Examinable Topics

UG3 Computer Communications & Networks (COMN)

Myungjin Lee
myungjin.lee@ed.ac.uk

Software-Defined Networking

- What is SDN?

- The working mechanism of OpenFlow

Software-Defined Networking

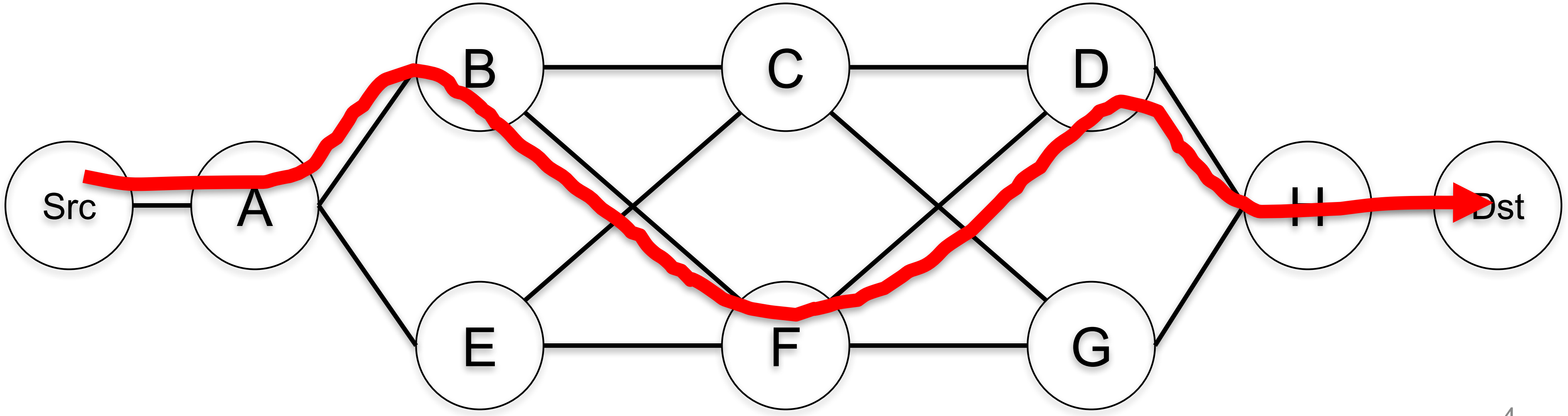
- Policy on matching flow entry in OpenFlow
  - Exact match has the highest priority
  - All wildcard entries have a priority associated with them
  - Higher priority entries match before lower priority ones

	Src IP	Src Port	Dst IP	Dst Port	Protocol
Flow:	1.2.3.4	10000	5.6.7.8	22	TCP

	IP Src	IP Dst	IP Prot	TCP sport	TCP dport	Priority	Action
Entry 1:	*	5.6.7.8	*	*	*	1	port6
Entry 2:	*	*	*	*	22	10	drop

