A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V
65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81	82	83	84	85	86

W	X	Y	Z
87	88	89	90

a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
97	98	99	100	101	102	103	104	105	106	107	108	109	110	111

p	q	r	s	t	u	v	w	x	y	z
112	113	114	115	116	117	118	119	120	121	122

storing all videos in YouTube is a database.

Database Management systems

Data = fact that can be recorded

Information = processed data or meaningful data.

Database = collection of similar / related data.

Eg. of data:- Number, images, text, videos, graph, audio.

Hexa decimal series :- 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F.

Octadecimal series :- 0, 1, 2, 3, 4, 5, 6, 7.

Data representation :-

1. Number

$(2134)_{10} \rightarrow$

2	2134
2	1067 - 0
2	533 - 1
2	266 - 0
2	133 - 0
2	66 - 1
2	33 - 0
2	16 - 1
2	8 - 0
2	4 - 0
2	2 - 0
	1 - 0

$(100001010110)_2$

Q Find the binary equivalent of 13115.

2	13115
2	6557 - 1
2	3278 - 1
2	1639 - 0
2	819 - 1
2	409 - 1
2	204 - 1
2	102 - 1
2	51 - 0
2	25 - 1
2	12 - 1
2	6 - 0
2	3 - 0
2	2 - 1
	1

11 0011 0111 1011

Binary to decimal

1010 → ?

1010 = 10

2^3 2^2 2^1 2^0

$(2^3 \times 1) + (2^2 \times 0) + (2^1 \times 1) + (2^0 \times 0)$

$8 + 0 + 2 + 0 = 10$

ASCII → 8 bit encoding scheme

ASCII values of A - 65, B - 66, ..., Z - 90 | a = 97, b = 98, ..., z = 122

Convert the text "Hello" to binary using 8 bit ASCII encoding ~~spreadsheet~~.

Ans:- Step i:- find the ASCII value of each letter.

H - 72
e - 101
l - 108
l - 108
o - 111

H →

2 | 72	
2 | 36 - 0	
2 | 18 - 0	
2 | 9 - 0	
2 | 4 - 1	
2 | 2 - 0	
1 - 0	

01001000

e →

2 | 101	
2 | 50 - 1	
2 | 25 - 0	
2 | 12 - 1	
2 | 6 - 0	
2 | 3 - 0	
1 - 1	

01100101

l →

2 | 108	
2 | 54 - 0	
2 | 27 - 0	
2 | 13 - 1	
2 | 6 - 1	
2 | 3 - 0	
1 - 1	

01101100

0 → 2 | 111
2 | 55-1
2 | 27-1
2 | 13-1
2 | 6-1
2 | 3-0
2 | 1-1

01101111

For Hello
11111
88888 → 40 bits

For Hello → Binary system is;

01001000 01100101 01101100 01101100 01101111

---

1. Data inconsistancy
2. Data redundancy
3. Multiple user view (not present)
4. Security (multi level user view) (not present).

Types of database :-

Relational, Multimedia, Graph, optical oriented.

eg. neo4j

Database Management system (DBMS) :-

Set of programs that manages Database

Creating, updating, Retrieving, Deleting.

DBMS + DB = DBS = Database system.

Architecture of DBS:

User → Application software

↓ ← SQL

DBMS

↓

Database

Structured Query Language (SQL)

SQL is only input to DBMS.

Data Model:-

DBMS understands only SQL Language.

column headers → Attributes

Rows → Tuple

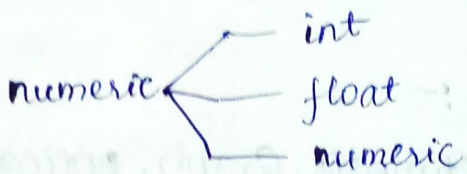
Structure of the relation → schema

Degree of a relation → No. of attributes

cardinality of a solution → No. of Tuple

1. Create

CREATE TABLE tablename (attribute1, datatype 1, attribute2 datatype 2);



Text / char
varchar

Eg:- Create table Book (BID int, Title varchar(5), Author varchar(10), price numeric(4,2));

BOOK

BID	Title	Author	price
1.	ABC	Tom	100
2.	xyz.	Harry	172.53

Eg: For the following schema create a relation book using SQL query book.

STUDENTS

ID	Roll No.	Name	Marks
1	A120	abc	92.50
2	A320	xyz	32.73
3	B260	lmn	53.21

Ans:
Create table student
(ID int, Rollno. varchar(5), Name. varchar(20),
Marks. numeric (5,2));

SELECT QUERY

Q.

book

ID	Title*	Author	Price
1	abc	Tom	100
2	xyz	Bob	200
3	lmn	Tom	150
4	pqr	Tom	50

SELECT Title* FROM book# ⇒

Title
abc
xyz
lmn
pqr

SELECT Title FROM book WHERE Author = 'Tom';

Q. Retrieve the title and price of those books authored by Tom.

SELECT Title, price FROM book WHERE Author = 'Tom';

Title	price
abc	100
lmn	150
pqr	50

Q. Retrieve the Title of all book written by Tom and price less than 150.

Ans:- SELECT Title FROM book WHERE Author = 'Tom' AND Price < 150;

Q. Retrieve all details of book authored by Tom and price greater than 100.

Ans:- SELECT * FROM book WHERE Author = 'Tom' AND Price > 100;

Q. Find out the number of Tuples inside the book.

Ans:- SELECT count(*) FROM book;

Q. Retrieve the max. price of a book authored by Tom.

Ans:- SELECT max(price) FROM book WHERE Author = 'Tom';

Construct SQL queries for the following questions using table below

hotel

ID	Name	City	Tariff
1	abc	TVM	1000
2	def	Kollam	500
3	abc	Kochi	2000
4	xyz	TVM	1500

Create a table hotel for the above schema

Retrieve all details of hotels.

Retrieve the names of hotels in Trivandram city.

Retrieve the hotels in Trivandram city with Tarriff greater than 500.

Retrieve the minimum tariff for the hotel abc.

DROP TABLE hotel ; → table will be removed

UPDATE query

UPDATE tablename SET attr1 = value1, att 2 = value2, ...
WHERE condition

Eg:-

book

ID	Title	Author	price
1	abc	tom	100
2	xyz	bob	200
3	lmn	tom	150
4	pqr	tom	50

considering the above schema,

i. UPDATE the price of the book titled lmn to 500.

Ans:- UPDATE book SET price = 500 WHERE Title = 'lmn';

ii. UPDATE the author of the book as Charles and title as 'xyz' and price as 300 for the book with ID=1.

Ans:- UPDATE book SET Author = 'Charles', Title = 'xyz', price = 300 WHERE ID = 1 ;

DELETE query:- → deleting the rows

DELETE from table name WHERE condition.

3. Delete all the details of the books with price less than 200

Ans:- DELETE from book WHERE price < 200;

4. Delete the details of books authored by tom and price greater than 100.

Ans:- DELETE from book WHERE Author = 'tom' and price > 100;

INSERT query:-

INSERT INTO table name values (value 1, value 2, value 3, ...);

Q. Insert a new tuple with ID. 5, title - wxy, Author - rita, price = 350 into the table book.

Ans:- INSERT INTO book VALUES (5, 'wxy', 'rita', 350);

Q. Insert two or more values:-

Ans. INSERT INTO book VALUES (1, 'abc', 'tom', 100),
(2, 'xyz', 'bob', 200),
(3, 'lmn', 'tom', 150),
(4, 'wxy', 'tom', 50);

Consider the below table Student and write SQL queries for the following questions.

Student

ID	Name	Mark
1	ani	20
2.	Charles	25
3.	bob	15
4.	tom	17
5.	grace	22

1. Create the table Student with above schema
2. Insert the data as given in the table.
3. Update the mask of tom as 23.
4. Update the name of student having ID-3 as bobby.
5. Update the name and mask of student with ID-1 as Alex and 23 respectively.
6. Delete the details of those students whose scored masks less than 20.

1. create table Student (ID numeric(5), Name varchar(50), Mark numeric(5));

2. INSERT INTO Student values (1, 'ani', 20),
(2, 'Charles', 25),
(3, 'bob', 15),
(4, 'tom', 17),
(5, 'grace', 22);

3. UPDATE Student SET Mark = 23 where Name != 'tom';

4. UPDATE Student SET Name = 'bobby' where ID = 3;

5. UPDATE Student SET Name = 'Alex' and Mark = 23
where ID = 1;

6. Delete from Student where Mark < 20;

Note:
=>
CREATE table Student (att 1 dt 1 primary key, att 2 dt

Data Defination	Data manipulation	Primary key
DDL lang	DML Lang	
create	INSERT	
DROP	SELECT	
~~Alter~~	COPY	
	UPDATE	
	DELETE	

IMPORT

→ copy tablename from 'path' csv header;

EXPORT

→ copy tablename to 'path' csv header;

Normalization

Anamoly - a problem creator

→ insertion
→ deletion
→ updation

Row redundancy can be removed by setting primary key.

By splitting, we can reduce redundancy.

