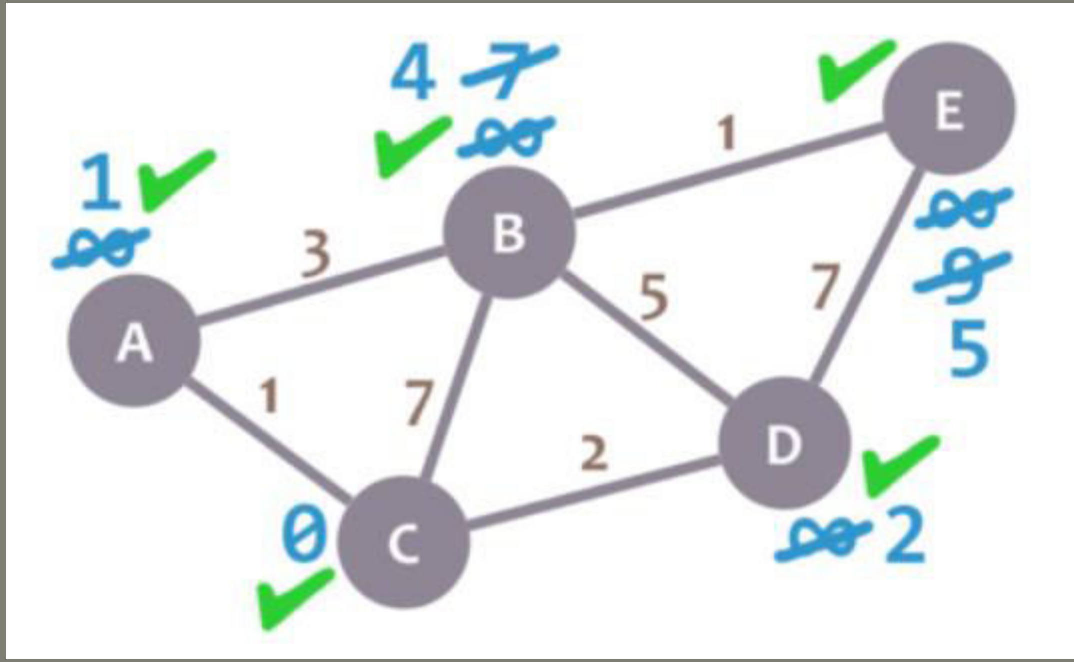
4. Now, select the node D(as it has the minimum distance among unvisited node) and repeat the algorithm.



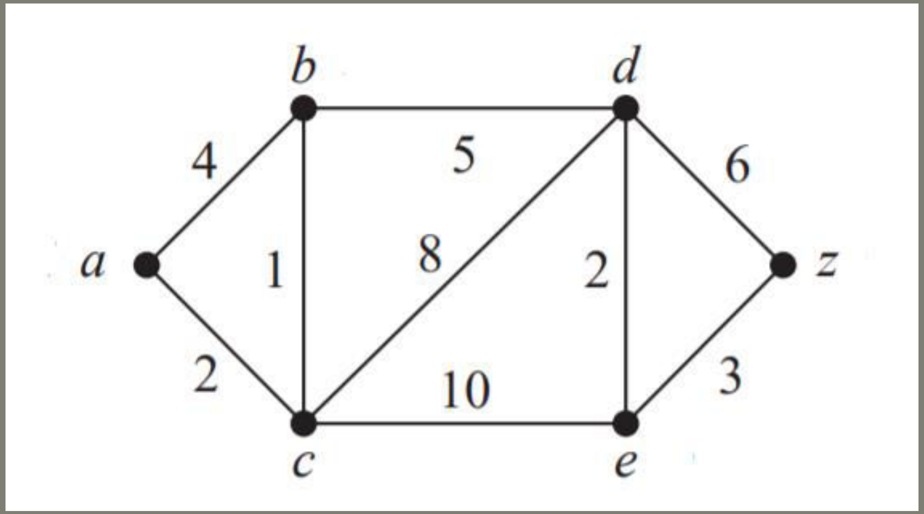
5. Now, select the node B(as it has the minimum distance among unvisited node) and repeat the algorithm.



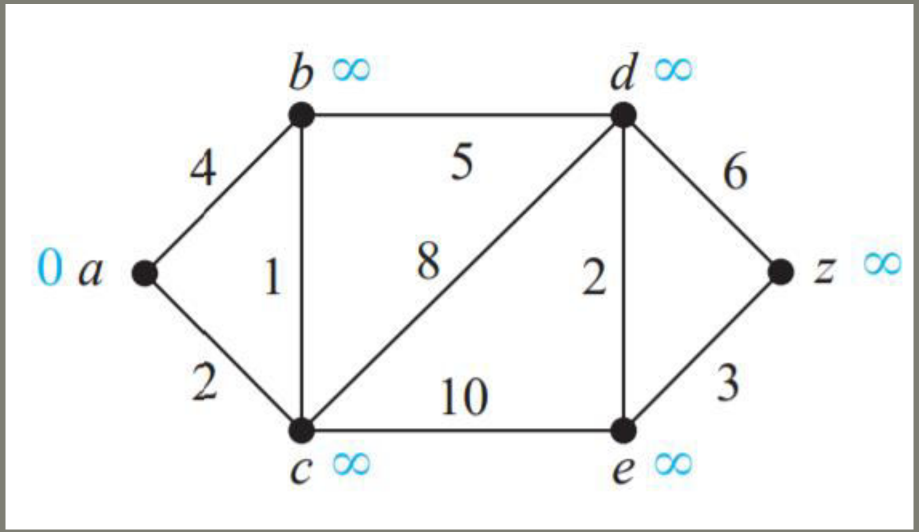
6. E doesn't have any non-visited neighbors, so we don't need to check anything. We mark it as visited.



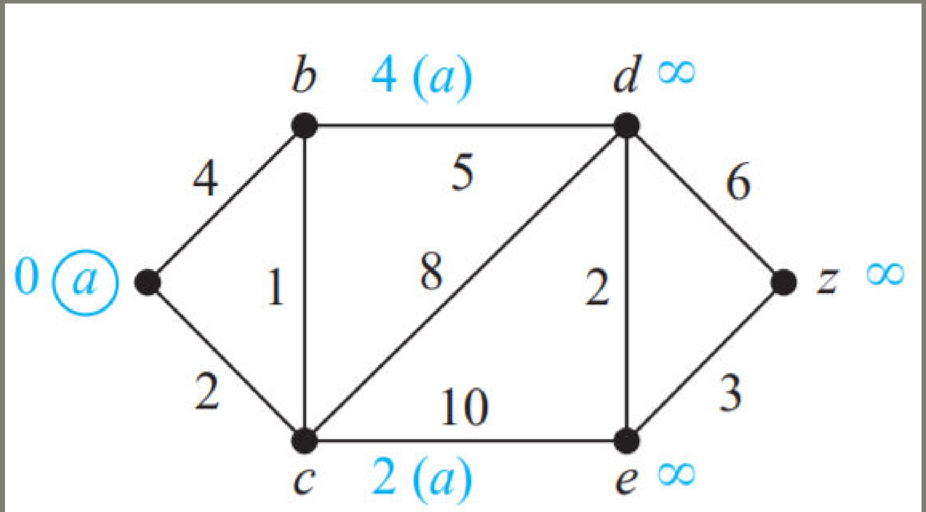
Use Dijkstra's algorithm to find the length of a shortest path between the vertices a and z in the weighted graph displayed in below Figure.



Step: 1

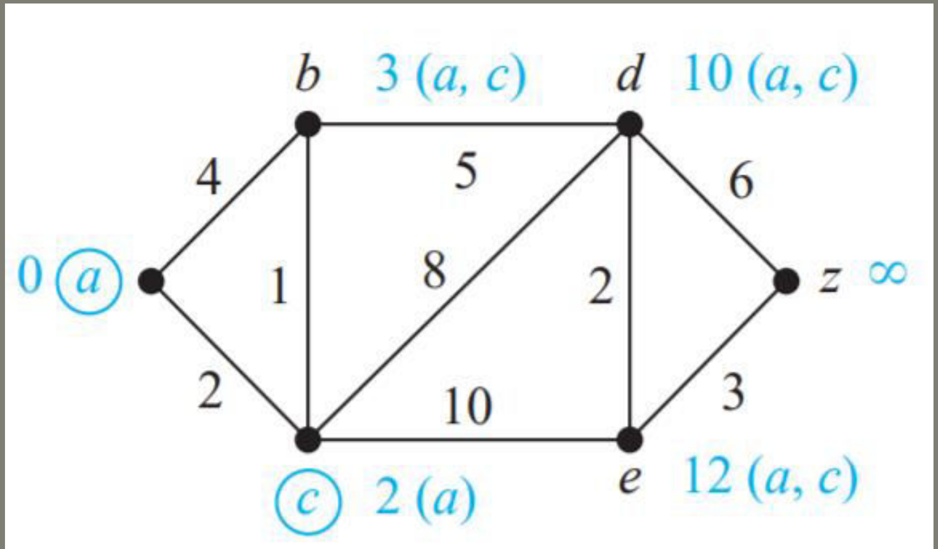


Step: 2



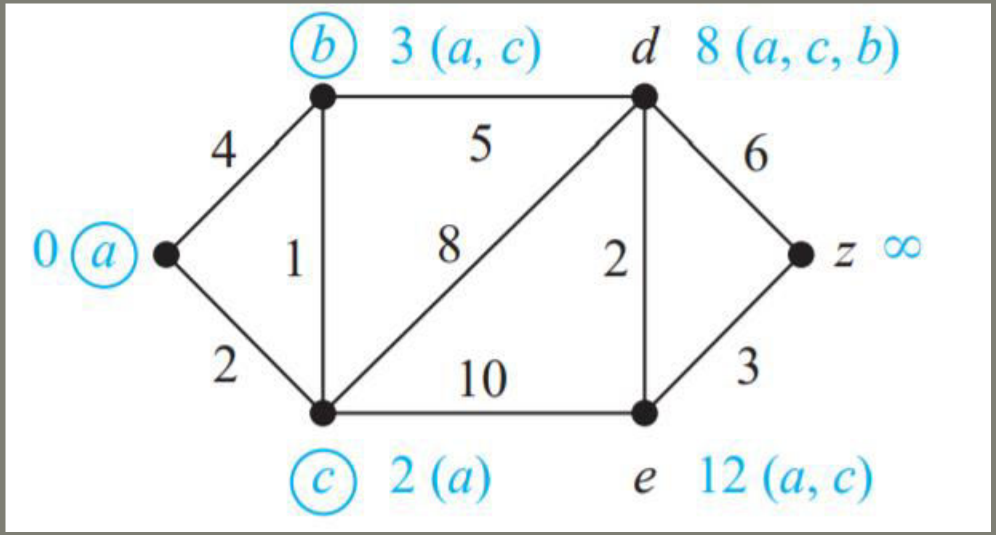
b
4 (a)
d ∞
4
5
6
0 a
1
8
2
z ∞
2
10
3
c
2 (a)
e ∞

Step: 3

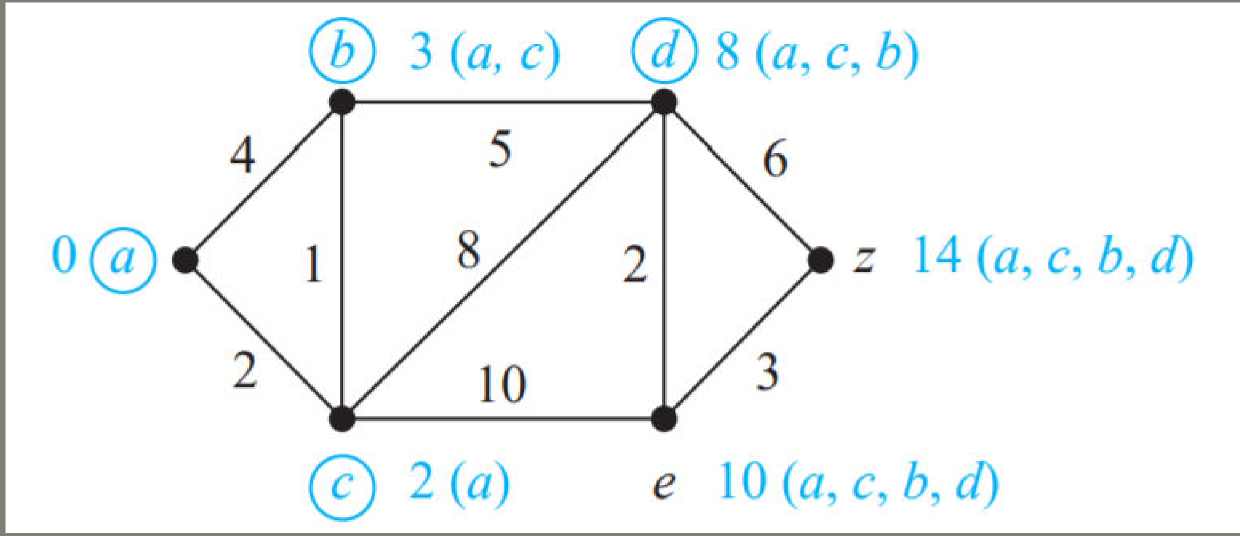


b
3 (a, c)
d
10 (a, c)
4
5
6
0 a
1
8
2
z ∞
2
10
3
c
2 (a)
e
12 (a, c)