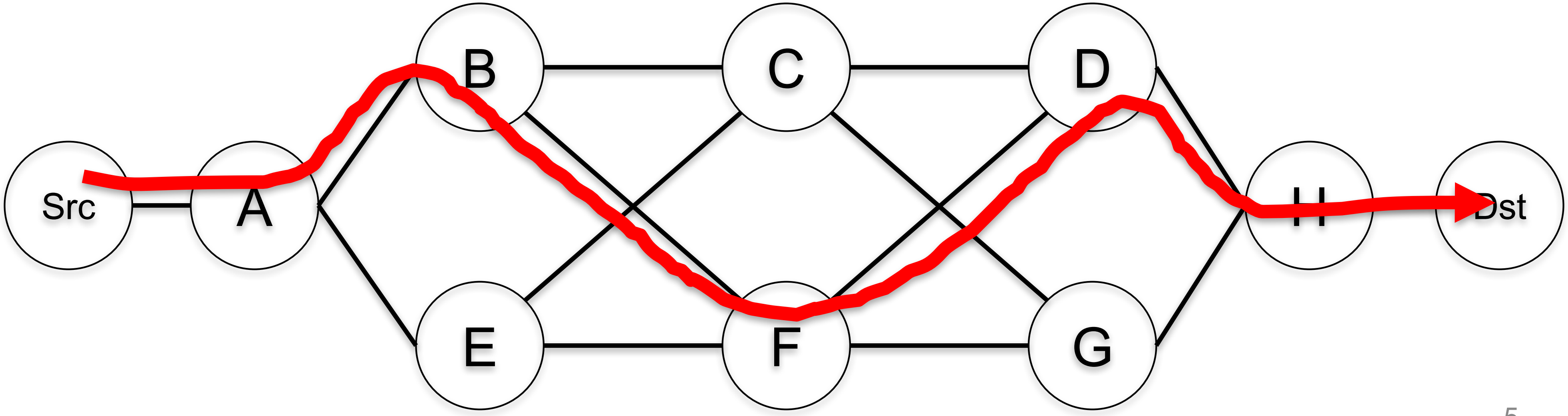
Software-Defined Networking

- Example: identifying path of a flow
  - A flow is defined as <srcIP, srcPort, dstIP, dstPort, proto>
  - How many paths exist from Src to Dst?
  - Assumption: Forwarding rules are written as wildcards



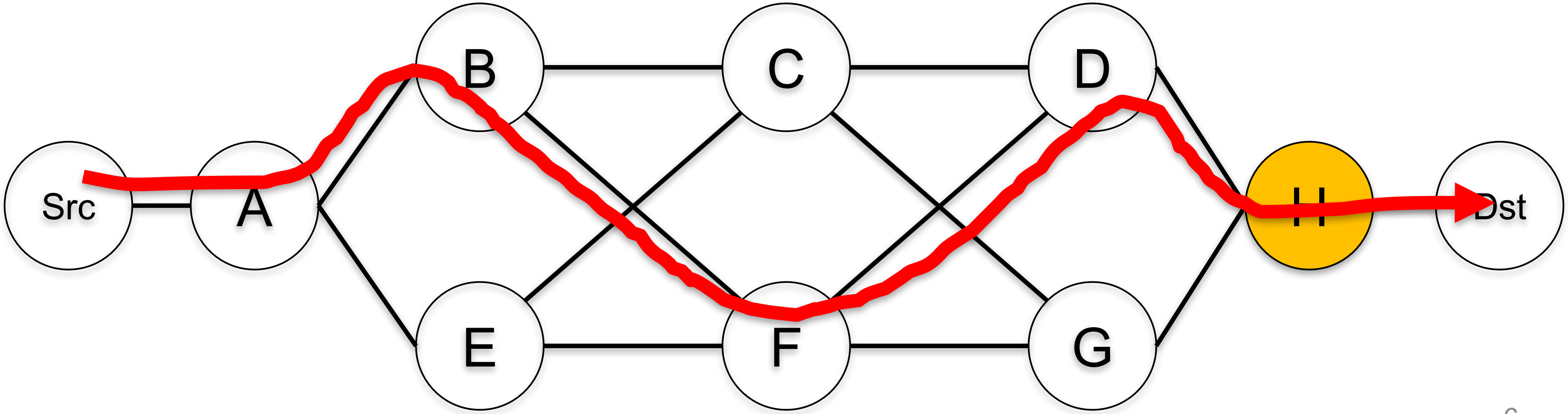
Software-Defined Networking

- Q: How to determine the path taken by packets of a flow?
- A: Do reverse search



