IAE ACADEMY INSTITUTE FOR ENGINEERS

GATE CS

IAE Academy free study material · Updated 13 October 2026

Computer Networks Formula Sheet

INSIDE

- 59 formulas in 7 topics
- When to use each one
- Taken from the free topic notes on iaeacademy.in

Online, one page per topic: **iaeacademy.in/study/computer-networks/formula-sheet**

FREE COUNSELLING

Preparing for GATE CS? Talk to IAE Academy.

Or call **+91 84204 36524**

Every day, 10:30 am to 7:30 pm

Chat on WhatsApp

Free. Not for sale. Pass it on.

Layering and switching: circuit, packet and virtual circuit

HEADER EFFICIENCY

$$\eta = \frac{\text{payload bits}}{\text{payload bits} + \text{header (and trailer) bits}}$$

Fraction of transmitted bits that is useful data; with several layers, add up the headers of all layers that wrap the payload.

MESSAGE DELAY IN A CIRCUIT-SWITCHED NETWORK

$$T_{\text{circuit}} = T_{\text{setup}} + \frac{L}{R} + T_{\text{prop,path}}$$

L = message size in bits, R = rate of the reserved circuit in bits per second (R/n of the link rate when n circuits share it); T_setup is the set-up time given in the question (teardown ignored); T_prop,path is the sum of the propagation delays of all links. Switch processing is ignored unless stated.

ONE PACKET OVER K LINKS (STORE-AND-FORWARD)

$$T = k \cdot \frac{L}{R} + \sum_{i=1}^{k} \frac{d_i}{s}$$

The whole packet is transmitted again on every link, so transmission delay is counted k times; all links at rate R, no queueing or processing. If the links have different propagation delays, add the delays given.

A MESSAGE SENT AS N EQUAL PACKETS OVER K LINKS

$$T = (k + n - 1) \cdot \frac{p}{R} + \sum_{i=1}^{k} \frac{d_i}{s}$$

Each packet has p bits including its header, all links at rate R, no queueing, and the packets go back to back. The first packet needs k transmission times; each later packet arrives one transmission time after the one before it (pipelining). T is the time from the first bit leaving the source until the last bit of the last packet reaches the destination.

N EQUAL PACKETS OVER LINKS OF DIFFERENT RATES

$$T = \sum_{i=1}^{k} \frac{p}{R_i} + (n - 1)\frac{p}{R_{\min}} + \sum_{i=1}^{k} \frac{d_i}{s}$$

R_i = rate of link i, R_min = rate of the slowest link (the bottleneck). The first packet crosses every link once; the later packets leave the bottleneck one p/R_min apart and keep that spacing. With all rates equal this reduces to the previous formula. No other traffic, no queueing apart from waiting for the packet ahead.

PAYLOAD SIZE THAT MINIMISES TOTAL DELAY

$$x_{\text{opt}} = \sqrt{\frac{M\,h}{k - 1}}$$

M = message bits, h = header bits per packet, k >= 2 links of equal rate, ignoring the rounding of M/x to a whole number. Derivation: delay times R is (k-1+M/x)(x+h) = (k-1)x + (k-1)h + M + Mh/x; setting its derivative with respect to x to zero gives (k-1) = Mh/x^2. Propagation does not depend on x.

Performance metrics: delay, throughput and link utilisation

TRANSMISSION DELAY

$$T_t = \frac{L}{R}$$

L = packet size in bits, R = link rate in bits per second. Depends on the packet and the rate, not on the length of the wire.

PROPAGATION DELAY

$$T_p = \frac{d}{v}$$

d = length of the link, v = propagation speed in the medium. Independent of the packet size and the link rate.

ROUND-TRIP TIME (NEGLIGIBLE ACK SIZE)

$$\mathrm{RTT} = 2\,T_p$$

For a short probe or acknowledgement whose transmission delay is negligible. If transmission delays matter, add them explicitly.