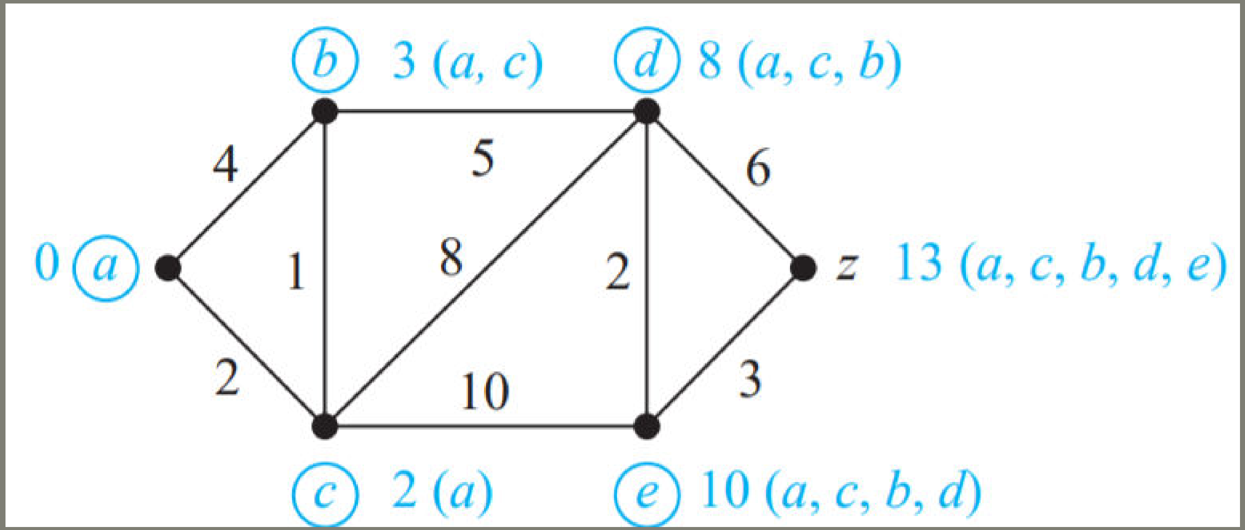
Step: 4


b 3 (a, c)
d 8 (a, c, b)
4
5
6
0 a
1
8
2
z ∞
2
10
3
c 2 (a)
e 12 (a, c)

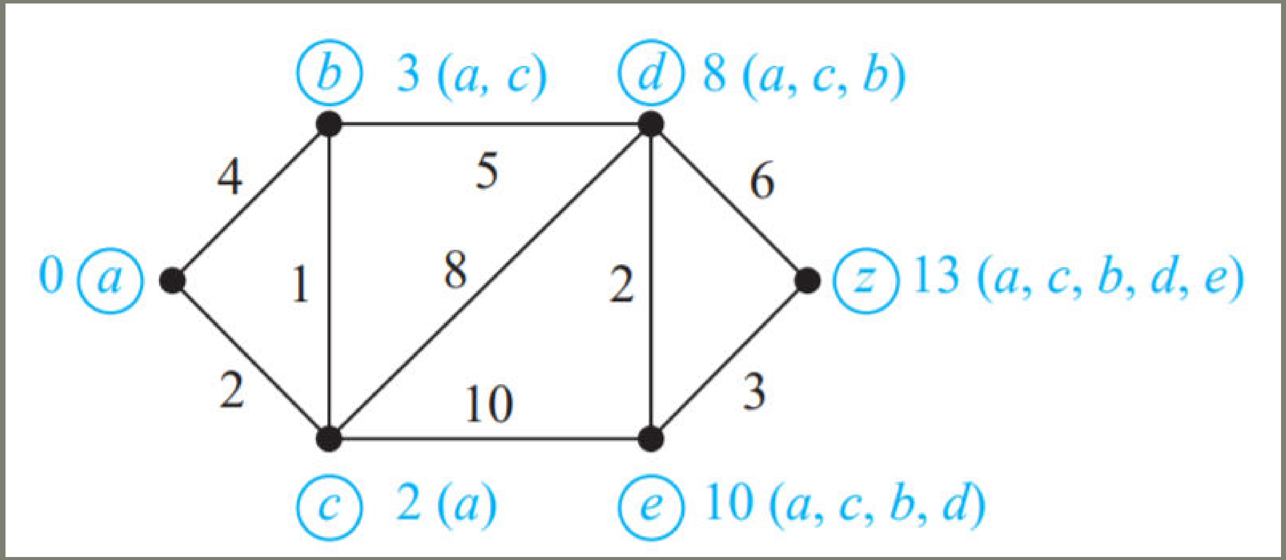
Step: 5


b 3 (a, c)
d 8 (a, c, b)
4
5
6
0 a
1
8
2
z 14 (a, c, b, d)
2
10
3
c 2 (a)
e 10 (a, c, b, d)

Step: 6


b 3 (a, c)
d 8 (a, c, b)
4
5
6
0 a
1
8
2
z 13 (a, c, b, d, e)
2
10
3
c 2 (a)
e 10 (a, c, b, d)

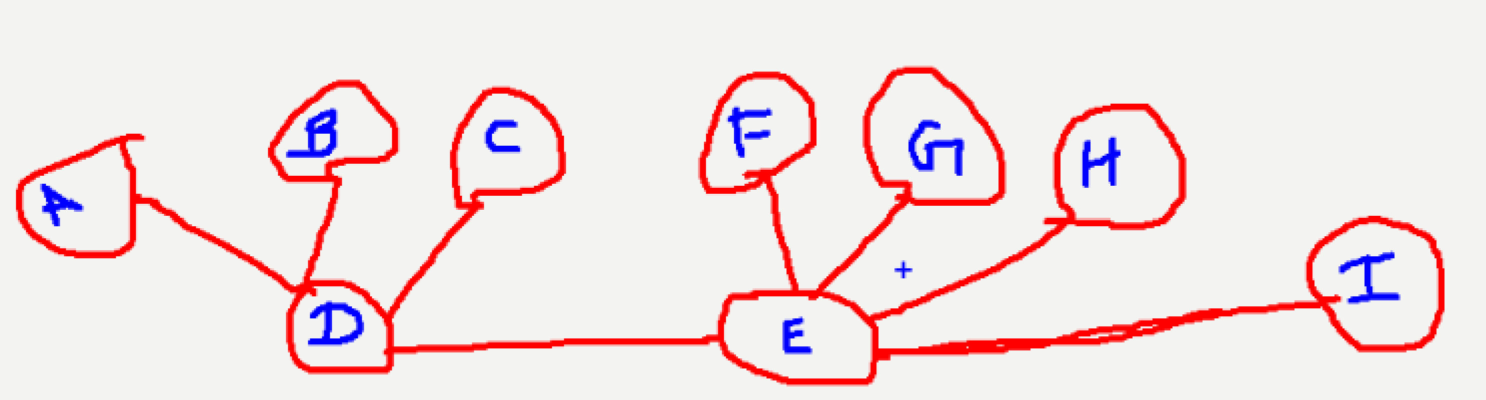
Step: 7


b 3 (a, c)
d 8 (a, c, b)
4
5
6
0 a
1
8
2
z 13 (a, c, b, d, e)
2
10
3
c 2 (a)
e 10 (a, c, b, d)

GRAPH TRAVERSAL:

- **Graph traversal** (also known as **graph search**) refers to the process of visiting (checking and/or updating) each vertex in a graph. Such traversals are classified by the order in which the vertices are visited. [Tree traversal] is a special case of graph traversal.

1. **BFS(Breadth First Search):**
   The Breadth First Search (BFS) traversal is an algorithm, which is used to visit all of the nodes of a given graph. In this traversal algorithm one node is selected and then all of the adjacent nodes are visited one by one. After completing all of the adjacent vertices, it moves further to check another vertices and checks its adjacent vertices again. BFS uses Queue Data structure.



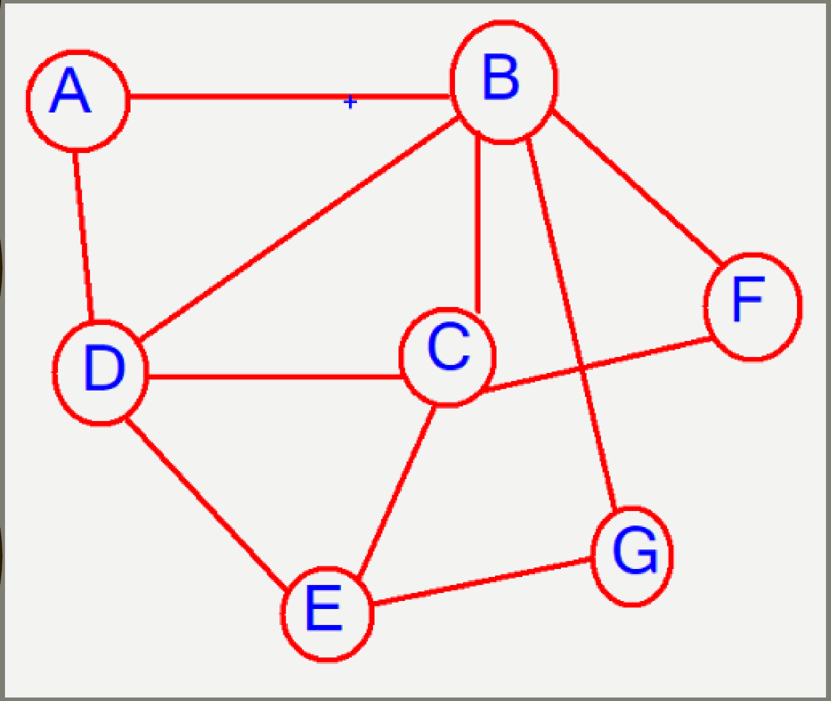
BFS : D, A, B, C, E, F, G, H, I

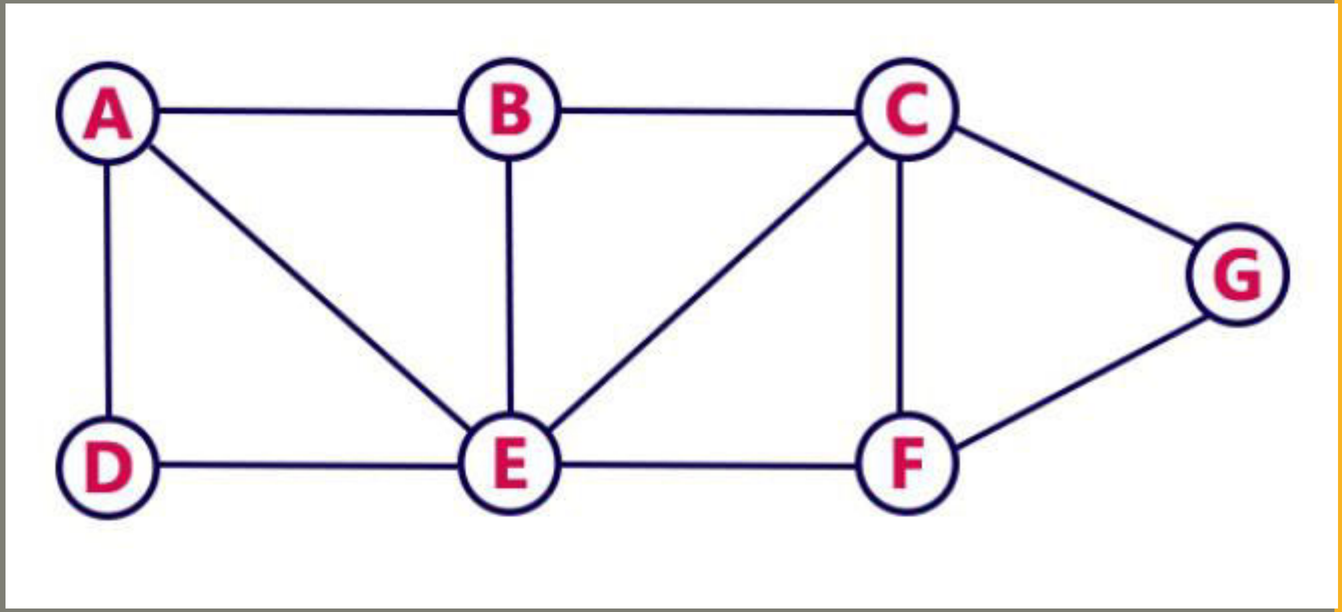
OR

BFS : D, E, C, B, A, I, H, G, F

GRAPH TRAVERSAL:

BFS(Breadth First Search):





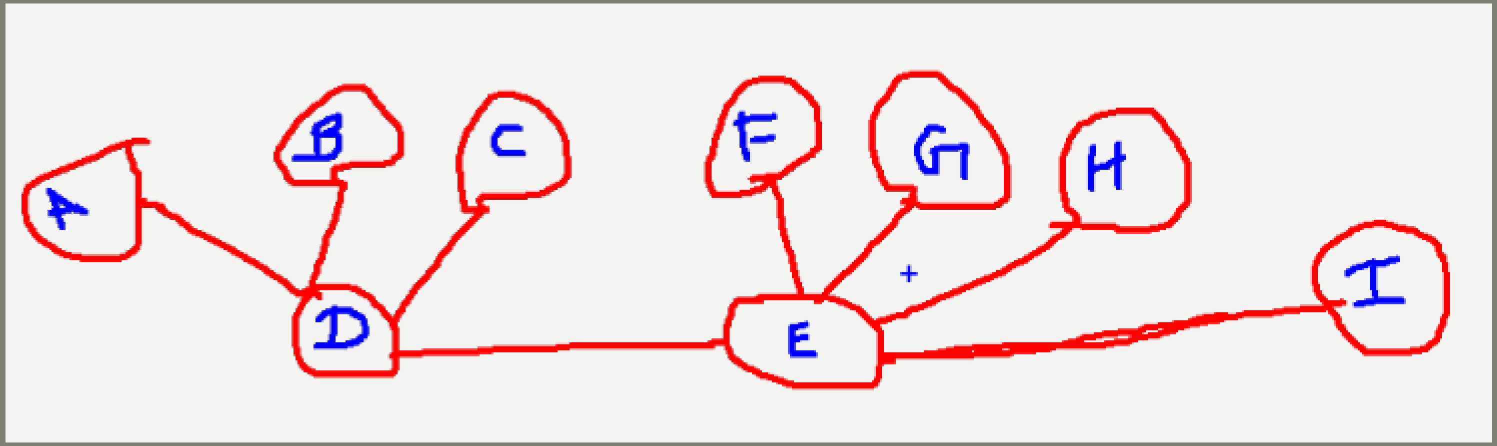
BFS : A, B, D, C, F, G, E

BFS : A, B, D, E, C, F, G

GRAPH TRAVERSAL:

2. **DFS(Depth First Search):**

   The Depth First Search (DFS) is a graph traversal algorithm. In this algorithm one starting vertex is given, and when an adjacent vertex is found, it moves to that adjacent vertex first and try to traverse in the same manner. DFS uses Stack Data structure.



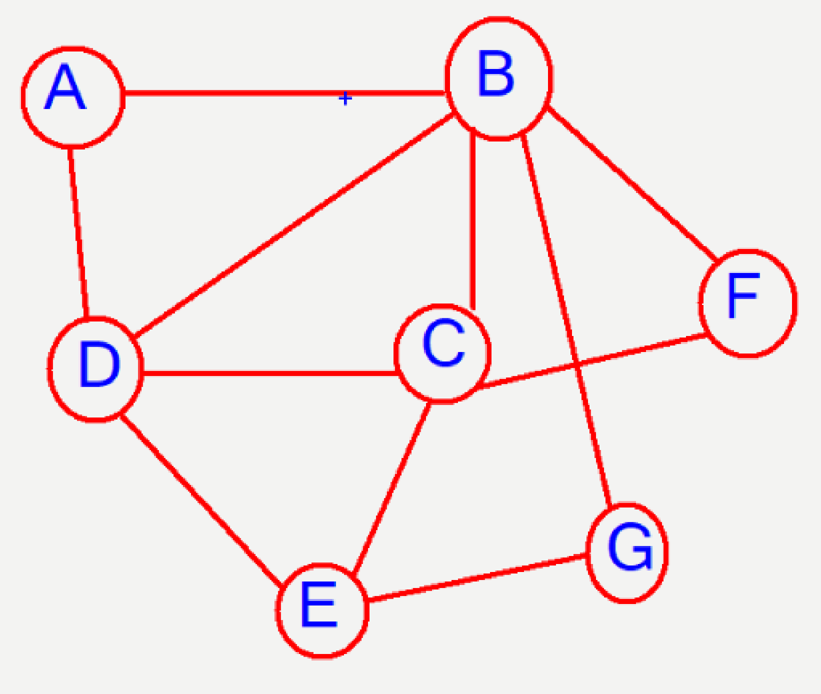
DFS : D, E, I, H, G, F, C, B, A

OR

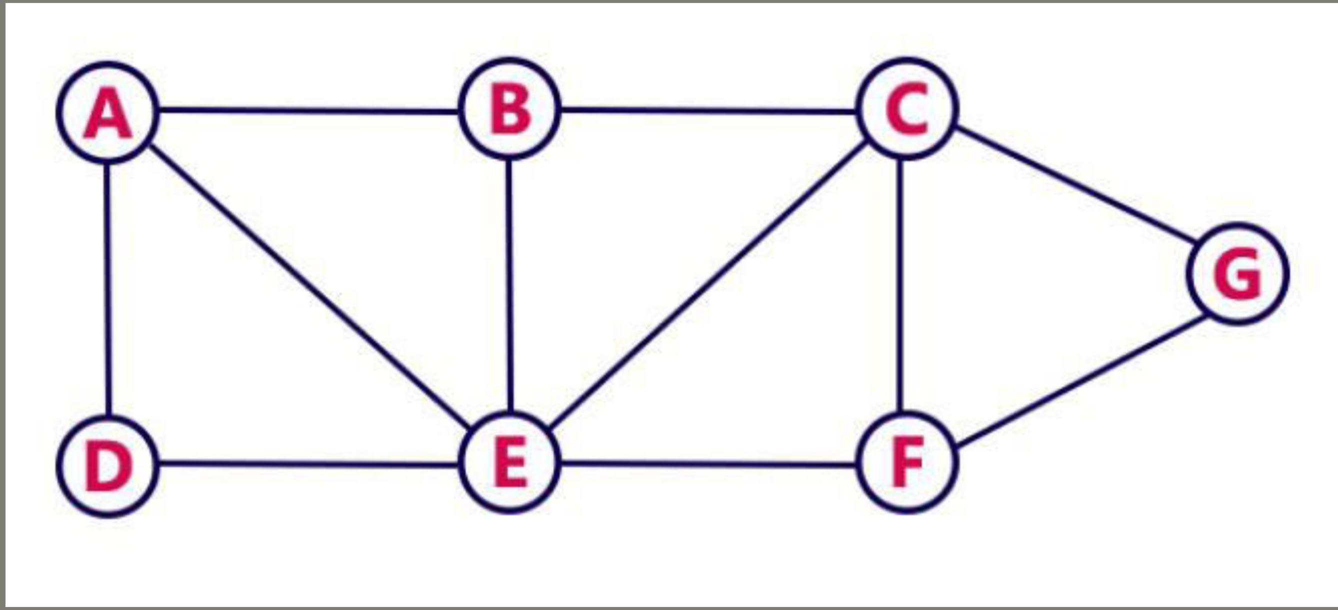
DFS : D, A, B, C, E, F, G, H, I

GRAPH TRAVERSAL:

2. **DFS(Depth First Search):**



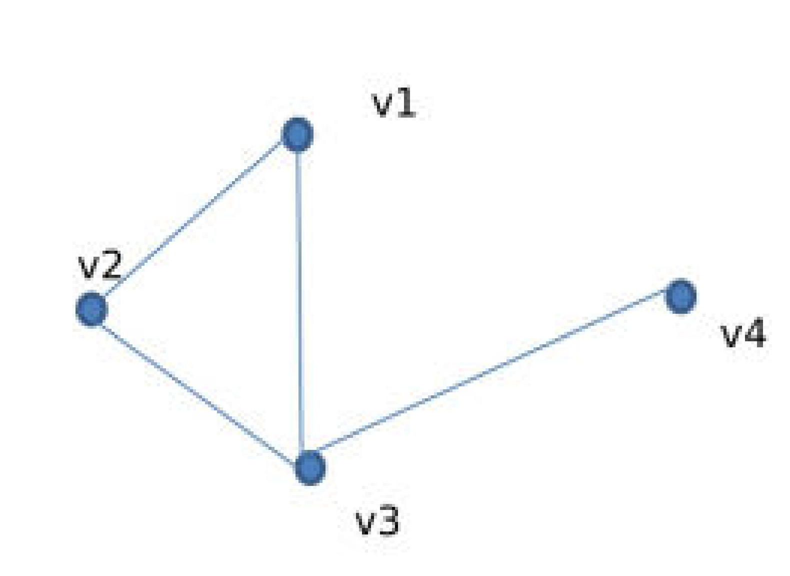
DFS : A, B, D, C ,E, G, F
OR
DFS: A, D, B, F, C, E, G



DFS : A, B, C, G, F, E, D
OR
DFS: A, E, D, B, C, F, G

DEGREE SEQUENCE:

- **Degree Sequence** of a graph is the list of degree of all the vertices of graph in non-increasing or non- decreasing order.

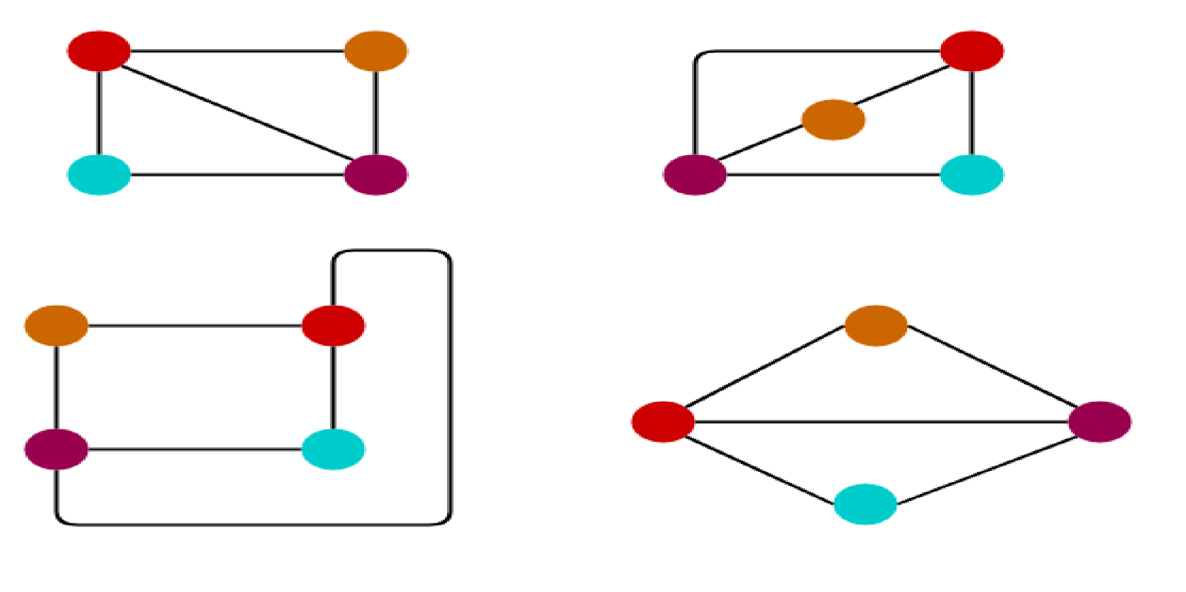


Non- increasing : 3, 2, 2, 1
Non- decreasing : 1, 2, 2, 3

ISOMORPHISM:

- Graph Isomorphism is a phenomenon of existing the same graph in more than one forms. Such graphs are called as **Isomorphic graphs**. When two simple graphs are isomorphic, there is a one-to-one correspondence between vertices of the two graphs that preserves the adjacency relationship.