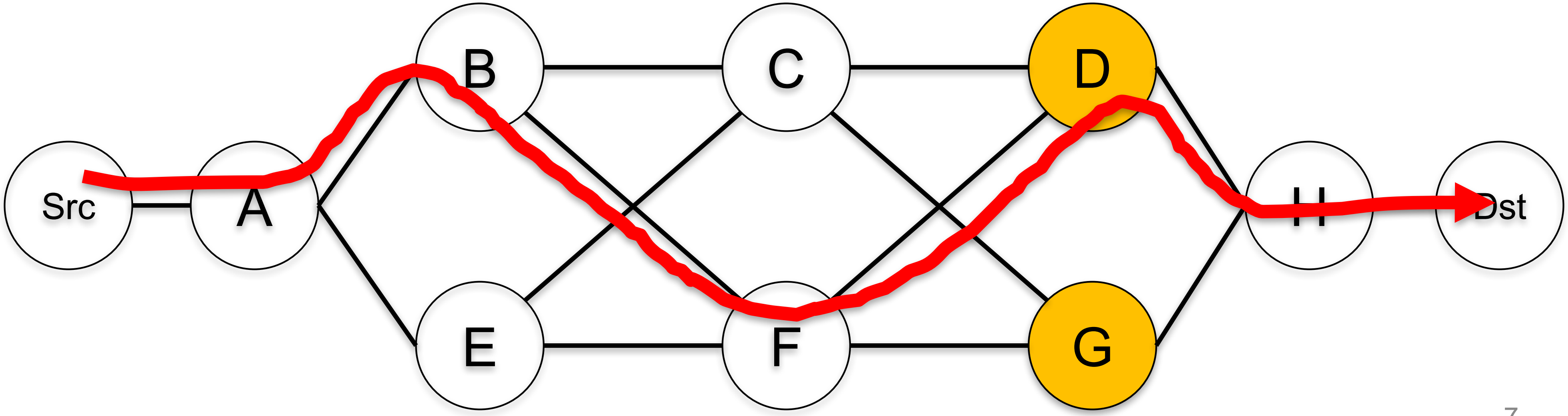
Software-Defined Networking

- Step 1: The controller writes a rule at switch H
  - Forward packets of the red flow to the controller (Rule1) while giving the rule higher priority than forwarding rules’
  - It would know that the packets pass switch H
- Step 2: The controller removes Rule 1



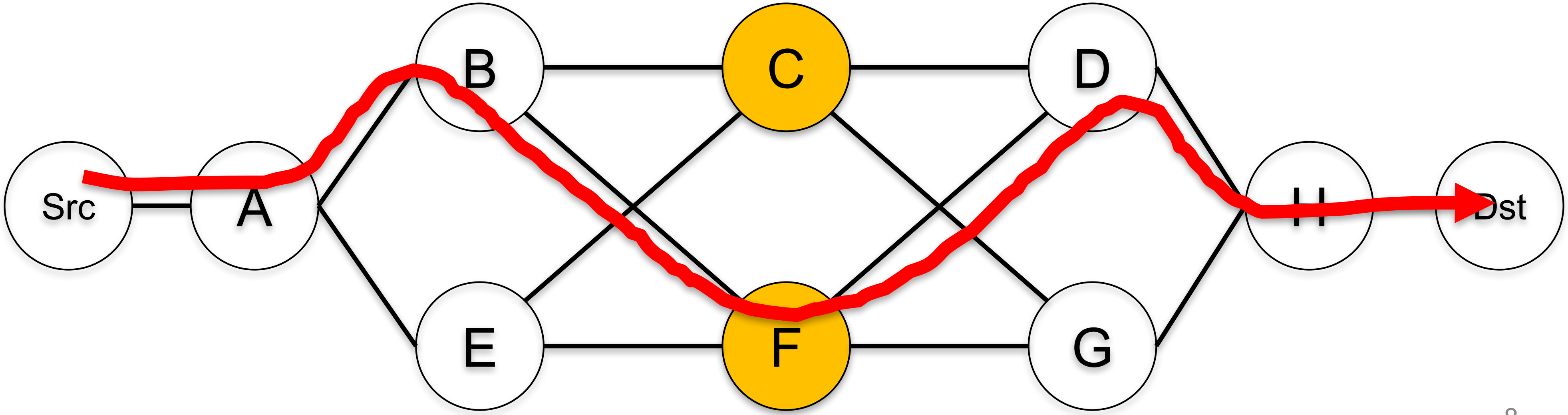
Software-Defined Networking

- Step 3: The controller writes a rule at switches D & G
  - Forward packets of the red flow to the controller (Rule1) while giving the rule higher priority than forwarding rules'
  - It would know that the packets pass switch D
- Step 4: The controller removes Rule 1



Software-Defined Networking

- Step 5: The controller writes a rule at switches C & F
  - Forward packets of the red flow to the controller (Rule1) while giving the rule higher priority than forwarding rules'
  - It would know that the packets pass switch F
- Step 6: The controller removes Rule 1



