

INST 346 Exam 1 Topic List
September 26, 2017

Networks

- Types of networks (social networks, computer networks, entity-relationship networks, ...)
- Node-link diagrams
- Layered Internet architecture (encapsulation)

The Internet

- Structure of the “Internet core” (ISPs, IXP’s, CDN’s)
- Packet switching (store and forward, routing)
- Types of delays (transmission, propagation, queueing, processing)
- Computing throughput
- Causes of packet loss (buffer overflow, bit errors, excessive delay)
- Traceroute

Service Model

- Example applications (Web, email, streaming video, ...)
- Client-server architecture
- Processes
- Sockets
- Port numbers
- Designing an application level protocol

The Web

- HTTP message formats (GET, Conditional GET, HEAD, POST, DELETE, response)
- HTTP interaction
- Sequence diagrams
- HTTP use of TCP (persistent, non-persistent)
- Cookies
- Proxy servers

Email

- Email architecture (mail servers, “user agents”)
- Mail message format
- SMTP (message format, interaction)
- POP3 (message format, interaction)
- Interaction scenario
- 7-bit ASCII limitation

Socket Programming

- Creating sockets in Python 3 (UDP, TCP)
- Using UDP sockets in Python 3
- Using TCP sockets in Python 3

Domain Name System

- Application “stacking” (one application layer protocol using another)
- Distributed hierarchical architecture (root servers, TLD servers, authoritative servers)
- Resource records
- DNS message format

Streaming

- Media coding (audio, video)
- Quality of service requirements (continuous replay, packet loss, latency, adaptation)
- Client-side buffering
- DASH

Transport Layer

- Socket addressing (UDP, TCP)
- Checksums

UDP

- UDP service goals (unordered, unguaranteed “best effort” delivery)
- Applications that use UDP
- UDP message format

TCP

- TCP service goals (ordered, guaranteed, eventual delivery)
- Applications that use TCP
- Reliable data transfer (checksums, acknowledgement, timeouts, pipelining)
- Reading Finite State Model diagrams
- TCP message format
- Adaptation (timeout tuning, flow control)

Overarching skills

- Timing analysis
- Protocol inspection using Wireshark
- Understanding how protocol layers work together
- Understanding why protocols differ
- Designing new protocols
- Protocol implementation
- Understanding the consequences of design decisions (technical, social)