Foreign key

Eg:-

University

Rollno	Name	Age	BID	Branch	HOD_name	HOD_Ph
1	A	19	101	CSE	XYZ	123
2	B	20	102	ECE	ABC	567
3	C	20	101	CSE	XYZ	123
4	D	19	101	CSE	XYZ	123

Student

Rollno	Name	Age
1	A	19
2	B	20
3	C	20
4	D	19

BID	Branch	HOD_name	HOD_Ph
101	CSE	XYZ	123
102	ECE	ABC	567

Rollno	BID
1	101
2	102
3	101
4	101

Split the following into multiple tables so as to reduce redundancies.

Registration

S_ID	Name	CID	Course	F_ID	F_Name	Salary
1	abc	c_1	CTPS	F1	Bob	70,000
2	def	c_2	CSE	F2	Tom	80,000
3	ghi	~~C1~~	CTPS	~~F1~~	Bob	70,000
4	jkl	~~C1~~	CTPS	~~F1~~	Bob	70,000

pri

Student

SID	Name
1	abc
2	def
3	ghi
4	jkl

primary key

~~Faculty~~

FID	CID	course
F1	C1	CTPS
F2	C2	CSE

primary key

Faculty

F_ID	F_name	Salary
F1	Bob	70,000
F2	Tom	80,000

SID.	CID	FID
1	C1	F1
2	C2	F2
3	C1	~~F1~~
4	C1	~~F1~~

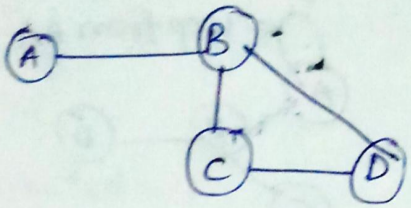
CQL - Cypher Query Language

Graph - a data structure ~~is~~ contain two components

$G = (V, E)$

V → vertex (node)

E → Edge



$V = \{A, B, C, D\}$

$E = \{AB, BC, CD, BD\}$

Types of edges:-

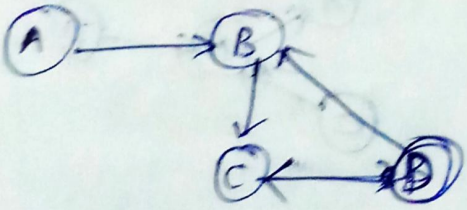
→ undirected edges (Bidirectional)

→ directed edges

→ weighted edges

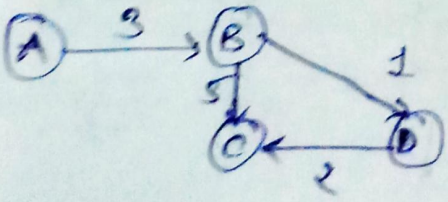
Directed edges;

Path AC = {AB, BC}



weighted edges:-

weighted edges can be directed or undirected

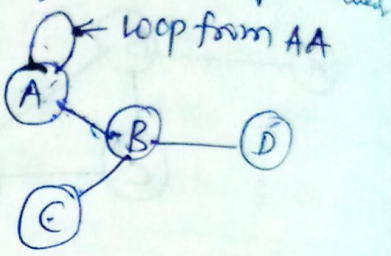


Graph representation:-

1. Adjacency matrix
2. Adjacency list

Undirected:-

Degree is the number of associated edges with at particular node.



Degree of A = 2
B = 3
C = 1
D = 1

Directed:-

INDEGREE is degree that counts for into the particular node.

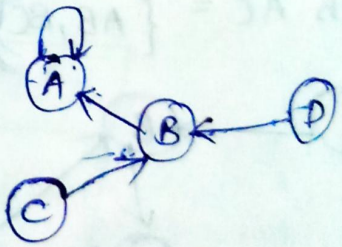
OUTDEGREE is degree for out of the particular node.

INDEGREE of

A = 2
B = 2
C = 0
D = 0

OUTDEGREE of

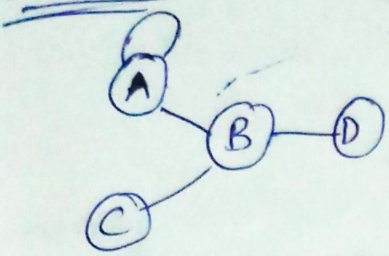
A = 1
B = 1
C = 1
D = 1



1. Adjacency matrix

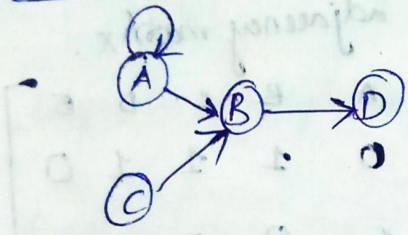
is a square matrix of order n where 'n' is no. of vertices

undirected:



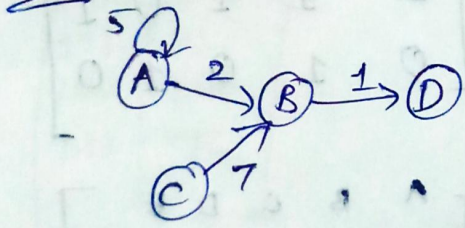
	A	B	C	D
A	1	1	0	0
B	1	0	1	1
C	0	1	0	0
D	0	1	0	0

Directed:-



	A	B	C	D
A	1	1	0	0
B	0	0	0	1
C	0	1	0	0
D	0	0	0	0

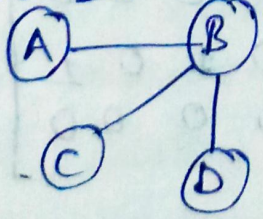
weighted :-

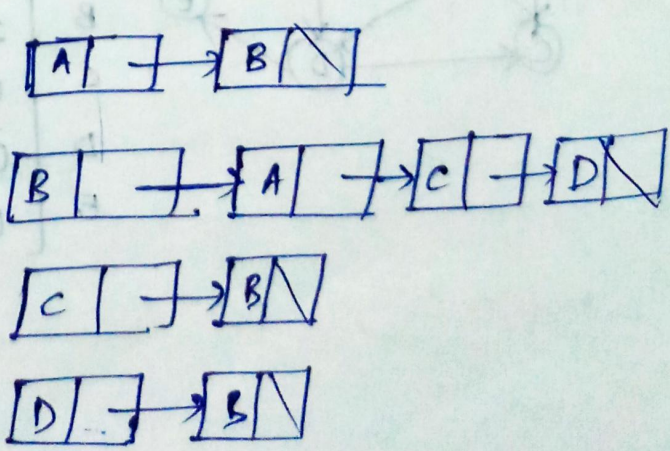


	A	B	C	D
A	5	2	0	0
B	0	0	0	1
C	0	7	0	0
D	0	0	0	0

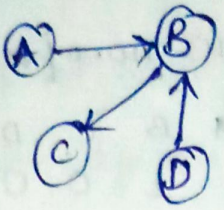
2. Adjacent list

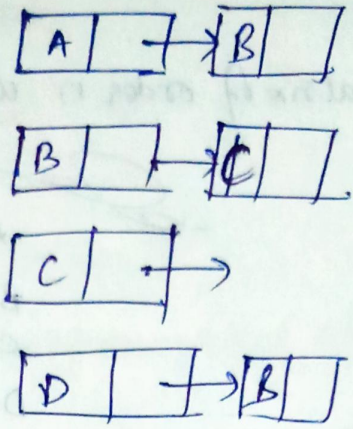
undirected :-





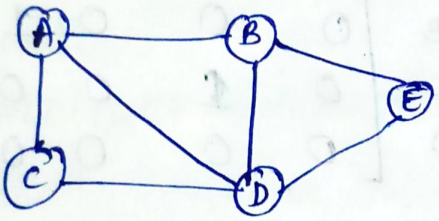
directed:





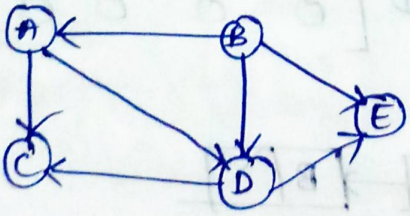
Represent the following graphs using adjacency matrix.

Q
1.



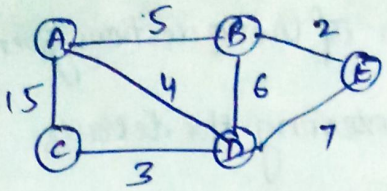
	A	B	C	D	E
A	0	1	1	1	0
B	1	0	0	1	1
C	1	0	0	1	0
D	1	1	1	0	1
E	0	1	0	1	0

2.



	A	B	C	D	E
A	0	0	1	1	0
B	1	0	0	1	1
C	0	0	0	0	0
D	0	0	1	0	1
E	0	0	0	0	0

3.



	A	B	C	D	E
A	0	5	15	4	0
B	5	0	0	6	2
C	15	0	0	3	0
D	4	6	3	0	7
E	0	2	0	7	0

neo4j

Neophorje is a ~~software~~, DBMS storing draft data

CQL → Cypher Query Language

Edges in a graph are not mandatory

Property values, tuple values are present in a node.