Introduction

- Understanding of basic concepts and terminologies
  - Bandwidth (or Capacity)
  - Throughput
  - Delay
  - Loss
  - BDP (Bandwidth-delay product)
  - Layering
  - Encapsulation

  and so forth

Introduction

- Characteristics of packet-switching
  - statistical multiplexing
  - store-and-forward
  - queuing delay and loss

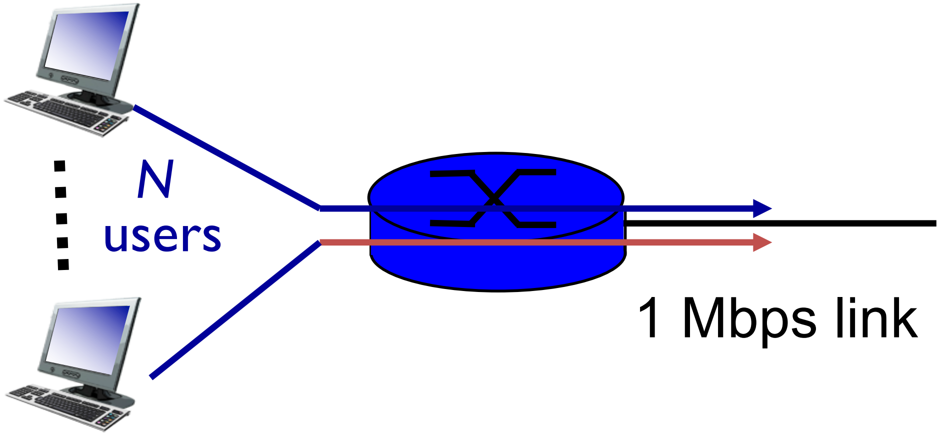
- Packet switching vs circuit switching
  - Pros and cons of each switching method

Packet switching versus circuit switching

packet switching allows more users to use network!

example:

- 1 Mb/s link
- each user:
  - 100 kb/s when "active"
  - active 10% of time



- *circuit-switching:*
  - 10 users
- *packet switching:*
  - with 35 users, probability > 10 active at same time is less than .0004 *

Q: how did we get value 0.0004?

Q: what happens if > 35 users ?

* Check out the online interactive exercises for more examples

Binomial Probability Distribution

- A fixed number of observations (trials), n
  - e.g., 20 tosses of a coin
- Binary random variable
  - e.g., Head or tail in coin toss
  - Often called as success or failure
  - Prob of success is p, and prob of failure is 1-p
- Constant probability for each observation

Binomial example

- Take the example of 5 coin tosses

- What’s the probability that you flip exactly 3 heads in 5 coin tosses?

Binomial distribution

- Solution:
- One way to get exactly 3 heads: HHHTT

- What's the probability of this exact arrangement?
  - P(heads) x P(heads) x P(heads) x P(tails) x P(tails) $= (1/2)^3 \times (1/2)^2$
- Another way to get exactly 3 heads: THHHT
  - Probability of this exact outcome $= (1/2) \times (1/2)^3 \times (1/2)$ $= (1/2)^3 \times (1/2)^2$

Binomial distribution

- In fact, $(1/2)^3$ x $(1/2)^2$ is the probability of each unique outcome that has exactly 3 heads and 2 tails
- So, the overall probability of 3 heads and 2 tails is: $(1/2)^3$ x $(1/2)^2$ + $(1/2)^3$ x $(1/2)^2$ + $(1/2)^3$ x $(1/2)^2$ + ….. for as many unique arrangements as there are
- But how many are there??

$\binom{5}{3}$ ways to arrange 3 heads in 5 trials

Outcome	Probability
THHHT	$(1/2)^3$ *x* $(1/2)^2$
HHHTT	$(1/2)^3$ *x* $(1/2)^2$
TTHHH	$(1/2)^3$ *x* $(1/2)^2$
HTTHH	$(1/2)^3$ *x* $(1/2)^2$
HHTTH	$(1/2)^3$ *x* $(1/2)^2$
THTHH	$(1/2)^3$ *x* $(1/2)^2$
HTHTH	$(1/2)^3$ *x* $(1/2)^2$
HHTHT	$(1/2)^3$ *x* $(1/2)^2$
THHTH	$(1/2)^3$ *x* $(1/2)^2$
HTHHT	$(1/2)^3$ *x* $(1/2)^2$