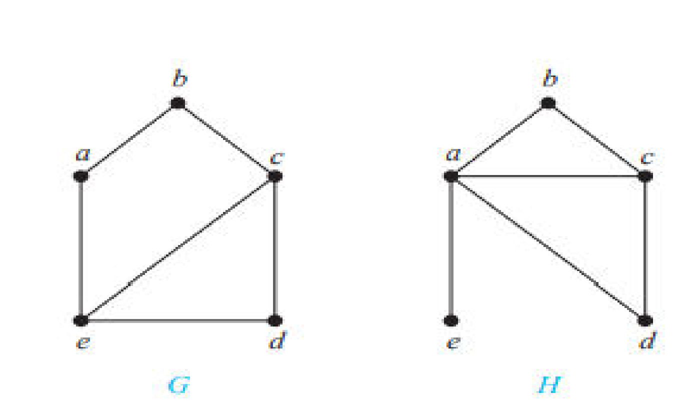
- Following are the necessary condition for Two Graphs G and G to be isomorphic:
  1. Both Graphs should have same number of edges
  2. Both Graphs should have same number of vertices
  3. Degree sequence of both Graphs should be same
  4. Their edge connectivity is retained



Graph Isomorphism Example

ISOMORPHISM:

- Show that the graphs G = (V , E) and H = (W, F), displayed in Figure are isomorphic or not.

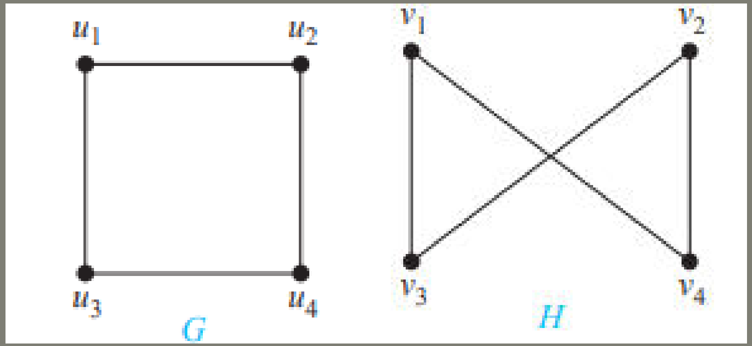


- Both Graph G and H have same number of vertices: 5
- Both Graph has same number of edges: 6
- Degree Sequence of G is : 3, 3, 2, 2, 2
- Degree Sequence of H is : 4, 3, 2, 2, 1

Since the degree sequence of Graphs are different. They are not isomorphic

ISOMORPHISM:

- Show that the graphs G = (V , E) and H = (W, F), displayed in Figure 8, are isomorphic



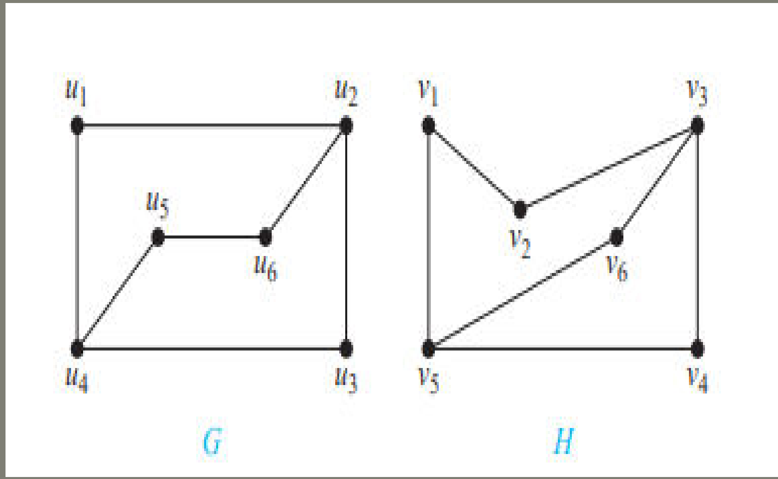
- Both Graph G and H have same number of vertices: 4
- Both Graph has same number of edges: 4
- Degree Sequence of G is : 2, 2, 2, 2
- Degree Sequence of H is : 2, 2, 2, 2
- Finding Correspondence between vertices:

  (i)$u_1=v_1$
  (ii)$u_2=v_4$
  (iii)$u_3=v_3$
  (iv)$u_4=v_2$

$$G = \begin{bmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix} \qquad H = \begin{bmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix}$$

Now, Find the adjacency matrix For G with ordering((u_1, u_2, u_3, u_4) and H for ordering(v_1,v_4,v_3,v_2)
Since both matrix are same the graph are isomorphic

- Show that the graphs G = (V , E) and H = (W, F), displayed in Figure are isomorphic or not



- Both Graph G and H have same number of vertices: 6
- Both Graph has same number of edges: 7
- Degree Sequence of G is : 3, 3, 2, 2, 2, 2
- Degree Sequence of H is : 3, 3, 2, 2, 2, 2
- Finding Correspondence between vertices:

  (i)$u_1=v_6$
  (ii)$u_2=v_3$
  (iii)$u_3=v_4$
  (iv)$u_4=v_5$
  (v)$u_5=v_1$
  (vi)$u_6=v_2$

$$G = \begin{bmatrix} 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 \end{bmatrix} \quad H = \begin{bmatrix} 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 \end{bmatrix}$$

Now, Find the adjacency matrix For G with ordering((u_1, u_2, u_3, u_4, u_5, u_6) and H for ordering(v_6, v_3 v_4, v_5, v_1, v_2)
If both matrix are same then the graph are isomorphic

MATHEMATICAL FOUNDATION FOR COMPUTER SCIENCE

Prepared by: Er. Ankit Kharel

Nepal college of information technology

GRAPH THEORY

TREES:

- Tree is a connected undirected graph with no simple circuits
- Because a tree cannot have a simple circuit, a tree cannot contain multiple edges or loops. Therefore any tree must be a simple graph.

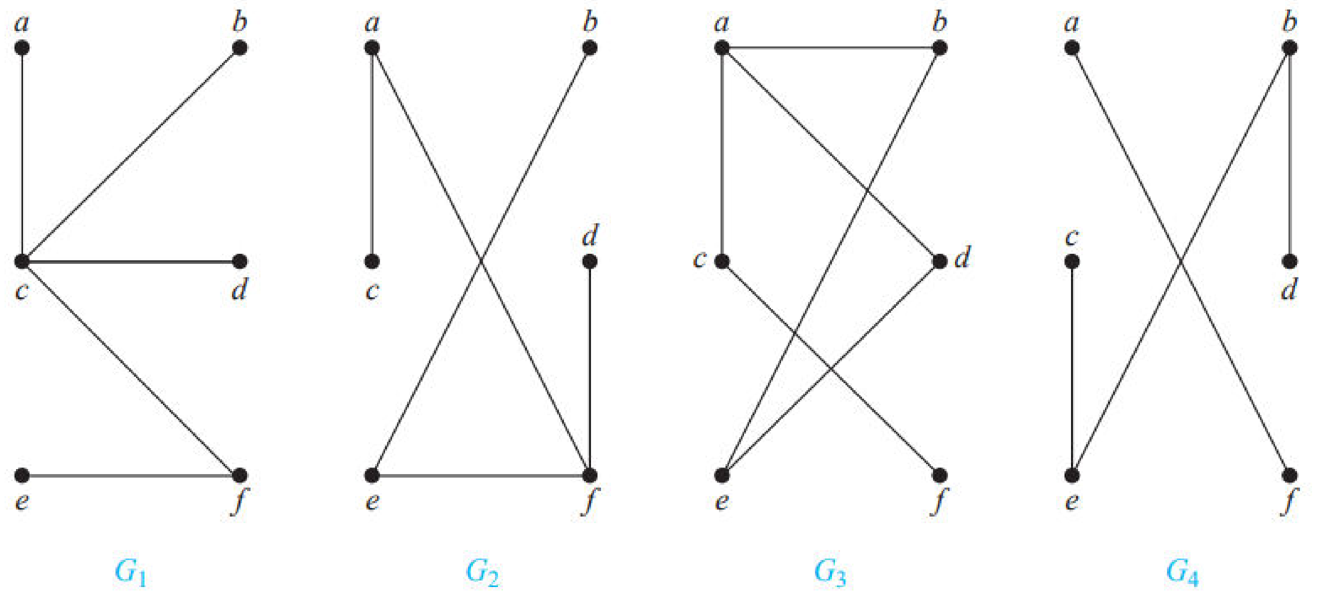


FIGURE 2 **Examples of Trees and Graphs That Are Not Trees.**

- G1 and G2 are trees
- G3 is not a tree because e, b, a, d, e is a simple circuit in this graph. Finally, G4 is not a tree because it is not connected.

ROOTED TREES:

- A rooted tree is a tree in which one vertex has been designated as the root and every edge is directed away from the root.

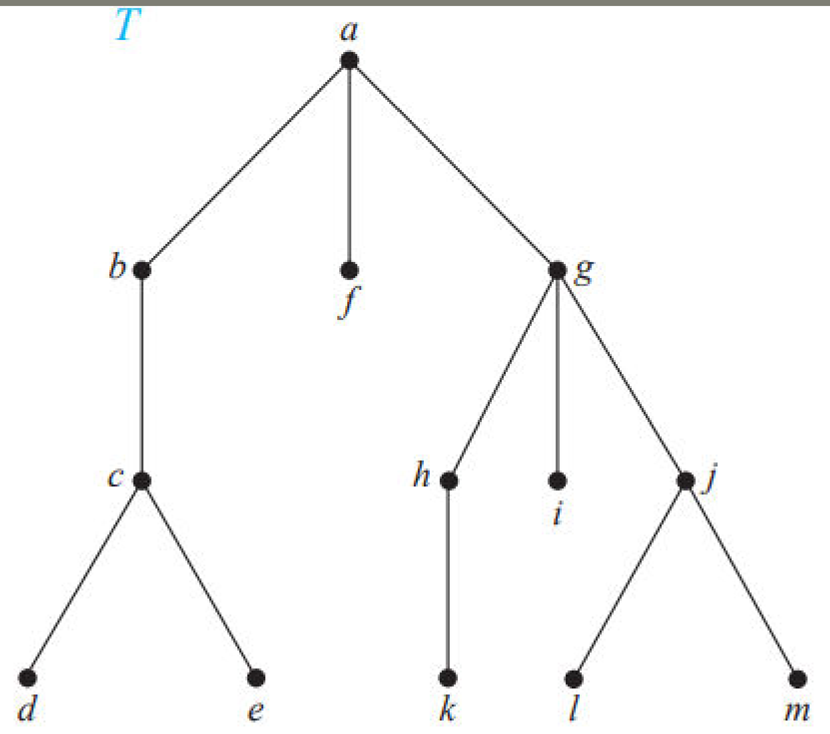
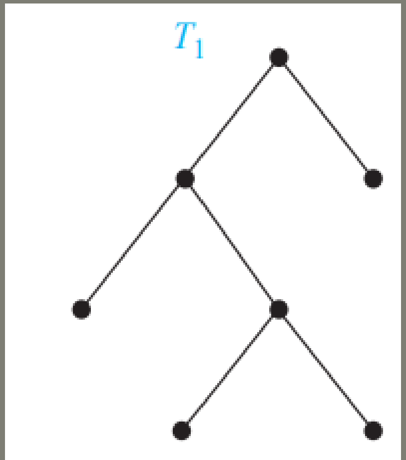


FIGURE 5 **A Rooted Tree *T*.**

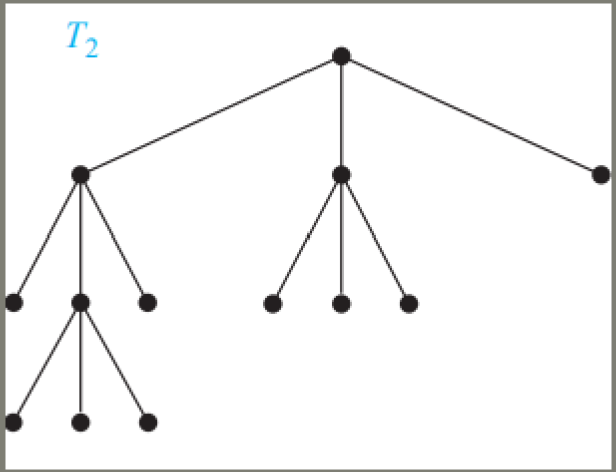
- A vertex of a rooted tree is called a l**eaf** if it has no children.
- Vertices that have children are called **internal vertices**.
- Vertices with the same parent are called **siblings**
- The **ancestors** of a vertex other than the root are the vertices in the path from the root to this vertex, excluding the vertex itself and including the root.
- The **descendants** of a vertex v are those vertices that have v as an ancestor

ROOTED TREES:

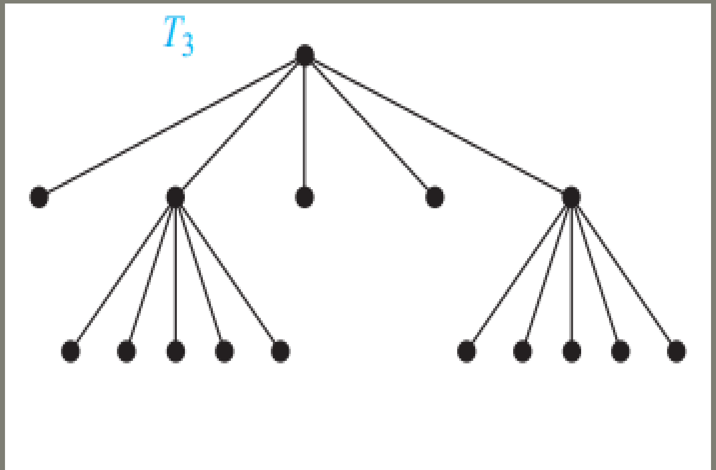
- A rooted tree is called an **m-ary** tree if every internal vertex has no more than m children. The tree is called a **full m-ary** tree if every internal vertex has exactly m children. An m-ary tree with m = 2 is called a binary tree.