Student

Roll	Name
123	abc
260	bcd
312	xyz

Student

Roll = 123
Name = abc
id = 0

Student

Roll = 260
name xyz
id = 1

Student

Roll 312
name xyz
id = 2

id = identifier

Node :-

i. creating

ii. retrieving or selecting

iii. deleting.

iv. updating

Write the CQL queries for each of the questions given below

1. Create the nodes required for entering the details shown in the following table.

Course

Roll	Name	Class	Marks
1	Abi	CSE	20
2	Ani	ECE	25
3	Ami	CSE	30
4	Balu	ECE	20

Node :-

1. creating - CREATE
2. Retrieving or selecting - MATCH & RETURN
3. Deleting - MATCH & DELETE
4. Updating - MATCH & SET

create (n: Course) RETURN n;

CREATE (n: Course {Roll: 1, Name: "Abi", Class: "CSE", Marks: 20}) Return

2. List all the nodes label course.

Match (n: Course) Return n;

3. Display all nodes for CSE students.

Ans:- Match (n: Course{class: "CSE"}) Return n;

4. List the nodes relating to Abi and Balu.

Ans:- Match (n: Course {Name: "Abi"

5. List the roll no. and name for ECE students.

Ans:- Match (n: Course {class: "ECE"}) Return n.Roll, n.Name;

Roll	Name
2	Ani
4	Balu

or

Match (n: Course) where n.Class = "ECE" Return n.Roll, n.Name;

5. List the details of all students who scored masks above 20.

Ans:- Match (n:course) where n.Marks > 20 Return n;

6. Delete the node having name Ami

Ans:- Match (n: Course{Name: "Ami"}) Delete n;

7. Delete the nodes with class CSE and Marks 30

Ans:- Match(n: Course{class: "CSE", Marks: 30})
Delete n;

8. Update the name of roll no 3 to Adli.

Ans:- ~~Match(n: Course {Name: 'Adi'})~~

Match(n: course {Roll: 3})

set n.Name = "Adli", Return n;

9. Update the class and marks of Abi to EEE and 25 respectively.

Ans:- Match(n: Course {~~Name~~ : "Abi"})

Set n. Class = "EEE" ~~and~~, Marks = 25;

Consider the below table and write the CQL queries for the following questions.

1. Create the nodes corresponding to the details shown in table
2. List all nodes with label book.
3. List all books with title abc.
4. List all available books.
5. List the price and title for all books authored by Imn.
6. Delete the details of all unavailable books.
7. Delete the details of all books with price > 300.
8. Update the availability of the book with Id 3 as True

9. Update the Authors Name and price of the book titled def to xyz and 1000 respectively.

book

Id	Title	Author	Available	Price
1.	abc	xyz	false	200
2.	a12	lmn	True	150
3.	bcd	x12	false	1500
4.	def	yz	True	200

1. Create (n: book { Id : 1, Title : "abc", Author: "xyz", Available: false, Price = 200 }) Return n;

2. Match(n: book) Return n;

3. Match (n: book { Title : "abc" }) Return n;

4. Match (n: book { Available : True }) Return n;

5. Match (n: book { Author = "lmn" }) Return n.price, n.title;

6. Match (n: book { Available : "false" }) Delete n;

7. Match (n: book) where n.price > 300 Delete n;

8. Match (n: book { Id : 3 }) set n.Availability = "True";

9. Match (n: book { Title : "def" }) set n.Author = "xyz", n.Price = 1000;

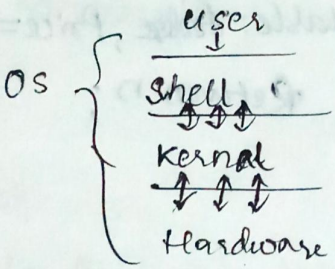
Operating System

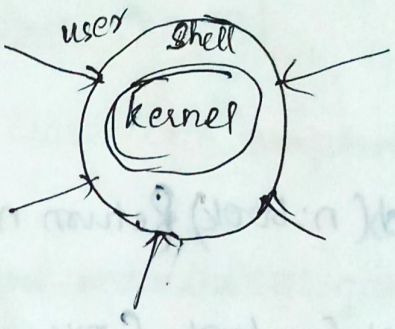
Software is of two types:

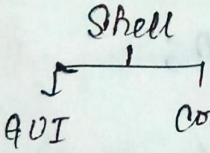
1. Application software
2. System software → utility, OS

Operating System → shell, kernel (core) (heart)

Without OS, hardware will not work







ls - listing

man _ manual

GUI - windows manager

Functions of OS

→ user interaction ⟷ shell

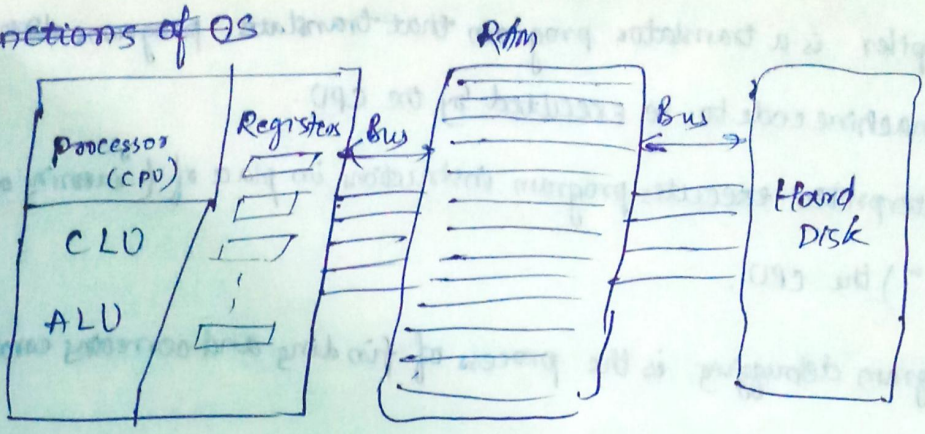
→ file management

→ Device management → device drivers → kernal

→ Memory management

→ process management

Functions of OS



ROM

PROM → Programmable ROM

EPROM → Erasable programmable ROM

EEPROM → Electrically Erasable ROM

Bios is software in ROM.

POST - Power On self Test

RAM
OS

ROM
BIOS
Boot Loader

An operating system is software that has the job of managing and interacting with the hardware resources of a computer.

Because an operating system is intrinsic to the operation a computer, it is referred to as system software.

storage devices
↑↓
Input Devices → processing devices → output devices
↓↑
communication devices.

An OS acts as the "middle man" between the hardware and executing application program.

A compiler is a translator program that translates programs directly into machine code to be executed by the CPU.

An interpreter executes program instructions in place of ("running on top of") the CPU.

→ Program debugging is the process of finding and correcting errors ("bugs") in a computer program.

Semantic errors - logic errors in program.

A process is a program in execution.

1. Create a process

All properties of process are stored in PCB. Process Control Block

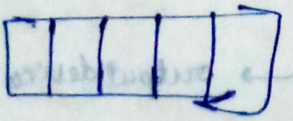
↳ like PID → processor ID

state

Process states;

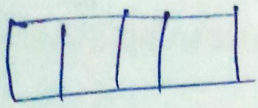
1. New → whenever a process is created it is new state

Job Queue:-



Contains those processes

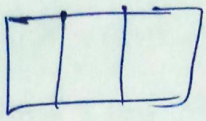
2. Ready – when a process is ready to be executed.



Ready queue stores the processes of ready

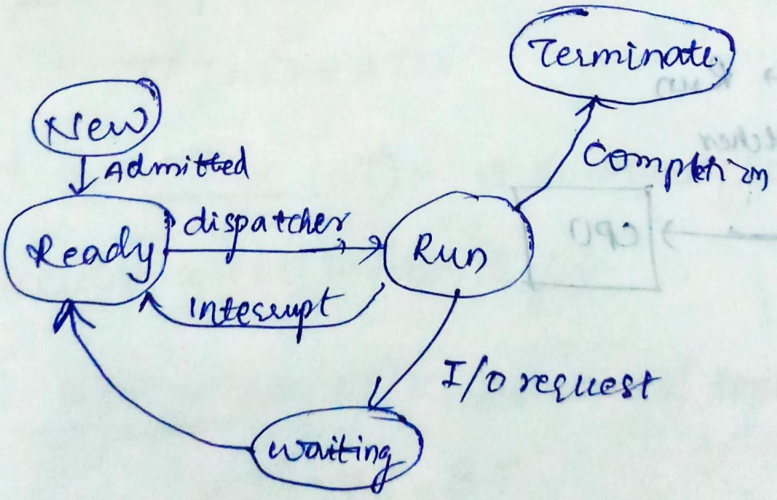
3. Running state – when a process is currently executing

4. Waiting – generally waiting for IO devices
Input from users



5. Terminate – process completed.

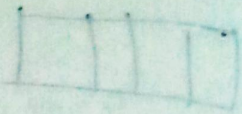
Process state Transition Diagram:-



Scheduling algorithm is handled by Dispatcher software.