ONE PACKET OVER N LINKS, STORE-AND-FORWARD

$$T_{\text{end-to-end}} = \sum_{i=1}^{N} \left(\frac{L}{R_i} + T_{p,i} \right)$$

Each router receives the whole packet before forwarding it. Queuing and processing delays are zero unless given. With equal links this is $N\,(L/R + T_p)$.

P PACKETS BACK TO BACK OVER N EQUAL LINKS

$$T = (N + P - 1)\,\frac{L}{R} + N\,T_p$$

All packets have the same size L, are sent without gaps, and no queuing occurs other than waiting behind earlier packets. The first packet needs N transmission times; each later packet finishes one transmission time after the one before.

THROUGHPUT OF A PATH

$$\text{Throughput} = \min(R_1, R_2, \ldots, R_N)$$

Steady transfer of a long stream over links in series with no other traffic, and a sender window large enough not to be the limit. Delays do not change this rate; they only shift when the stream starts and ends.

FILE OF F BITS IN PACKETS OF L BITS, LINKS IN SERIES

$$T = \frac{F}{R_b} + \sum_{i \neq b} \frac{L}{R_i} + \sum_{i=1}^{N} T_{p,i}$$

Store-and-forward; $R_b = \min R_i$ is the bottleneck rate (link b); F is a whole number of packets of L bits sent back to back; no other traffic, no processing delay. Every link other than the bottleneck adds one packet transmission time. With equal links and $F = PL$ this is $(N + P - 1)L/R + N\,T_p$. For a very large F the first term dominates.

BANDWIDTH-DELAY PRODUCT

$$\mathrm{BDP} = R \times T_p$$

Number of bits that fill the link from one end to the other (bits in flight in one direction). Using $R \times \mathrm{RTT}$ gives the bits that can be in flight during one round trip.

WINDOW-LIMITED THROUGHPUT

$$\text{Throughput} = \min\left(R,\ \frac{W}{T_{\text{cycle}}}\right)$$

W = sender window in bits (a TCP window is given in bytes, so multiply by 8). T_{cycle} = time from sending the first bit of a window to receiving its acknowledgement: $T_t + 2T_p$ for frames of transmission time T_t, or simply RTT when transmission time is negligible. With $W = NL$ this gives $U = N/(1 + 2a)$, as below.

STOP-AND-WAIT UTILISATION

$$U = \frac{T_t}{T_t + 2T_p} = \frac{1}{1 + 2a}$$

No errors, acknowledgement transmission time and processing time negligible. If the ACK has a non-negligible transmission time T_{ack}, the cycle is $T_t + 2T_p + T_{ack}$.

EFFECTIVE THROUGHPUT

$$\text{Throughput} = U \times R$$

R = link rate. Equivalently, one frame of L bits is delivered per cycle: $L/(T_t + 2T_p)$.

STOP-AND-WAIT WITH FRAME LOSS

$$U = \frac{1 - p}{1 + 2a}$$

Each frame is lost independently with probability p, acknowledgements are never lost, and every failed attempt costs one full cycle $T_t + 2T_p$ (timeout equal to the cycle). The expected number of transmissions per frame is $1/(1 - p)$.

USEFUL THROUGHPUT WITH HEADER OVERHEAD

$$\text{Useful throughput} = U \times R \times \frac{L - H}{L}$$

L = whole frame in bits including the header, H = header bits. Use this only when the question asks for data (payload) throughput; otherwise headers count as transmitted bits.

SLIDING-WINDOW UTILISATION

$$U = \min\left(1,\ \frac{N}{1 + 2a}\right)$$

N = sender window in frames, $a = T_p/T_t$, no errors, ACK transmission negligible. Applies to Go-Back-N and Selective Repeat alike when there are no losses. $N = 1$ gives stop-and-wait.

WINDOW NEEDED TO KEEP THE LINK BUSY

$$N_{\min} = \lceil 1 + 2a \rceil$$

Smallest window giving $U = 1$. Equivalently, the window must cover one full cycle: $N\,T_t \geq T_t + 2T_p$.