T1 is a full binary tree because each of its internal vertices has two children



T2 is a full 3-ary tree because each of its internal vertices has three children

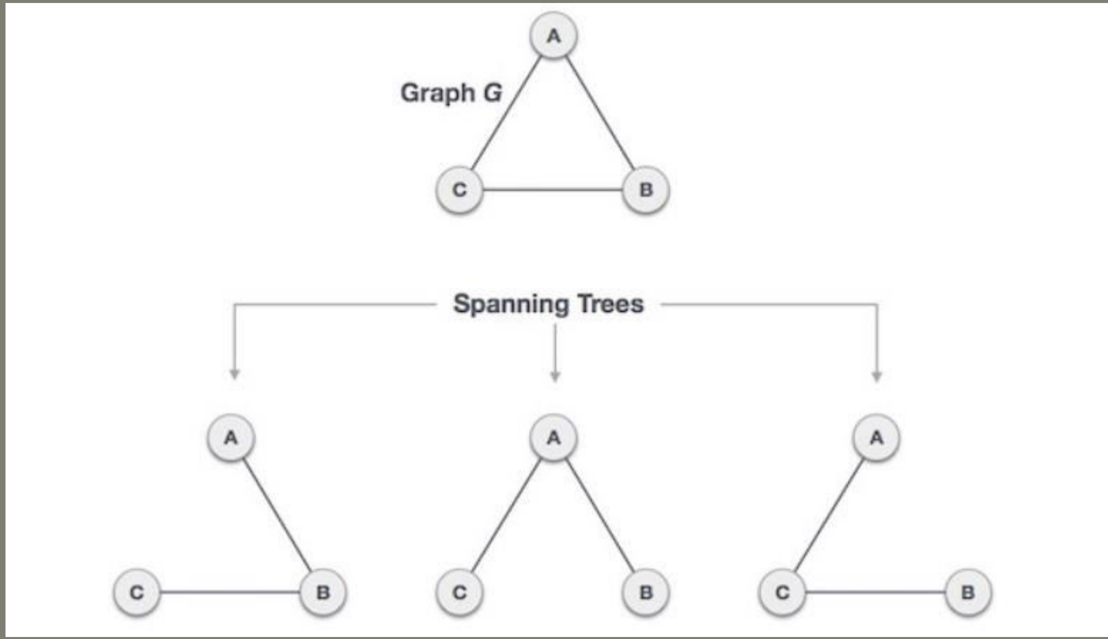


In T3 each internal vertex has five children, so T3 is a full 5-ary tree

SPANNING TREE:

- A **spanning tree** is a subset of Graph G, which has all the vertices covered with minimum possible number of edges. Formally, Let G be a simple graph. A spanning tree of G is a subgraph of G that is a tree containing every vertex of G.

- Every connected and undirected Graph G has at least one spanning tree. A disconnected graph does not have any spanning tree, as it cannot be spanned to all its vertices.



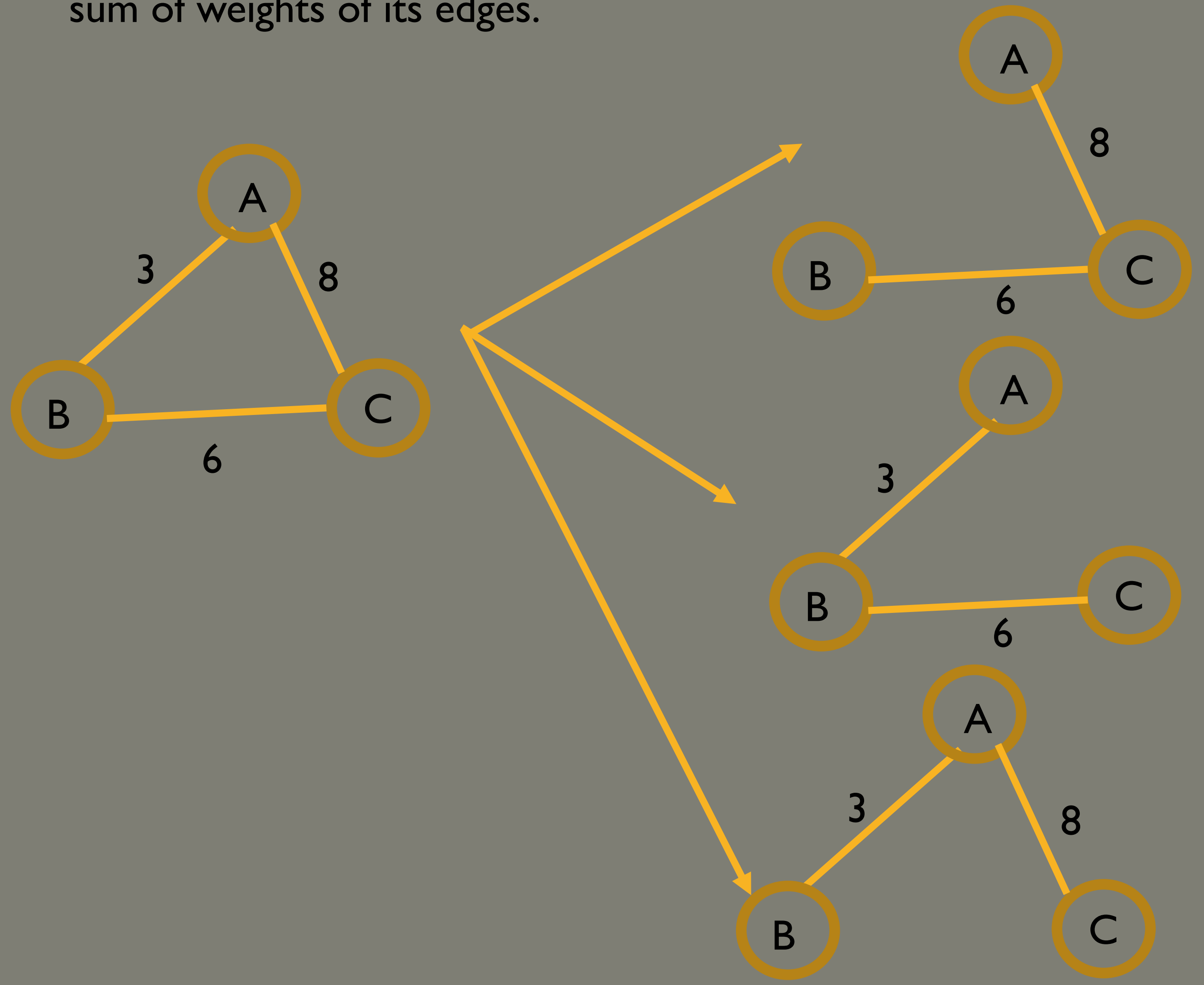
A complete undirected graph can have maximum $\mathbf{n^{n-2}}$ number of spanning trees, where **n** is the number of nodes. In the above addressed example, **n is 3,** hence $\mathbf{3^{3-2} = 3}$ spanning trees are possible.

SPANNING TREE:

- **General Properties of Spanning Tree**
  - ✓ A connected graph G can have more than one spanning tree.
  - ✓ All possible spanning trees of graph G, have the same number of edges and vertices.
  - ✓ The spanning tree does not have any cycle (loops).
  - ✓ Removing one edge from the spanning tree will make the graph disconnected, i.e. the spanning tree is **minimally connected**.
  - ✓ Adding one edge to the spanning tree will create a circuit or loop, i.e. the spanning tree is **maximally acyclic**.
  - ✓ Spanning tree has **n-1** edges, where **n** is the number of nodes (vertices).
  - ✓ A complete graph can have maximum n^{n-2} number of spanning trees.

MINIMUM SPANNING TREE(MST:

- A minimum spanning tree in a connected weighted graph is a spanning tree that has the smallest possible sum of weights of its edges.



Total cost = 14

Total cost = 9
Minimum Spanning Tree

Total cost = 11

MINIMUM SPANNING TREE(MST:

- There are two algorithms for finding Minimum Spanning Tree:
  (a) Kruskal’s Algorithm
  (b) Prim’s Algorithm

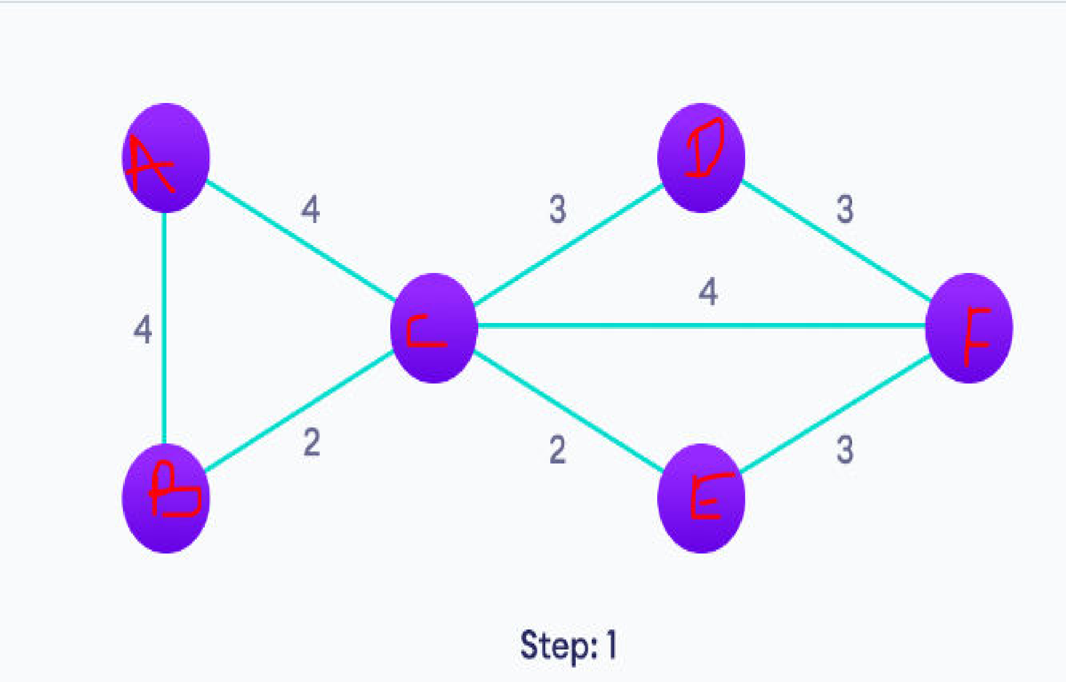
(a)KRUSKAL’s ALGORITHM:

The steps for implementing Kruskal's algorithm are as follows:

- Remove all the loops and parallel edges if present. In case of parallel edges, keep the one which has the least cost associated and remove all others.
- Sort all the edges from low weight to high
- Take the edge with the lowest weight and add it to the spanning tree. If adding the edge created a cycle, then reject this edge.
- Keep adding edges until we reach all vertices.

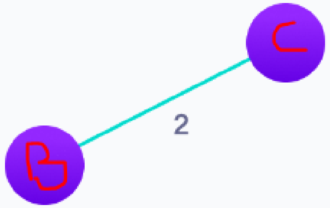
MINIMUM SPANNING TREE(MST:

Find The minimum spanning tree using Kruskal's Algorithm.