New → Ready

→ Terminate



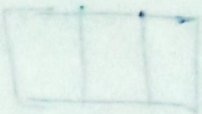
Ready → Run

→ Terminate

Run → Ready

Terminate

waiting



~~Multiprogramming~~

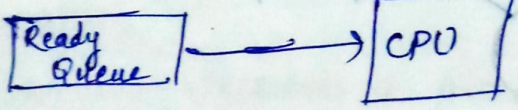
~~Time sharing – a technique which takes~~

Premptive

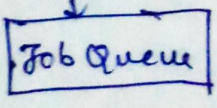
Non-Premptive

Ready → Run

Dispatcher



new process



Scheduling Algorithms :-

1. FCFS (First come First serve)
2. SJF (Shortest Job First)
3. Round Robin

1. Arrival Time (AT) :- Time of arrival of process entering the ready state queue.

2. Completion Time (CT) :- The time at which the process completes i.e., enters terminate state

3. Turn Around Time (TAT) :- It is actually the time at which process reaches ready queue and leaves ready queue.

$$TAT = CT - AT$$

4. Burst Time (BT) :- A duration of time a process requires a CPU to finish its job.

5. Waiting Time (WT) :- Duration of time the process waits

$$WT = TAT - BT$$

$$WT = CT - AT - BT$$

1. FCFS

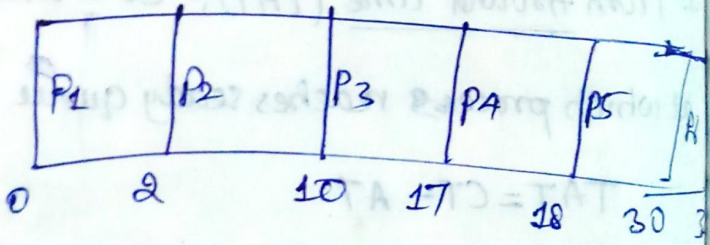
The process ~~of which arr~~ with least arrival time is given higher priority

First Come First Serve ; Criterial = AT
Mode = Non-premptive

Q. Draw the Gantt chart and calculate the average Turn Around Time and Average waiting time for the following processes using FCFS scheduling.

PID	AT	BT
P1	0	2
P2	1	8
P3	2	7
P4	3	1
P5	4	12
P6	5	4

Gantt Chart



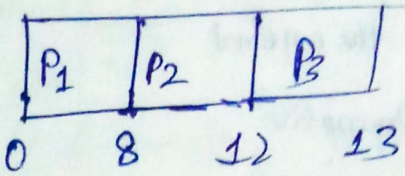
WT TAT-BT	PID	CT	TAT (CT-AT)
0	P1	2	2
1	P2	10	9
8	P3	17	15
14	P4	18	15
14	P5	30	26
25	P6	32	29

Average TAT = $\frac{96}{6} = 16$

Average WT = $\frac{62}{6} = 10.33$

Q.

PID	AT	BT
P_1	0.0	8
P_2	0.4	4
P_3	1.0	1



PID	CT	TAT (CT-AT)	WT (TAT-BT)
P_1	8	8	0
P_2	12	11.6	7.6
P_3	13	12	11

Avg TAT = $\frac{31.6}{3} = 10.5$

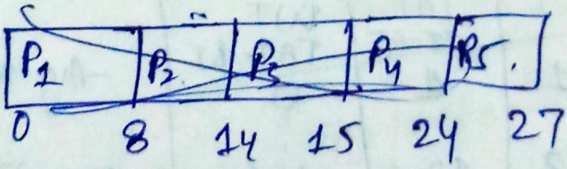
Avg WT = $\frac{18.6}{3} = 6.2$

Q.

PID	BT	AT	CT	TAT (CT-AT)	WT (TAT-BT)
P_1	8	0	8	8	0
P_2	6	2	23	21	15
P_3	1	2	24	22	21
P_4	9	1	17	16	7
P_5	3	3	27	24	21

Avg TAT = $\frac{91}{6}$ = 15.1

Avg WT = $\frac{64}{6}$ = 10.6



SJF Shortest Job first

Burst time is the criteria

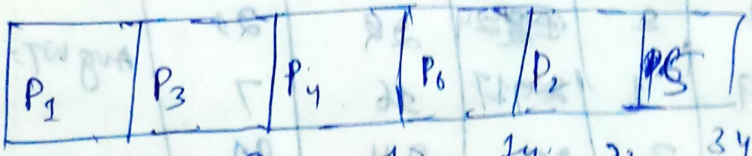
Mode - Non-Premptive

Q Draw Gantt chart and calculate TAT and WT using SJF

PID	AT	BT	CT	TAT CT-AT	WT TAT-BT
P_1	0	2	2	2	0
P_2	1	8	22	21	13
P_3	2	7	9	7	0
P_4	3	1	10	7	6
P_5	4	12	34	30	18
P_6	5	4	14	9	5

Avg. TAT = $\frac{76}{6} = 12.6$

Avg. WT = $\frac{42}{6} = 7$

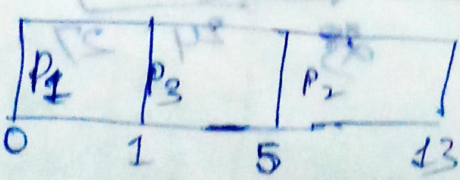


Q.

PID	AT	BT	CT	TAT CT-AT	WT TAT-BT
P_1	0.0	1	1	1	0
P2	0.4	8	13	12.6	4.6
P3	1.0	4	5	4	0

Avg. TAT = $\frac{17.6}{3}$

Avg. WT = $\frac{4.6}{3}$

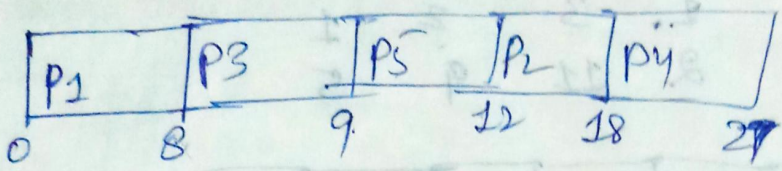


Q.

PID	BT	AT	CT	TAT CT-AT	WT TAT-BT
1	8	0	8	8	0
2	6	2	18	16	10
3	1	2	9	7	6
4	9	1	27	26	17
5	3	3	12	9	6

Avg. WT = $\frac{39}{5}$ = 7.8

Avg. TAT = $\frac{56}{5}$ = 11.2



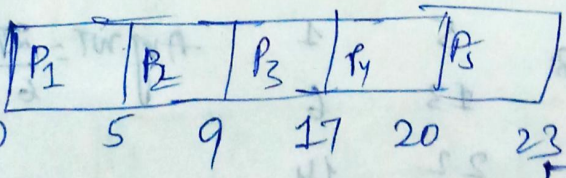
Q.

PID	AT	BT	CT	TAT CT-AT	WT TAT-BT
P1	0	5	5	5	0
P2	1	4	9	8	4
P3	2	8	19	15	7
P4	3	3	20	17	14
P5	4	3	23	19	16

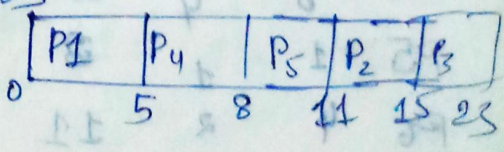
Avg. TAT = $\frac{64}{5}$ = 12.8

Avg. WT = $\frac{41}{5}$ = 8.2

Using FCFS:-



Using SJF



PID	AT	BT	CT	TAT	WT
P1	0	5	5	5	0
P2	1	4	15	14	10
P3	2	8	23	21	13
P4	3	3	8	5	2
P5	4	3	11	7	3

Avg. TAT = $\frac{52}{5}$ = 10.4

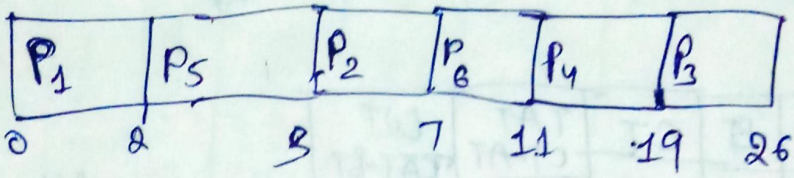
Avg. WT = $\frac{28}{5}$ = 5.6

Q.

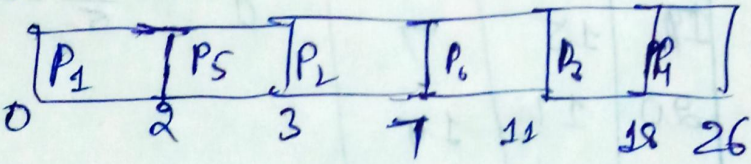
PID	BT	AT	CT	TAT	WT
By FCFS P1	2	0	2	2	0
P2	4	2	7	5	1
P3	7	5	26	21	14
P4	8	4	19	15	7
P5	1	1	3	2	1
P6	4	2	11	9	5

Avg TAT = $\frac{54}{6}$ = 9

Avg WT = $\frac{28}{6}$ = 4.6



By SJF



PID	BT	AT	CT	TAT	WT
P1	2	0	2	2	0
P2	4	2	7	5	1
P3	7	5	18	13	6
P4	8	4	26	22	14
P5	1	1	3	2	1
P6	4	2	11	9	5

Avg. TAT = $\frac{53}{6}$ = 5.7

Avg. WT = $\frac{27}{6}$ = 4.5

Round Robin

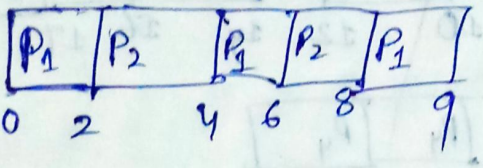
Criteria : AT

Mode : Pre-emptive

For the following processes perform Round Robin scheduling and draw the Gantt chart.