GENERAL RULE FOR SEQUENCE NUMBERS

$$W_s + W_r \leq 2^k$$

W_s = sender window, W_r = receiver window, both in frames; k = bits in the sequence number. Go-Back-N has $W_r = 1$, giving $W_s \leq 2^k - 1$; Selective Repeat with $W_s = W_r$ gives $W_s \leq 2^{k-1}$.

MAXIMUM WINDOW, GO-BACK-N

$$N_{\max}^{GBN} = 2^k - 1$$

k = bits in the sequence number. A window of 2^k would make a whole new window look like a retransmission of the old one.

MAXIMUM WINDOW, SELECTIVE REPEAT

$$N_{\max}^{SR} = 2^{k-1}$$

Sender and receiver windows are equal; the two windows together must not exceed the 2^k numbers.

SEQUENCE BITS FOR A GIVEN WINDOW

$$k_{GBN} = \lceil \log_2(N + 1) \rceil, \qquad k_{SR} = \lceil \log_2(2N) \rceil$$

N = sender window size in frames. Use $N_{\min}$ from the utilisation formula when the question asks for the field that keeps the link fully busy.

Data link layer: error detection, medium access control and Ethernet

DETECTION WITH MINIMUM DISTANCE D

$$\text{errors always detected} = d - 1$$

A code of minimum distance d detects every pattern of up to d - 1 bit errors.

CORRECTION WITH MINIMUM DISTANCE D

$$\text{errors always corrected} = \left\lfloor \frac{d-1}{2} \right\rfloor$$

Decoding to the nearest codeword corrects up to this many bit errors.

CRC FRAME LENGTH

$$n = m + r, \qquad r = \deg G(x)$$

m data bits plus an r-bit remainder give an n-bit frame; the generator has r + 1 bits.

NUMBER OF HAMMING CHECK BITS

$$2^r \geq m + r + 1$$

The smallest r for which the syndrome can point to any single-bit error in an (m + r)-bit codeword.

TRANSMISSION TIME AND THE RATIO A

$$T_t = \frac{L}{B}, \qquad a = \frac{T_p}{T_t}$$

L = frame length in bits, B = bandwidth in bits per second, T_p = one-way propagation delay. A small a means a station hears another's frame early in its transmission, so carrier sensing works well.

PURE ALOHA THROUGHPUT

$$S = G\, e^{-2G}$$

S is the fraction of time that carries a successful frame; the maximum is about 18.4% at G = 0.5.

SLOTTED ALOHA THROUGHPUT

$$S = G\, e^{-G}$$

The maximum is about 36.8% at G = 1.

MINIMUM FRAME SIZE FOR CSMA/CD

$$L \geq 2\, T_p\, B$$

L = frame length in bits, T_p = one-way propagation delay (distance divided by propagation speed), B = bandwidth in bits per second. Put another way, the transmission time must be at least 2 T_p.

Routing: distance vector and link state

BELLMAN-FORD (DISTANCE-VECTOR) UPDATE

$$D_x(y) = \min_{v \in N(x)} \{\, c(x,v) + D_v(y) \,\}$$

For y not equal to x (D_x(x) = 0). x is the router, N(x) its neighbours, c(x,v) the (positive) link cost to neighbour v, D_v(y) the cost to destination y that v advertised. The neighbour v giving the minimum is the next hop.

DIJKSTRA RELAXATION

$$d(v) = \min\{\, d(v),\ d(u) + c(u,v) \,\}$$

Applied to every neighbour v of the node u just made permanent (only tentative neighbours can improve). Requires all link costs to be non-negative.

DIJKSTRA RUNNING TIME

$$T = O(n^2)\ (\text{array}), \qquad T = O((n+e)\log n)\ (\text{binary heap})$$

n routers, e links; a straightforward implementation scans for the minimum, a heap makes extraction logarithmic.

IPv4: addressing, CIDR, fragmentation and NAT

ADDRESSES IN A BLOCK

$$\text{addresses} = 2^{32-p}$$

For a block with prefix length p, counting the network and broadcast addresses.