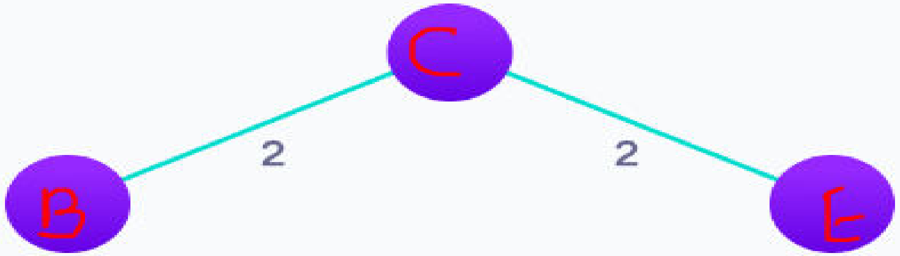
STEP: 1 Sort all the edges in ascending order

{B ,C}	{C ,E}	{D ,C}	{E ,F}	{D ,F}	{B ,A}	{A ,C}	{F ,C}
2	2	3	3	3	4	4	4



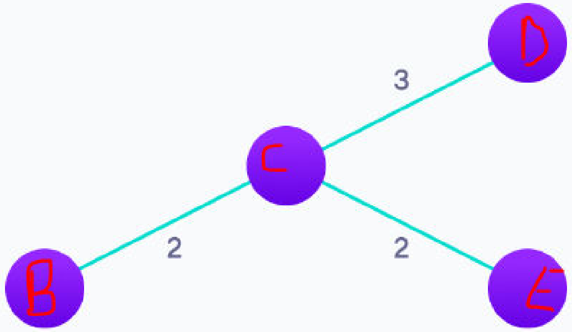
Step: 2

Choose the edge with the least weight, if there are more than 1, choose anyone



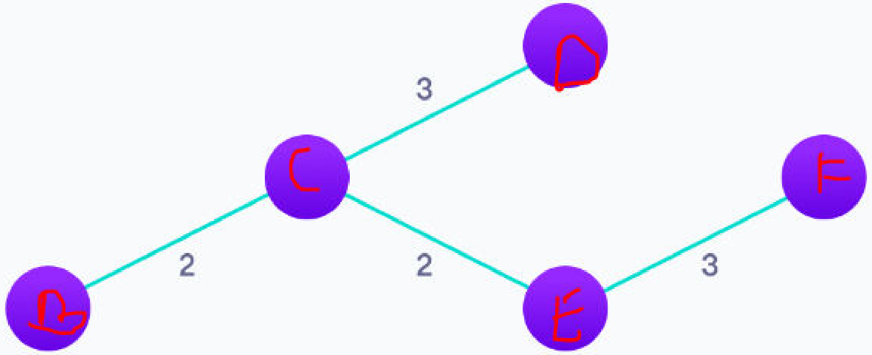
Step: 3

Choose the next shortest edge and add it



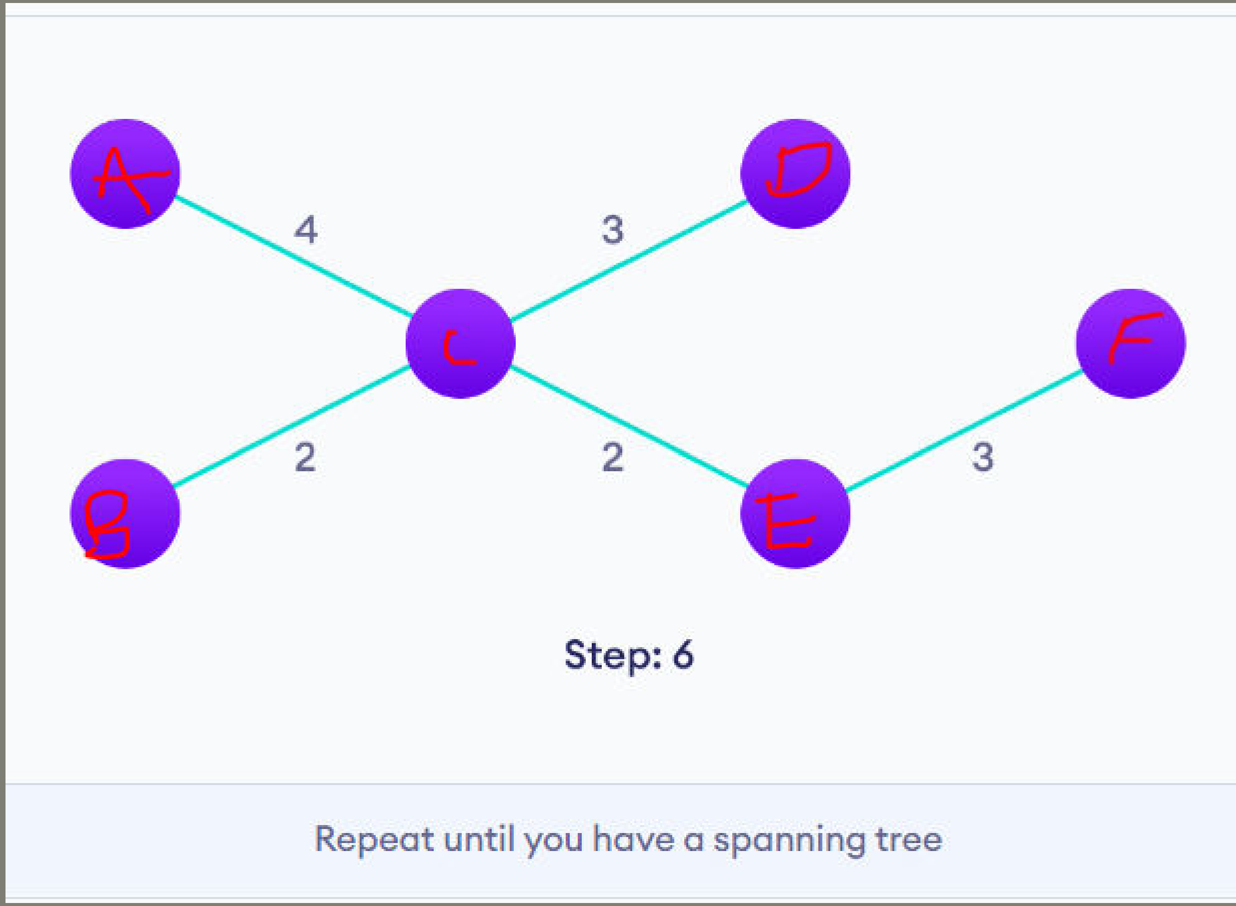
Step: 4

Choose the next shortest edge that doesn't create a cycle and add it



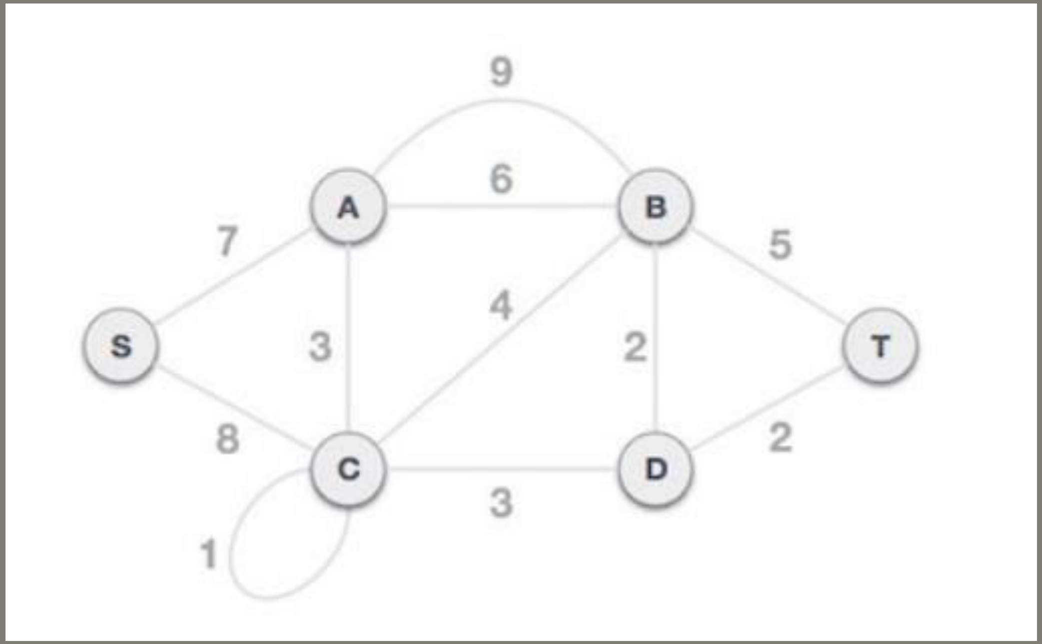
Step: 5

Choose the next shortest edge that doesn't create a cycle and add it


A
D
C
F
B
E
4
3
2
2
3
Step: 6
Repeat until you have a spanning tree

MINIMUM SPANNING TREE(MST:

Find The minimum spanning tree using Kruskal's Algorithm.

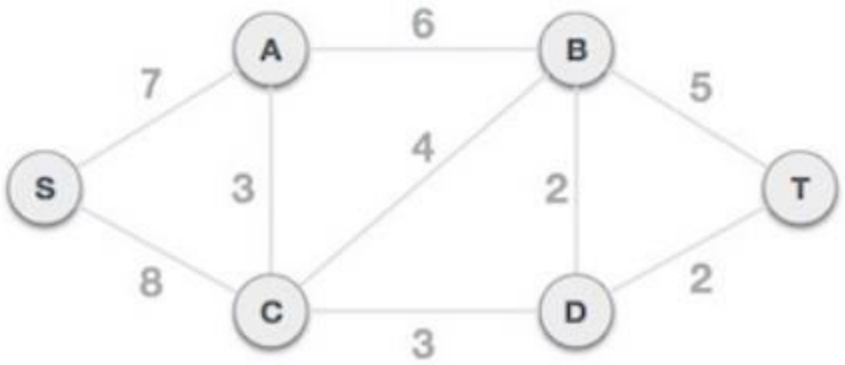


STEP: I Remove loops and parallel edges:

{B ,C}	{C ,E}	{D ,C}	{E ,F}	{D ,F}	{B ,A}	{A ,C}	{F ,C}
2	2	3	3	3	4	4	4

MINIMUM SPANNING TREE(MST:

In case of parallel edges, keep the one which has the least cost associated and remove all others.



Step 2 - Arrange all edges in their increasing order of weight

The next step is to create a set of edges and weight, and arrange them in an ascending order of weightage (cost).

B, D	D, T	A, C	C, D	C, B	B, T	A, B	S, A	S, C
2	2	3	3	4	5	6	7	8

MINIMUM SPANNING TREE(MST:

- There are two algorithms for finding Minimum Spanning Tree:
  - (a) Kruskal's Algorithm
  - (b) Prim's Algorithm

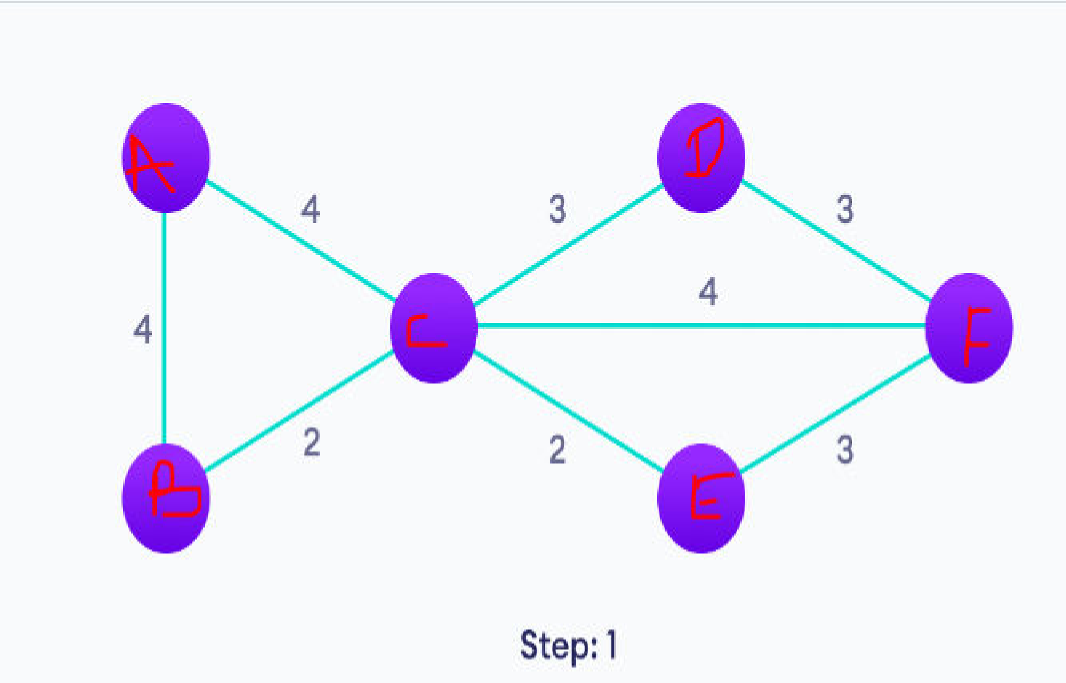
(a)PRIM's ALGORITHM:

The steps for implementing Prim's algorithm are as follows:

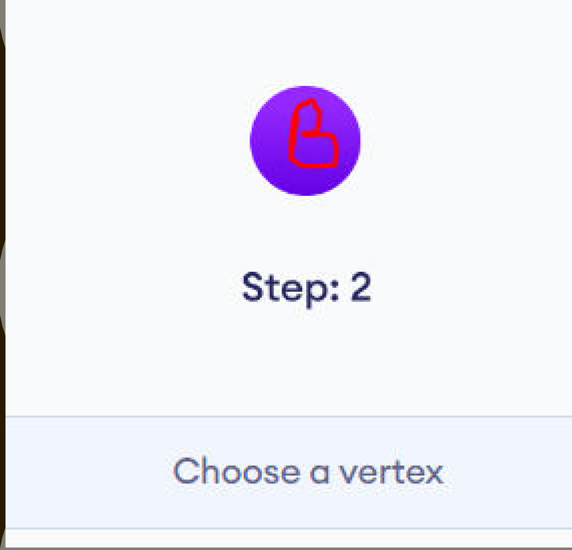
- Remove all the loops and parallel edges if present. In case of parallel edges, keep the one which has the least cost associated and remove all others.
- Choose any arbitrary vertex as a root
- Check outgoing edges and select the one with least cost and no cycle
- Repeat step(ii) until all vertices are covered.