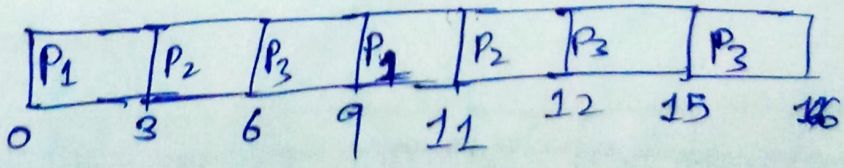
with the time slice t = 2 milliseconds

	AT	BT
P_1	0	5
P_2	1	4



t = 3 milliseconds

	AT	BT	
P1,	0	5	②
P2	1	4	①
P3	2	7	④



	AT	BT	CT
P_1	0	8	23
P_2	1	6	21
P_3	2	2	6
P_4	3	9	28
P_5	4	3	19

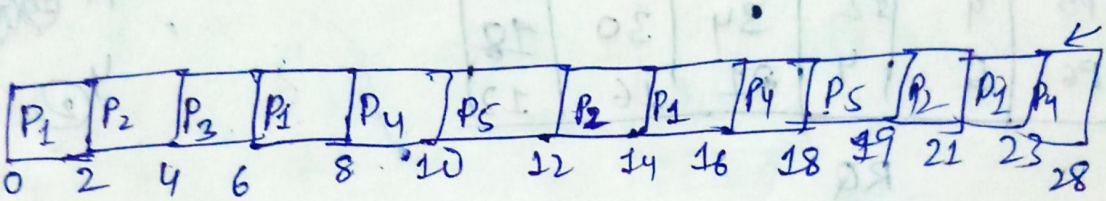
t = 2 milliseconds

RQ

RQ

P_1, P_2

P_1
P_4

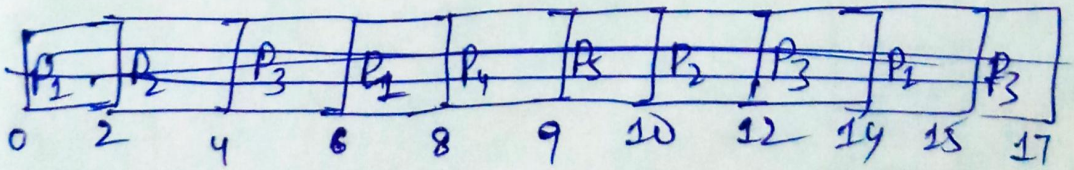


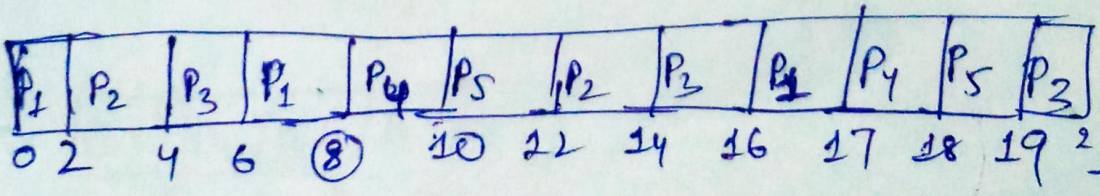
PID	AT	BT	CT
P1	0	5	17
P2	1	4	14
P3	2	8	23
P4	3	3	18
P5	4	3	19

RQ

P_1, P_2, P_3, P_1, P_4, P_5, P_2, P_3, P_1, P_4, P_5, P_3

TS = 2 milliseconds

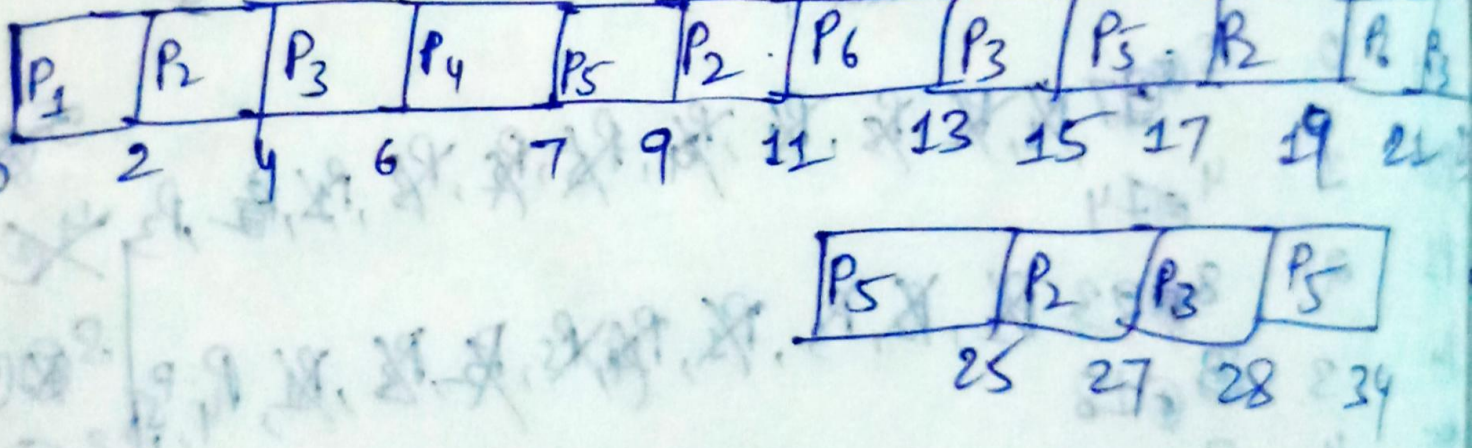




TS: 2 milliseconds

PID	AT	BT	CT	TAT	WT
P1	0	2	2	2	0
P2	1	8	27	26	18
P3	2	7	28	26	19
P4	3	1	7	4	3
P5	4	12	34	30	18
P6	5	4	21	16	12

RQ



SRTF (Shortest Remaining Time first)

Criteria : Burst Time

Mode : Pre-emptive

Schedule the following process using SRTF

	AT	BT
P1	0	8 7
P2	1	6 5
P3	2	2 x
P4	3	9
P5	4	3 x

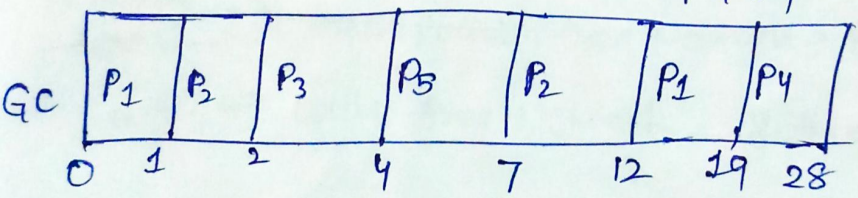
t=0 | P1 = 8

t=1 | P2 = 6 P1 = 7

t=1 | P1 = 7

t2 | P1 = 7, P3 = 2 P2 = 5

t=3 | P1 = 7, P2 = 5 P3 = 1, P4 = 9



	AT	BT
P1	0	5 4
P2	1	4 3
P3	2	8
P4	3	3
P5	4	3

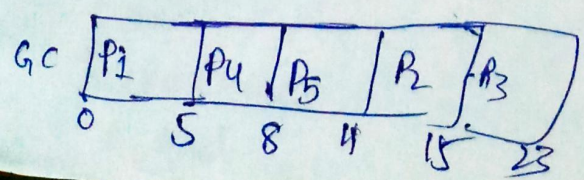
t=0 | P1 = 5

t=1 | P2 = 4 P1 = 4

P1 = 4

t=2 | P1 = 4, P3 = 8 P2 = 3

t=3 | P1 = 4, P2 = 3, P4 = 3 P3 = 7



Computer Networks :-

IP address :- Logical address

MAC address - NIC - Physical address

Classification of Networks :-

i. → Based on the geographical area

ii. → Based on mode of operation

iii. → Based on arrangement (topology)

i. PAN – Personal Area Network → Hotspot, Bluetooth.

LAN → Local Area Network → Wifi (wider than PAN)

MAN → Metro Politian Area Network → Between two building

WAN → Wide Area Network → Internet

ii. Mode of operation.

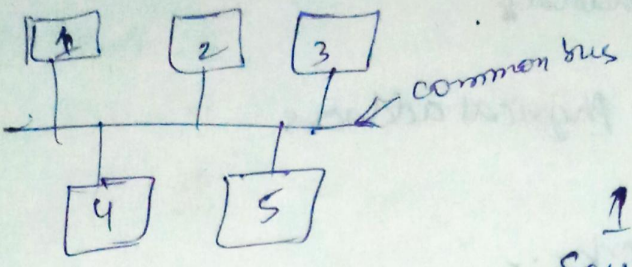
Closed Networks → Registered persons → ACMS, AHMS.

open Networks → Open to all → Eg:- Internet

iii. Based on Arrangement :-

bus
ring
star
heirarchial
hybrid

Bus is followed LAN.