USABLE HOSTS IN A BLOCK

$$\text{hosts} = 2^{32-p} - 2$$

For a block with prefix length p. The two subtracted addresses are the network and broadcast addresses. Valid for p up to 30.

NUMBER OF SUBNETS (CIDR CONVENTION)

$$\text{subnets} = 2^{s}$$

s is the number of host bits borrowed (new prefix length minus old prefix length). All-zeros and all-ones subnets are allowed.

NUMBER OF SUBNETS (OLDER CLASSFUL CONVENTION)

$$\text{subnets} = 2^{s} - 2$$

Only when the question excludes the all-zeros and all-ones subnet numbers.

BLOCK SIZE IN THE INTERESTING OCTET

$$\text{block size} = 256 - (\text{mask value in that octet})$$

The interesting octet is the one where the mask is neither 255 nor 0. Subnets start at multiples of the block size in that octet. If p is a multiple of 8 there is no such octet: the host part is whole octets.

MAXIMUM DATA PER FRAGMENT

$$D_{\max} = 8\left\lfloor \frac{\text{MTU} - H}{8} \right\rfloor$$

H is the header length in bytes (20 without options). Data in every fragment except the last must be a multiple of 8 bytes.

NUMBER OF FRAGMENTS

$$n = \left\lceil \frac{L - H}{D_{\max}} \right\rceil$$

L is the original total length in bytes, H the header length. Assumes a single fragmentation at one link and the same header length H in every fragment.

FRAGMENT OFFSET FIELD

$$\text{offset} = \frac{\text{bytes of data before this fragment}}{8}$$

Measured from the start of the original datagram's data. MF = 1 for every fragment except the last.

DATA BYTES CARRIED BY A FRAGMENT

$$\text{first} = 8 \times \text{offset}, \qquad \text{last} = 8 \times \text{offset} + (\text{total length} - 4 \times \text{HLEN}) - 1$$

Byte positions in the original data, numbered from 0. Total length is in bytes; HLEN is in 4-byte words.

TCP: flow control, congestion control and sockets

ACKNOWLEDGEMENT NUMBER

$$\text{ack} = \text{seq} + L \quad (+1 \text{ if the segment carries SYN or FIN})$$

For a segment that arrives in order, with sequence number seq and L data bytes. Because the ACK is cumulative, an out-of-order segment brings back the previous ack number again (a duplicate ACK).

SENDER'S EFFECTIVE WINDOW

$$\text{usable window} = \min(\text{rwnd},\ \text{cwnd})$$

rwnd is the receiver's flow-control limit; cwnd is the congestion window from the next section.

MAXIMUM THROUGHPUT WITH WINDOW W

$$\text{Throughput} = \min\left(\frac{W}{\text{RTT}},\ \text{link rate}\right)$$

W is in bits when the rate is in bits per second. Assumes no loss, header overhead ignored and ACK transmission time negligible. When the window equals the bandwidth-delay product, the pipe is exactly full. If a question defines one cycle as RTT plus the transmission time of the window (or of a segment), use that cycle in place of RTT.

LINK UTILISATION (SLIDING WINDOW, CONTINUOUS)

$$\text{Utilisation} = \min\left(1,\ \frac{W}{\text{link rate} \times \text{RTT}}\right)$$

Link rate times RTT is the bandwidth-delay product. Keep it in the same unit as W (bits or bytes).

SEQUENCE-NUMBER WRAP-AROUND TIME

$$T_{\text{wrap}} = \frac{2^{32} \times 8}{\text{rate in bits per second}} \text{ seconds}$$

The sequence field is 32 bits and counts bytes, so 2^{32} bytes pass before numbers repeat, if the sender sends continuously at the full rate.

MINIMUM SEQUENCE-NUMBER BITS

$$n_{\min} = \lceil \log_2(\text{rate in bytes per second} \times \text{lifetime in seconds}) \rceil$$

The condition is that sequence numbers (counting bytes) must not wrap around while a segment can still be alive in the network. The lifetime is the maximum segment lifetime given in the question. Convert the rate from bits to bytes per second by dividing by 8.