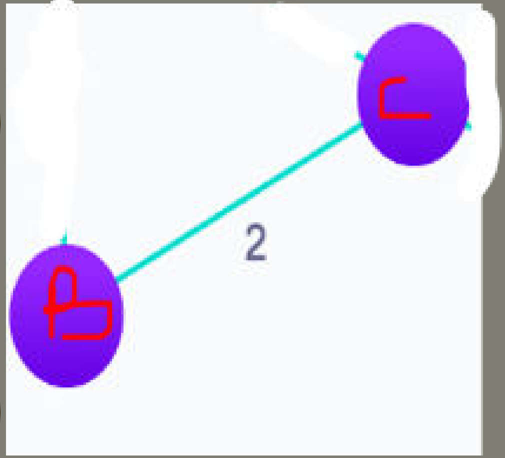
MINIMUM SPANNING TREE(MST:

Find The minimum spanning tree using Prim's Algorithm.

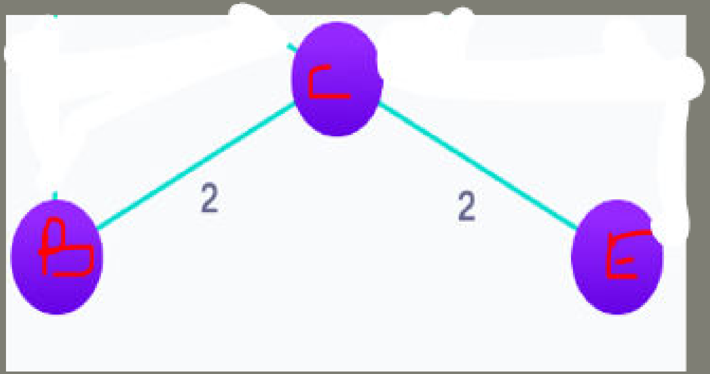


Step: I If loops and parallel edges are present remove it

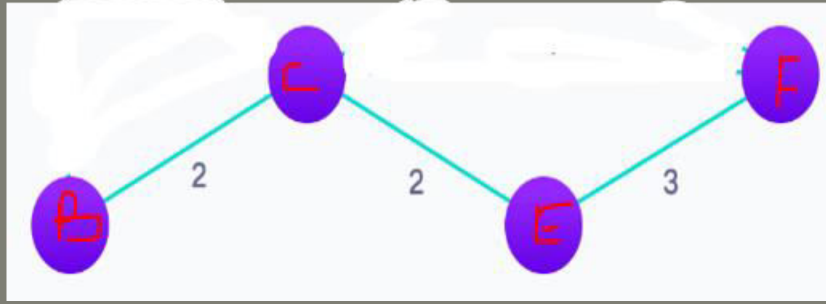




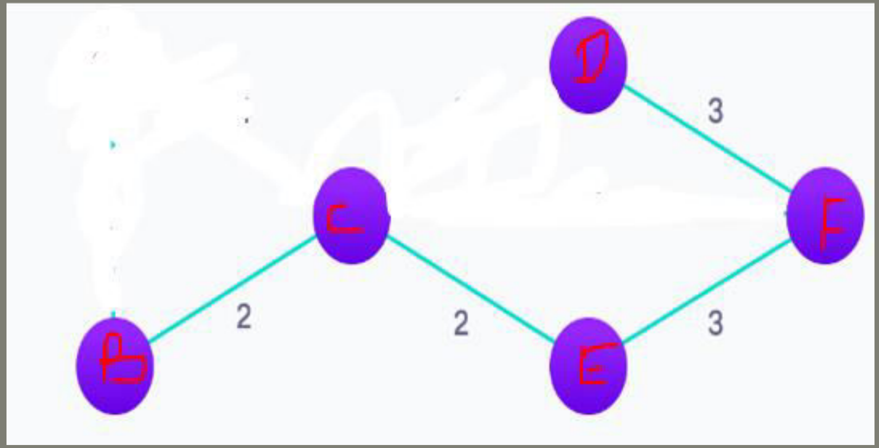
Step: 3 Choose the smallest weight connected edge



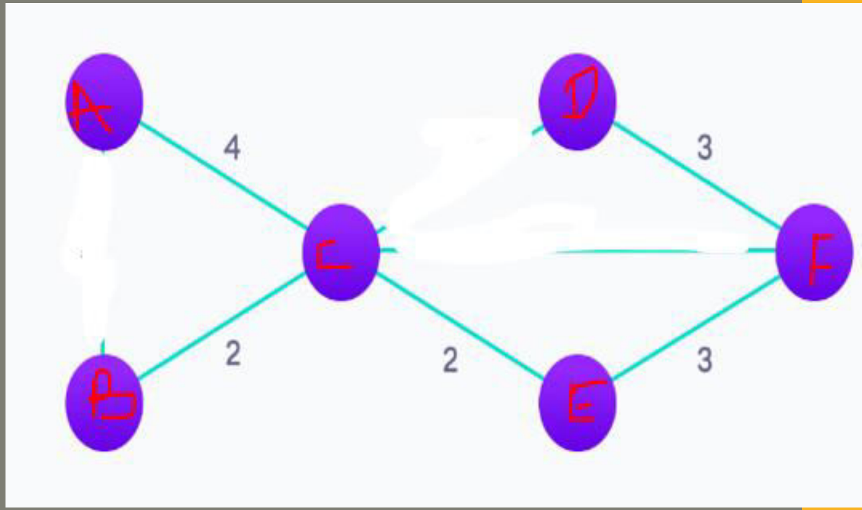
Step: 4



Step: 5



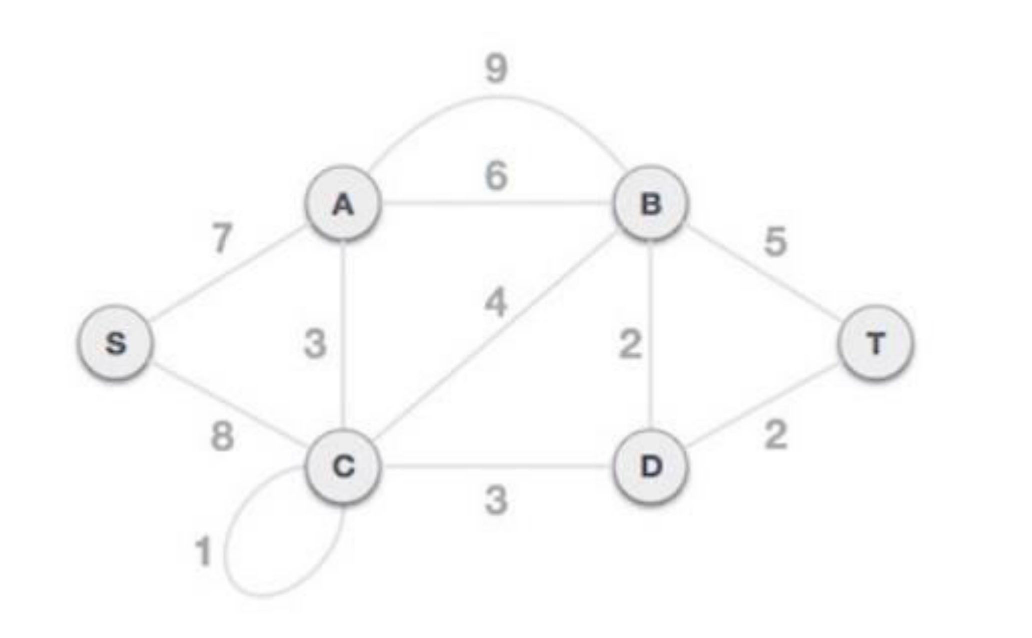
Step: 6



Step: 7
MST
Weight = 14

MINIMUM SPANNING TREE(MST:

Find The minimum spanning tree using Kruskal's Algorithm.



Question 1. *Prove using mathematical induction that for all $n \geq 1$,*

$$1 + 4 + 7 + \cdots + (3n-2) = \frac{n(3n-1)}{2}.$$

Solution.

For any integer $n \geq 1$, let P_n be the statement that

$$1 + 4 + 7 + \cdots + (3n-2) = \frac{n(3n-1)}{2}.$$

<u>Base Case.</u> The statement P_1 says that

$$1 = \frac{1(3-1)}{2},$$

which is true.

<u>Inductive Step.</u> Fix $k \geq 1$, and suppose that P_k holds, that is,

$$1 + 4 + 7 + \cdots + (3k-2) = \frac{k(3k-1)}{2}.$$

It remains to show that P_{k+1} holds, that is,

$$1 + 4 + 7 + \cdots + (3(k+1)-2) = \frac{(k+1)(3(k+1)-1)}{2}.$$

$$\begin{aligned}
1 + 4 + 7 + \cdots + (3(k+1)-2) &= 1 + 4 + 7 + \cdots + (3(k+1)-2) \\
&= 1 + 4 + 7 + \cdots + (3k+1) \\
&= 1 + 4 + 7 + \cdots + (3k-2) + (3k+1) \\
&= \frac{k(3k-1)}{2} + (3k+1) \\
&= \frac{k(3k-1) + 2(3k+1)}{2} \\
&= \frac{3k^2 - k + 6k + 2)}{2} \\
&= \frac{3k^2 + 5k + 2)}{2} \\
&= \frac{(k+1)(3k+2)}{2} \\
&= \frac{(k+1)(3(k+1)-1)}{2}.
\end{aligned}$$

Therefore P_{k+1} holds.

Thus, by the principle of mathematical induction, for all $n \geq 1$, P_n holds. $\square$

Question 2. *Use the Principle of Mathematical Induction to verify that, for n any positive integer, $6^n - 1$ is divisible by 5.*

Solution.

For any $n \geq 1$, let P_n be the statement that $6^n - 1$ is divisible by 5.

Base Case. The statement P_1 says that

$$6^1 - 1 = 6 - 1 = 5$$

is divisible by 5, which is true.

Inductive Step. Fix $k \geq 1$, and suppose that P_k holds, that is, $6^k - 1$ is divisible by 5.

It remains to show that P_{k+1} holds, that is, that $6^{k+1} - 1$ is divisible by 5.

$$\begin{aligned} 6^{k+1} - 1 &= 6(6^k) - 1 \\ &= 6(6^k - 1) - 1 + 6 \\ &= 6(6^k - 1) + 5. \end{aligned}$$

By P_k, the first term $6(6^k - 1)$ is divisible by 5, the second term is clearly divisible by 5. Therefore the left hand side is also divisible by 5. Therefore P_{k+1} holds.

Thus by the principle of mathematical induction, for all $n \geq 1$, P_n holds.

Question 3. *Verify that for all $n \geq 1$, the sum of the squares of the first $2n$ positive integers is given by the formula*

$$1^2 + 2^2 + 3^2 + \cdots + (2n)^2 = \frac{n(2n+1)(4n+1)}{3}$$

Solution.

For any integer $n \geq 1$, let P_n be the statement that

$$1^2 + 2^2 + 3^2 + \cdots + (2n)^2 = \frac{n(2n+1)(4n+1)}{3}.$$

<u>Base Case.</u> The statement P_1 says that

$$1^2 + 2^2 = \frac{(1)(2(1)+1)(4(1)+1)}{3} = \frac{3(5)}{3} = 5,$$

which is true.