10 arrangements *x* $(1/2)^3$ *x* $(1/2)^2$

The probability of each unique outcome (note: they are all equal)

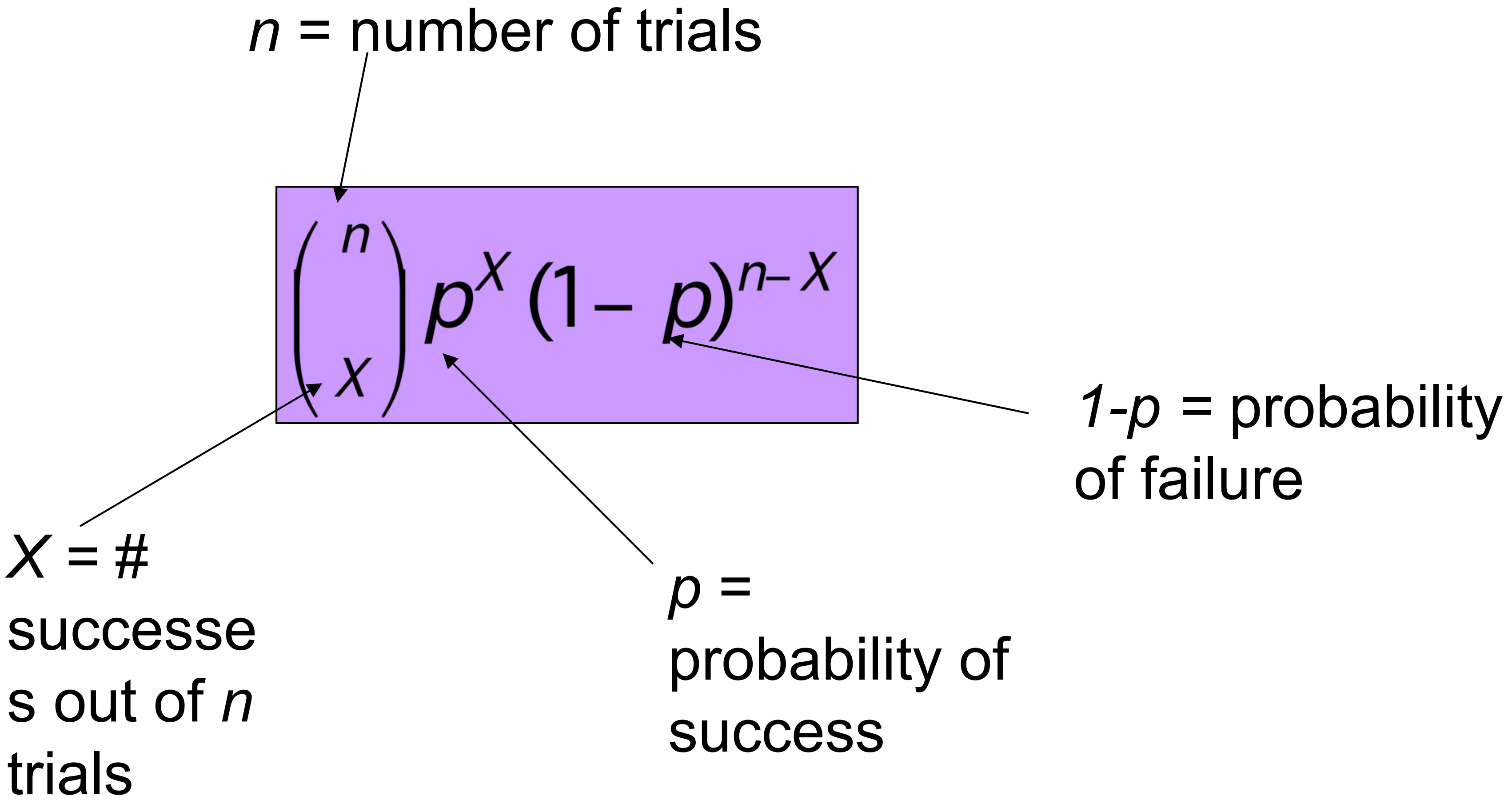
$${}_5C_3 = 5!/3!2! = 10$$

$\therefore$P(3 heads and 2 tails) = $\binom{5}{3}$ ***x P(heads)^{3} x P(tails)^{2}***