RETRANSMISSION TIMEOUT (STANDARD ESTIMATOR)

$$\text{RTO} = \text{SRTT} + 4 \times \text{RTTVAR}$$

SRTT $\leftarrow (1 - \frac{1}{8})\,\text{SRTT} + \frac{1}{8}\,R$ and RTTVAR $\leftarrow (1 - \frac{1}{4})\,\text{RTTVAR} + \frac{1}{4}\,|R - \text{SRTT}|$, where R is a new RTT sample. RTTVAR is updated first, using the old SRTT. On the first sample, SRTT = R and RTTVAR = R/2, so RTO = 3R.

MULTIPLICATIVE DECREASE

$$\text{ssthresh}_{\text{new}} = \max\left(\left\lfloor \frac{\text{cwnd}}{2} \right\rfloor,\ 2\right)\ \text{MSS}$$

cwnd is the window at the moment the loss is detected. Unless the question says otherwise, take cwnd/2 in whole MSS (rounded down for an odd window).

Application layer: DNS and HTTP

TIME FOR AN ITERATIVE RESOLUTION BY THE RESOLVER

$$T_{\text{DNS}} = R_{\text{host}} + \sum_{i} R_i$$

R_host is the RTT from host to local resolver and R_i the RTT from the resolver to each server it queries in turn. Assumes processing time is negligible and each query and its reply is one UDP exchange costing one RTT (no losses or retransmissions). With a cached TLD referral, drop the root term; with a cached final answer, only R_host remains.

AVERAGE ACCESS TIME WITH A WEB CACHE

$$T_{\text{avg}} = h\,T_{\text{hit}} + (1 - h)\,T_{\text{miss}}$$

h is the hit ratio (fraction of requests the cache answers), T_hit the time to get an object from the cache and T_miss the time to get it through the cache from the origin server. Assumes every request takes one of these two times.

NON-PERSISTENT, SERIAL

$$T = 2(n + 1)\,\text{RTT}$$

Each of the n+1 files costs 1 RTT for the handshake and 1 RTT for request and response.

NON-PERSISTENT, M PARALLEL CONNECTIONS

$$T = 2\,\text{RTT} + 2\left\lceil \frac{n}{m} \right\rceil \text{RTT}$$

Base file fetched alone first; objects then go in batches of m. Assumes equal object times and negligible transmission time.

PERSISTENT WITHOUT PIPELINING

$$T = (n + 2)\,\text{RTT}$$

One handshake, one RTT for the base file, one RTT for each object.

PERSISTENT WITH PIPELINING

$$T = 3\,\text{RTT}$$

One handshake, one RTT for the base file, one RTT for all objects together (requests sent without waiting). Transmission time is ignored.

FROM IAE ACADEMY

Preparing for GATE CS?

IAE Academy teaches GATE CS in Ballygunge, Kolkata, and online. Counselling is free: ask us about batches, fees or your preparation.

FREE COUNSELLING

Chat on WhatsApp

Call +91 84204 36524

Open every day, 10:30 am to 7:30 pm.

Scan to chat on WhatsApp

THE GATE CS COURSE

Classroom batches in Ballygunge and online batches. Fees and details are on the course page.

Scan: course page

See the course

FREE ON IAEACADEMY.IN

Notes and solved previous-year questions for Computer Networks, topic by topic.

Scan: the notes

Open the notes

Found this useful? Send it to a friend preparing for GATE.

Send to a friend on WhatsApp

IAE Academy free study material. The formulas come from the free Computer Networks notes on iaeacademy.in. Prepared and published by IAE Academy, Kolkata. Free to read and free to share, unaltered and with attribution.

IAE ACADEMY INSTITUTE FOR ENGINEERS

GATE coaching in Kolkata since 2012 · EC, EE, IN, CS and IT, DA, BT and CE · in the classroom and online

227A Rash Behari Ave, Ballygunge, Kolkata 700019

+91 84204 36524 · iaeacademy.in · Updated 13 October 2026

Get directions