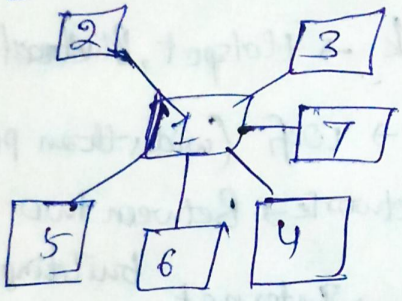
1 → 3

Source address Destination address

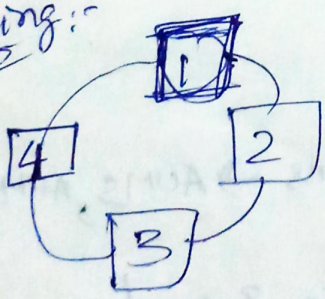
source address

Destination address

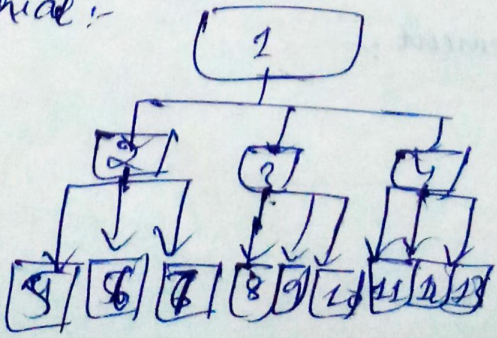
Star:



Ring:-



Heirarchial:-



Hybrid :-

Hybrid is a connection of all Bus, Star, Ring, Hierarchial etc.

~~Multiple Access Protocol~~

IPV4 → 32 bit

dotted decimal

—.—.—.—

Maximum value of IPV4

255.255.255.255

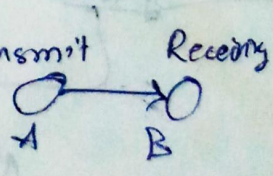
IPV6 → 128 bit

ping –

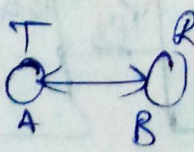
Default gateway is same for all same networked computers.

Multiple Access protocol :-

Simplex comm → only one way of communication.



Duplex comm –



Multiple access protocol

1. CSMA/CD - protocol for multiple media accessing

random access

only for BUS LAN Topology.

Ethernet LAN Media Access Protocol

an IEEE standard 802.3 (Ethernet)

802.3

CSMA/CD

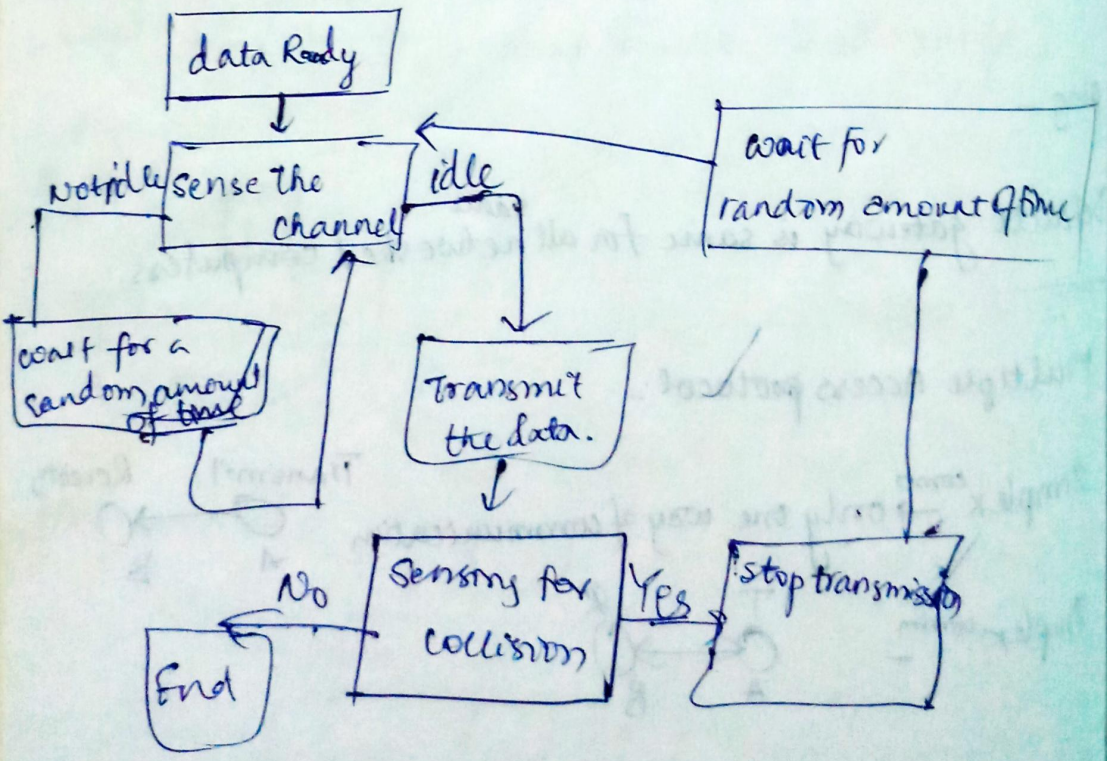
Carrier sense multiple Access with collision Detection

CSMA/CA

Avoidance

IEEE 802.11a for Wi-Fi

CSMA/CD



1) Data ready to be transmitted

2) Sense for carrier wave

if idle (absence of carrier wave)

→ Transmit data

→ listen for collision

If no collision then start for next transition or stop

else:

stop transmission and wait for a random amount of time (& again sense the channel for carrier wave during retransmission) and go to step 2.

else:

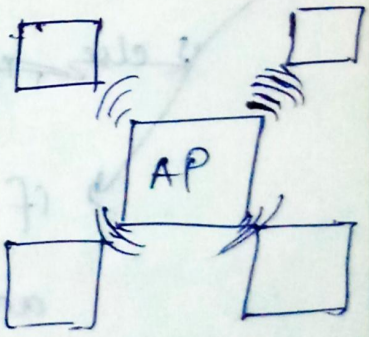
wait for random amount of time & go to step 2.

CSMA/CA → wireless LAN

IEEE 802.11a

Wi-fi

Wireless fidility



Hidden Terminal Problem

Exposed Terminal

IFS → Inter Space Time

RTS → Request to Send

CTS - Clear to Send

1. Ready to transmit

2. Check if channel is idle

→ if idle wait for IFS period of time.

→ if entire IFS time if the channel is idle, then the station will ~~reserve the channel~~ Send RTS to the destination station.

→ the destination station will reply with CTS message.

→ the source station starts data transmission.

else:

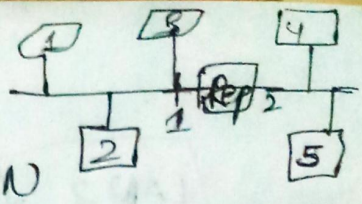
if entire IFS is not idle,

then wait for Random Amount of Time

~~else if~~ and go to Step 2.

if not idle wait for Random Amount of Time and goto Step2.

Interconnecting Devices:-



Repeater :- Extending the length of LAN

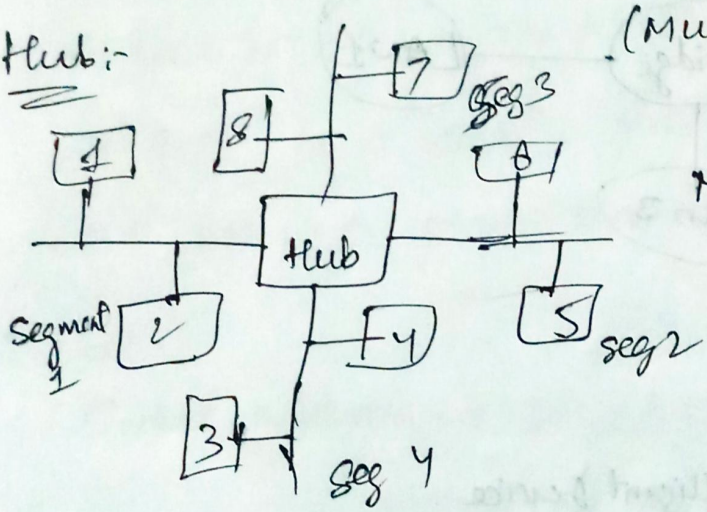
Repeat or regenerate the incoming signal

No filtering is done in repeater.

Repeater is not an intelligent device.

2 port device

Hub :- (Multiport Repeater)



Multiple ports it has.