= 10 x $(\frac{1}{2})^{5=}$31.25%

Binomial distribution, generally

Note the general pattern emerging → if you have only two possible outcomes (call them 1/0 or yes/no or success/failure) in *n* independent trials, then the probability of exactly *X* “successes”=



n = number of trials

$$\binom{n}{X} p^{X} (1-p)^{n-X}$$

1-p = probability of failure

X = # successes out of *n* trials

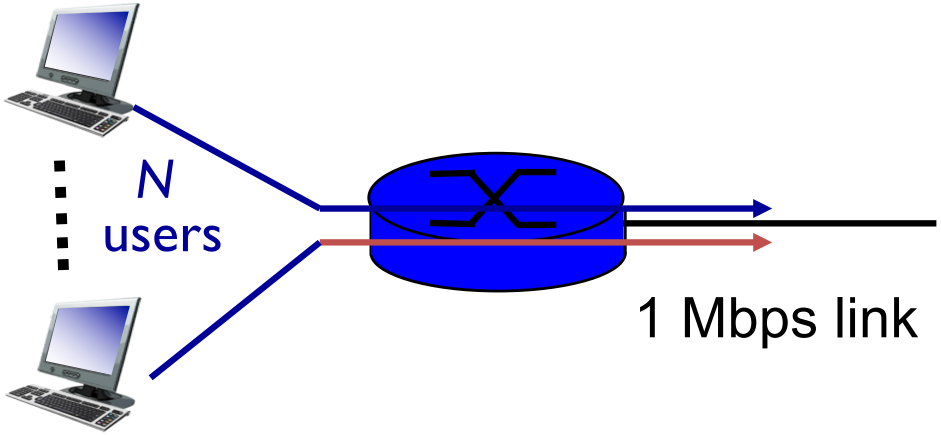
p = probability of success

Packet switching versus circuit switching

packet switching allows more users to use network!

example:

- 1 Mb/s link
- each user:
  - 100 kb/s when "active"
  - active 10% of time



- *circuit-switching:*
  - 10 users
- *packet switching:*
  - with 35 users, probability > 10 active at same time is less than .0004 *

Q: how did we get value 0.0004?

Q: what happens if > 35 users ?

* Check out the online interactive exercises for more examples

- N = 35 users
- Prob (# active users > 10) = 1 – Prob (#active = 10)

  – Prob (#active = 9)

  – Prob (#active = 8)

  …

  – Prob (#active = 0)

- Prob (#active = 10) = C(35, 10) * 0.1^{10} x 0.9^{25}

Application layer

- Client-server vs P2P
- Transport service requirements depending on applications
- TCP service vs UDP service

- Web and HTTP
  - non-persistent vs persistent HTTP
  - Response times
  - HTTP Cookie
  - Web caching

Application layer

- DNS
  - The working mechanism of DNS
- P2P
  - File distribution efficiency: client-server vs P2P
  - BitTorrent working mechanism

Transport layer

- Demultiplexing
  - Connectionless
  - Connection-oriented
- Reliable transport protocols
  - Stop-and-wait, Go-back-N, Selective Repeat
- TCP
  - slow start
  - fast retransmit
  - connection establishment
  - congestion control: AIMD
  - flow control
  - fairness

Network layer

- Longest Prefix Matching
- Router architecture
- Subnet: concepts
- Hierarchical addressing
- Understanding of DHCP, NAT, ICMP and IPv6

Network layer

- Routing algorithm
  - Link State algorithm: Dijkstra's algorithm
  - Distance Vector algorithm: Bellman-Ford algorithm
- Hierarchical routing
- Understanding of RIP, OSPF and BGP

Link layer

- Multiple access protocols
  - channel partitioning (TDMA, FDMA)
  - random access (Slotted ALOHA, Pure ALOHA, CSMA, CSMA/CD)
  - "taking turns" (polling, token passing)
- MAC address and ARP
- Ethernet
  - Switch self-learning mechanism
- Switch vs. Router
- Error detection (e.g., CRC)

Multimedia networking

- Streaming stored video
- Streaming live video
- Content-Distribution Network
- Protocols for real-time interactive applications
  - RTP
  - SIP
  - H.323
- Network support for multimedia applications
  - DiffServ and IntServ