<u>Inductive Step.</u> Fix $k \geq 1$, and suppose that P_k holds, that is,

$$1^2 + 2^2 + 3^2 + \cdots + (2k)^2 = \frac{k(2k+1)(4k+1)}{3}.$$

It remains to show that P_{k+1} holds, that is,

$$1^2 + 2^2 + 3^2 + \cdots + (2(k+1))^2 = \frac{(k+1)(2(k+1)+1)(4(k+1)+1)}{3}.$$

$$\begin{aligned}
1^2 + 2^2 + 3^2 + \cdots + (2(k+1))^2 &= 1^2 + 2^2 + 3^2 + \cdots + (2k+2)^2 \\
&= 1^2 + 2^2 + 3^2 + \cdots + (2k)^2 + (2k+1)^2 + (2k+2)^2 \\
&= \frac{k(2k+1)(4k+1)}{3} + (2k+1)^2 + (2k+2)^2 && \text{(by } P_k\text{)} \\
&= \frac{k(2k+1)(4k+1)}{3} + \frac{3(2k+1)^2 + 3(2k+2)^2}{3} \\
&= \frac{k(2k+1)(4k+1) + 3(2k+1)^2 + 3(2k+2)^2}{3} \\
&= \frac{k(8k^2+6k+1) + 3(4k^2+4k+1) + 3(4k^2+8k+4)}{3} \\
&= \frac{(8k^3+6k^2+k) + (12k^2+12k+3) + (12k^2+24k+12)}{3} \\
&= \frac{8k^3+30k^2+37k+15}{3}
\end{aligned}$$

On the other side of P_{k+1},

$$\frac{(k+1)(2(k+1)+1)(4(k+1)+1)}{3} = \frac{(k+1)(2k+2+1)(4k+4+1)}{3}$$

$$
\begin{aligned}
&= \frac{(k+1)(2k+3)(4k+5)}{3} \\
&= \frac{(2k^2+5k+3)(4k+5)}{3} \\
&= \frac{8k^3+30k^2+37k+15}{3}.
\end{aligned}
$$

Therefore P_{k+1} holds.

Thus, by the principle of mathematical induction, for all $n \geq 1$, P_n holds. $\square$

Question 4. *Consider the sequence of real numbers defined by the relations*

$$x_1 = 1 \text{ and } x_{n+1} = \sqrt{1 + 2x_n} \text{ for } n \geq 1.$$

Use the Principle of Mathematical Induction to show that $x_n < 4$ for all $n \geq 1$.

Solution.

For any $n \geq 1$, let P_n be the statement that $x_n < 4$.

Base Case. The statement P_1 says that $x_1 = 1 < 4$, which is true.

Inductive Step. Fix $k \geq 1$, and suppose that P_k holds, that is, $x_k < 4$.

It remains to show that P_{k+1} holds, that is, that $x_{k+1} < 4$.

$$\begin{aligned} x_{k+1} &= \sqrt{1 + 2x_k} \\ &< \sqrt{1 + 2(4)} \\ &= \sqrt{9} \\ &= 3 \\ &< 4. \end{aligned}$$

Therefore P_{k+1} holds.

Thus by the principle of mathematical induction, for all $n \geq 1$, P_n holds.

Question 5. *Show that* $n! > 3^n$ *for* $n \geq 7$.

Solution.

For any $n \geq 7$, let P_n be the statement that $n! > 3^n$.

Base Case. The statement P_7 says that $7! = 7 \times 6 \times 5 \times 4 \times 3 \times 2 \times 1 = 5040 > 3^7 = 2187$, which is true.

Inductive Step. Fix $k \geq 7$, and suppose that P_k holds, that is, $k! > 3^k$.

It remains to show that P_{k+1} holds, that is, that $(k+1)! > 3^{k+1}$.

$$\begin{aligned}
(k+1)! &= (k+1)k! \\
&> (k+1)3^k \\
&\geq (7+1)3^k \\
&= 8 \times 3^k \\
&> 3 \times 3^k \\
&= 3^{k+1}.
\end{aligned}$$

Therefore P_{k+1} holds.

Thus by the principle of mathematical induction, for all $n \geq 1$, P_n holds.

Question 6. *Let $p_0 = 1$, $p_1 = \cos\theta$ (for θ some fixed constant) and $p_{n+1} = 2p_1p_n - p_{n-1}$ for $n \geq 1$. Use an extended Principle of Mathematical Induction to prove that $p_n = \cos(n\theta)$ for $n \geq 0$.*

Solution.

For any $n \geq 0$, let P_n be the statement that $p_n = \cos(n\theta)$.

Base Cases. The statement P_0 says that $p_0 = 1 = \cos(0\theta) = 1$, which is true. The statement P_1 says that $p_1 = \cos\theta = \cos(1\theta)$, which is true.

Inductive Step. Fix $k \geq 0$, and suppose that both P_k and P_{k+1} hold, that is, $p_k = \cos(k\theta)$, and $p_{k+1} = \cos((k+1)\theta)$.

It remains to show that P_{k+2} holds, that is, that $p_{k+2} = \cos((k+2)\theta)$.

We have the following identities:

$$\cos(a+b) = \cos a\cos b - \sin a\sin b$$

$$\cos(a-b) = \cos a\cos b + \sin a\sin b$$

Therefore, using the first identity when $a = \theta$ and $b = (k+1)\theta$, we have

$$\cos(\theta + (k+1)\theta) = \cos\theta\cos(k+1)\theta - \sin\theta\sin(k+1)\theta,$$

and using the second identity when $a = (k+1)\theta$ and $b = \theta$, we have

$$\cos((k+1)\theta - \theta) = \cos(k+1)\theta\cos\theta + \sin(k+1)\theta\sin\theta.$$

Therefore,

$$\begin{aligned}
p_{k+2} &= 2p_1p_{k+1} - p_k \\
&= 2(\cos\theta)(\cos((k+1)\theta)) - \cos(k\theta) \\
&= (\cos\theta)(\cos((k+1)\theta)) + (\cos\theta)(\cos((k+1)\theta)) - \cos(k\theta) \\
&= \cos(\theta + (k+1)\theta) + \sin\theta\sin(k+1)\theta + (\cos\theta)(\cos((k+1)\theta)) - \cos(k\theta) \\
&= \cos((k+2)\theta) + \sin\theta\sin(k+1)\theta + (\cos\theta)(\cos((k+1)\theta)) - \cos(k\theta) \\
&= \cos((k+2)\theta) + \sin\theta\sin(k+1)\theta + \cos((k+1)\theta - \theta) - \sin(k+1)\theta\sin\theta - \cos(k\theta) \\
&= \cos((k+2)\theta) + \cos(k\theta) - \cos(k\theta) \\
&= \cos((k+2)\theta).
\end{aligned}$$

Therefore P_{k+2} holds.

Thus by the principle of mathematical induction, for all $n \geq 1$, P_n holds.

Question 7. *Consider the famous Fibonacci sequence* $\{x_n\}_{n=1}^{\infty}$*, defined by the relations* $x_1 = 1$*,* $x_2 = 1$*, and* $x_n = x_{n-1} + x_{n-2}$ *for* $n \geq 3$*.*

(a) Compute x_{20}*.*

(b) Use an extended Principle of Mathematical Induction in order to show that for $n \geq 1$*,*

$$x_n = \frac{1}{\sqrt{5}}\left[\left(\frac{1+\sqrt{5}}{2}\right)^n - \left(\frac{1-\sqrt{5}}{2}\right)^n\right].$$

(c) Use the result of part (b) to compute x_{20}*.*

Solution.

(a)

$$1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233, 377, 610, 987, 1597, 2584, 4181, 6765$$

(b) For any $n \geq 1$, let P_n be the statement that

$$x_n = \frac{1}{\sqrt{5}}\left[\left(\frac{1+\sqrt{5}}{2}\right)^n - \left(\frac{1-\sqrt{5}}{2}\right)^n\right].$$

Base Case. The statement P_1 says that

$$\begin{aligned}
x_1 &= \frac{1}{\sqrt{5}}\left[\left(\frac{1+\sqrt{5}}{2}\right)^1 - \left(\frac{1-\sqrt{5}}{2}\right)^1\right] \\
&= \frac{1}{\sqrt{5}}\left[\frac{1+\sqrt{5}}{2} - \frac{1-\sqrt{5}}{2}\right] \\
&= \frac{1}{\sqrt{5}}\left[\frac{1+\sqrt{5}-1+\sqrt{5}}{2}\right] \\
&= \frac{1}{\sqrt{5}}\left[\frac{2\sqrt{5}}{2}\right] \\
&= 1,
\end{aligned}$$

which is true. The statement P_2 says that

$$x_2 = \frac{1}{\sqrt{5}}\left[\left(\frac{1+\sqrt{5}}{2}\right)^2 - \left(\frac{1-\sqrt{5}}{2}\right)^2\right]$$

$$
\begin{aligned}
&= \frac{1}{\sqrt{5}}\left[\left(\frac{1+2\sqrt{5}+5}{4}\right) - \left(\frac{1-2\sqrt{5}+5}{4}\right)\right] \\
&= \frac{1}{\sqrt{5}}\left[\left(\frac{1+2\sqrt{5}+5-1+2\sqrt{5}-5}{4}\right)\right] \\
&= \frac{1}{\sqrt{5}}\left[\frac{4\sqrt{5}}{4}\right] \\
&= 1,
\end{aligned}
$$

which is again true.

Inductive Step. Fix $k \geq 1$, and suppose that P_k and P_{k+1} both hold, that is,

$$x_k = \frac{1}{\sqrt{5}}\left[\left(\frac{1+\sqrt{5}}{2}\right)^k - \left(\frac{1-\sqrt{5}}{2}\right)^k\right],$$

and

$$x_{k+1} = \frac{1}{\sqrt{5}}\left[\left(\frac{1+\sqrt{5}}{2}\right)^{k+1} - \left(\frac{1-\sqrt{5}}{2}\right)^{k+1}\right].$$

It remains to show that P_{k+2} holds, that is, that

$$x_{k+2} = \frac{1}{\sqrt{5}}\left[\left(\frac{1+\sqrt{5}}{2}\right)^{k+2} - \left(\frac{1-\sqrt{5}}{2}\right)^{k+2}\right].$$

$$
\begin{aligned}
x_{k+2} &= x_k + x_{k+1} \\
&= \frac{1}{\sqrt{5}}\left[\left(\frac{1+\sqrt{5}}{2}\right)^k - \left(\frac{1-\sqrt{5}}{2}\right)^k\right] + \frac{1}{\sqrt{5}}\left[\left(\frac{1+\sqrt{5}}{2}\right)^{k+1} - \left(\frac{1-\sqrt{5}}{2}\right)^{k+1}\right] \\
&= \frac{1}{\sqrt{5}}\left[\left(\frac{1+\sqrt{5}}{2}\right)^k + \left(\frac{1+\sqrt{5}}{2}\right)^{k+1} - \left(\frac{1-\sqrt{5}}{2}\right)^k - \left(\frac{1-\sqrt{5}}{2}\right)^{k+1}\right] \\
&= \frac{1}{\sqrt{5}}\left[\left(\frac{1+\sqrt{5}}{2}\right)^k \left(1+\frac{1+\sqrt{5}}{2}\right) - \left(\frac{1-\sqrt{5}}{2}\right)^k \left(1+\frac{1-\sqrt{5}}{2}\right)\right] \\
&= \frac{1}{\sqrt{5}}\left[\left(\frac{1+\sqrt{5}}{2}\right)^k \left(\frac{3+\sqrt{5}}{2}\right) - \left(\frac{1-\sqrt{5}}{2}\right)^k \left(\frac{3-\sqrt{5}}{2}\right)\right] \\
&= \frac{1}{\sqrt{5}}\left[\left(\frac{1+\sqrt{5}}{2}\right)^k \left(\frac{6+2\sqrt{5}}{4}\right) - \left(\frac{1-\sqrt{5}}{2}\right)^k \left(\frac{6-2\sqrt{5}}{4}\right)\right]
\end{aligned}
$$

$$= \frac{1}{\sqrt{5}}\left[\left(\frac{1+\sqrt{5}}{2}\right)^k\left(\frac{1+2\sqrt{5}+5}{4}\right)-\left(\frac{1-\sqrt{5}}{2}\right)^k\left(\frac{1-2\sqrt{5}+5}{4}\right)\right]$$

$$= \frac{1}{\sqrt{5}}\left[\left(\frac{1+\sqrt{5}}{2}\right)^k\left(\frac{1+\sqrt{5}}{2}\right)^2-\left(\frac{1-\sqrt{5}}{2}\right)^k\left(\frac{1-\sqrt{5}}{2}\right)^2\right]$$

$$= \frac{1}{\sqrt{5}}\left[\left(\frac{1+\sqrt{5}}{2}\right)^{k+2}-\left(\frac{1-\sqrt{5}}{2}\right)^{k+2}\right].$$

Therefore P_{k+2} holds.

Thus by the principle of mathematical induction, for all $n \geq 1$, P_n holds.

(c) Plugging $n = 20$ in a calculator yields the answer quickly.