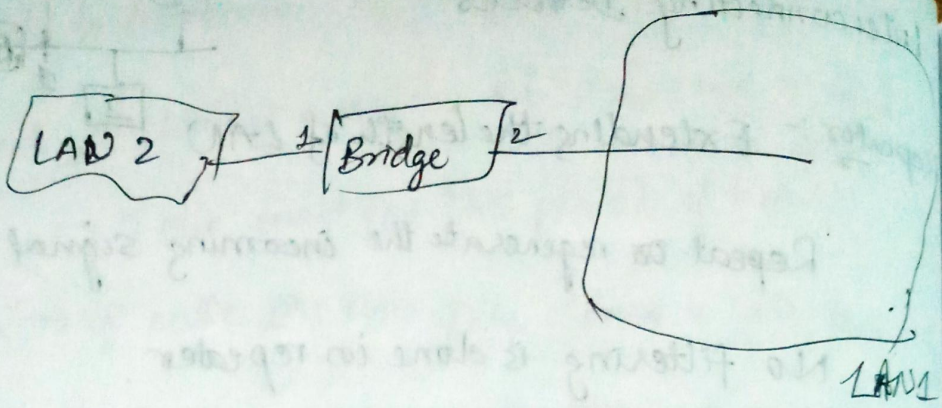
Bridge :-

It is an intelligent device

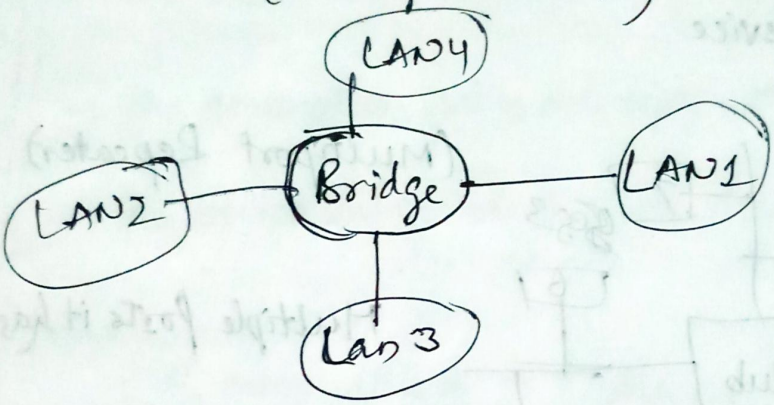
→ filtering and forwarding

Bridge uses MAC Address but not IP address

2 port devices



Switch:- (Multiport Bridge)



Router:-

Intelligent Device

Interconnecting different networks

Use IP address for forwarding

methods of communication:

→ Client / Server Architecture

→ P2P / Peer to Peer model → client & server have same importance

HTTP [Hyper Text Transfer Protocol]

- It is a client server protocol
- → used for web browsing
- → http server will always listen to port 80.
- → web browser is the HTTP client usually
- → HTTP is a stateless protocol.
- → HTTP is a connection oriented protocol.

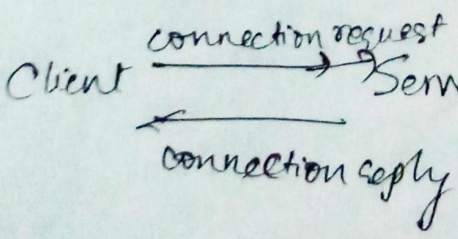
Socket:-

Socket address ⇒ IP address + Port no

HTTP:-

Connection Oriented Protocol

→ logical connection



Client.py

1. socket creation
2. Connect request

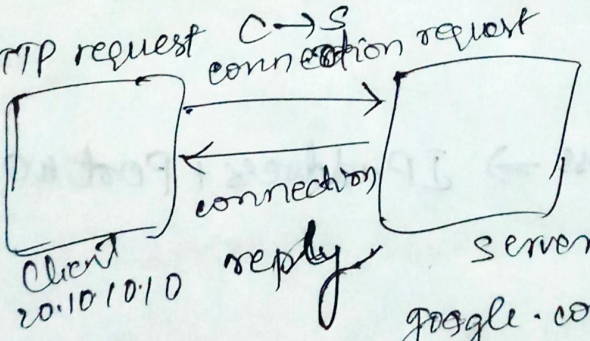
Data Transfer

server.py → IP add, port number

1. socket creation
2. bind an (ip + port) to the socket
3. listen for connection
4. accept

Data Transfer

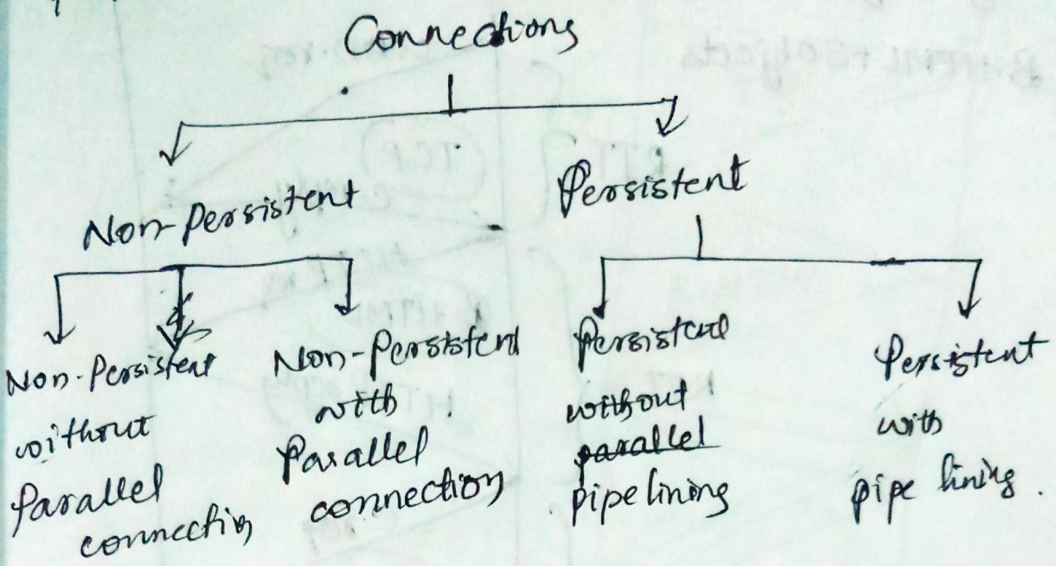
1. open browser
2. URL
3. http://google.com 80 /index.html
   port no — page location
4. connection establishment
5. HTTP request C → S



(DNS)

6. GET req or POST req.

TCP - Transfer Control Protocol

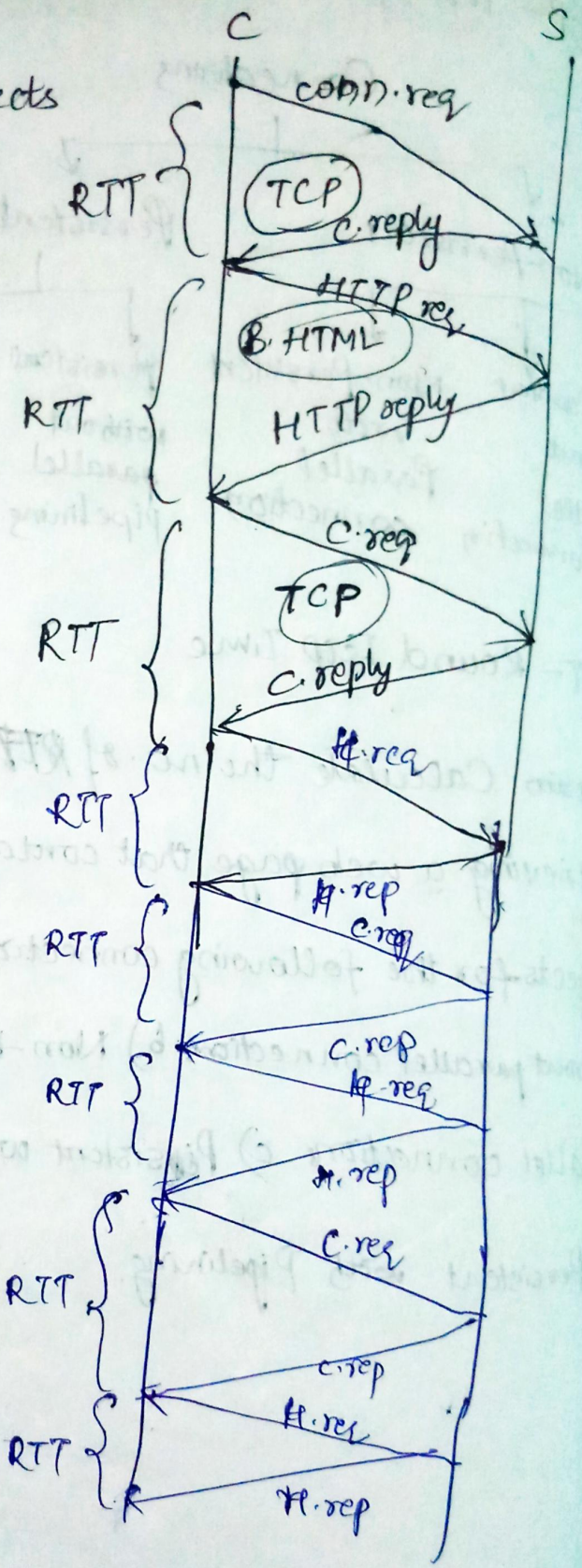


RTT - Round Trip Time

Calculate the no. of RTTs required for retrieving a web page that contains three reference objects for the following connections a) Non-Persistent http without parallel connections b) Non-Persistent http with parallel connections c) Persistent without pipelining, d) Persistent with Pipelining.

3 ref objects

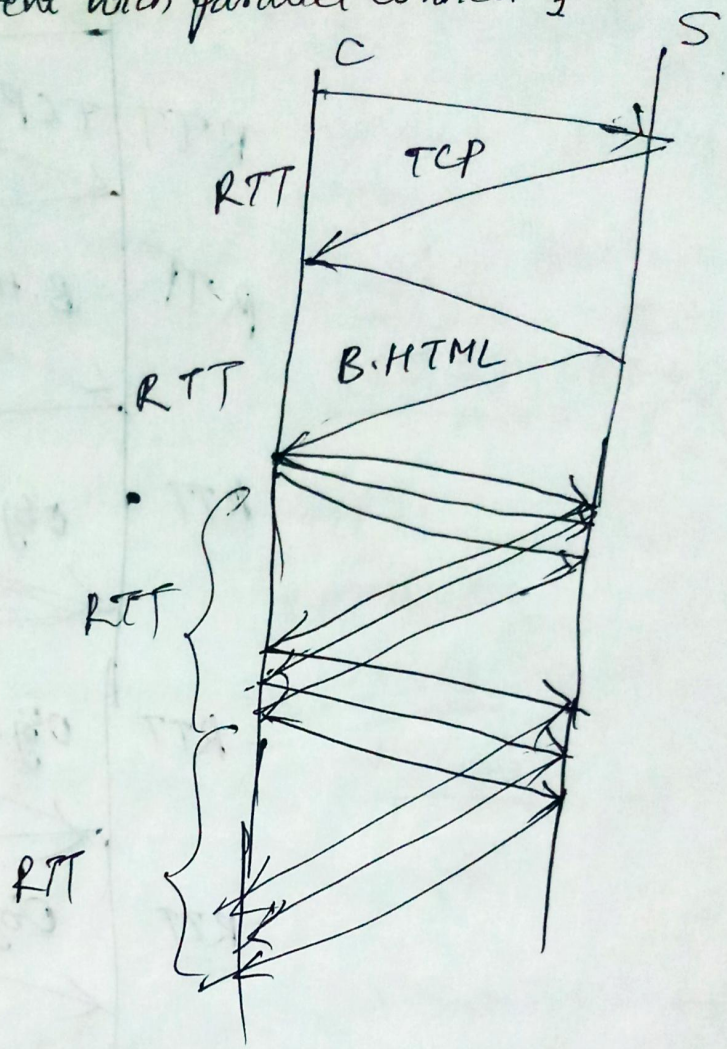
B. HTML + 3 objects



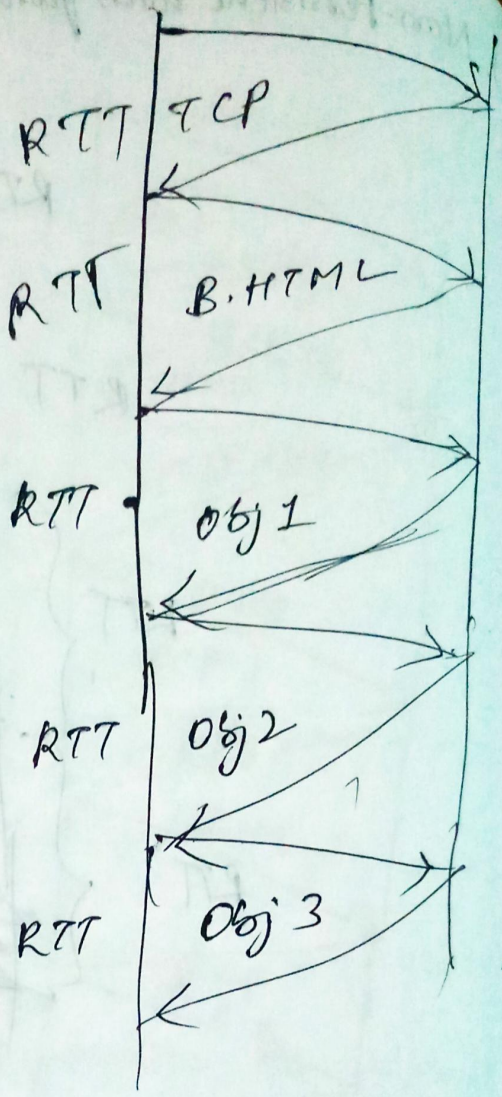
a. 8 RTT

b. 4 RTT

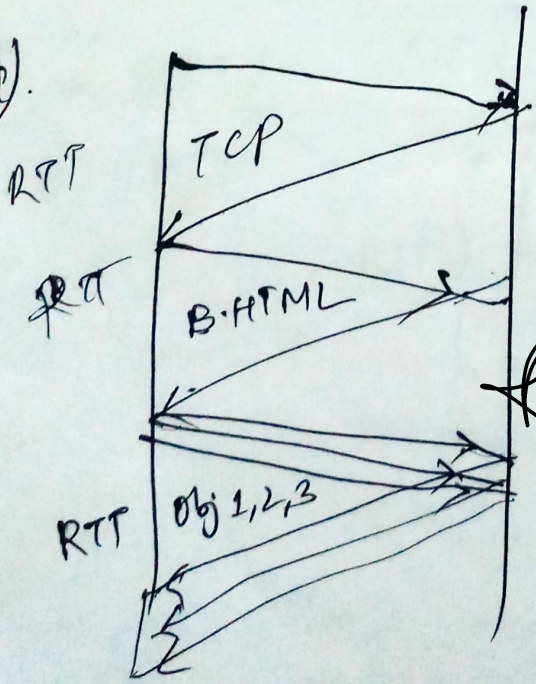
b). Non-Persistent with parallel Connection



c).


RTT
TCP
RTT
B.HTML
RTT
Obj 1
RTT
Obj 2
RTT
Obj 3

d).


RTT
TCP
RTT
B.HTML
RTT
Obj 1,2,3

Ram Bhat

Obtain the number of rtt's 7 ref objects.

i. TCP → 1
B·HTML → 1
7×2 = 14
RTT = 16

ii. TCP → 1
B·HTML → 1
7×1 = 7
RTT = 9

2×2 = 2
= 2
RTT = 6

iii. TCP → 1
B·HTML → 1
7×1 = 7
RTT = 9

iv. TCP = 1
B·HTML = 1
1× = 1
RTT = 3

IP addressing :-

IP → Internet Protocol

IPv4	IPv6
1. 32 bits or 4 bytes	1. 128 bits or 16 bytes
2. only 2^{32} IP addresses are possible.	2. only 2^{128} IP addresses are possible.
3. Dotted decimal notation — . — . — . — Eg :- 128.168.20.7	3. Odon hexadecimal notation 2 bytes : 2 bytes : 2 bytes : 2 bytes : 2 bytes : 2 bytes : 2 bytes : 2 bytes Eg : A12F : 1234 : 9FAB : 8732 : 1234 : 000A : 1111 : ABCD

Obtain the network address corresponding to the IP addr

192.168.211.50 if subnet mask address is 255.0.0.0

Ans:- 192.0.0.0.

IP addr AND subnet mask address

192.168.211.50 AND 255.0.0.0

Q. IP addr - 192.168.211.50 Subnet Mask address 255.2

Ans:- 192.168.208.0

192.168.208.0

192.168.211.50

255.255.248.0

192.168.208.0

211 → 1101 0011

248 → 1111 1000

1101 0.000

0

128
+64
+16
208

24 = 16
26 = 64
27 = 128
208

Class	Start Addr	End Addr	
A	0.0.0.0	127.255.255.255	0
B	128.0.0.0	191.255.255.255	10
C	192.0.0.0	223.255.255.255	110
D	224.0.0.0	239.255.255.255	1110
E	224.0.0.0	255.255.255.255	1111

obtain the classes for the following IP addresses.

1. 230.1.5.7 D
2. 190.2.3.4 B
3. 10.1.23 A
4. 242.5.6.7 E

n bits | 32 − n bits

network bits | host bits

eg:- 12 bits are network bits

20 bits are host bits → 32 − 12 = 20

Identify the network bits

11111111.11110000.00000000.00000000

Mask address | 255.240.0.0

class	no. of n/w bits	no. of host bits	Mask addr
A	8	24	255.0.0.0
B	16	16	255.255.0.0
C	24	8	255.255.255.0
D	—	—	Multicast Address
E	—	—	Reserved addresses

Consider the IP address 130.20.30.40 and calculate

i. the class of the IP address

ii. the no. of network bits and no. of host bits present.

iii. the total no. of networks possible and the total number of hosts possible in each network.

iv. Obtain the mask addr and calculate the network address.

Ans: i. Belongs to class B

10000

2 | 130
2 | 65 — 0
2 | 32 — 1
2 | 16 — 0
2 | 8 — 0
2 | 4 — 0
2 | 2 — 0
1 — 0

1000

ii. No. of network bits — 16

No. of host bits — 16

iii. 2^{16} → Networks possible

2^{16} → Networks possible

iv. 11111111.11111111.00000000.00000000

Mask address — 255.255.0.0

Network address

130.20.30.40

255.255.0.0

130.20.0.0