

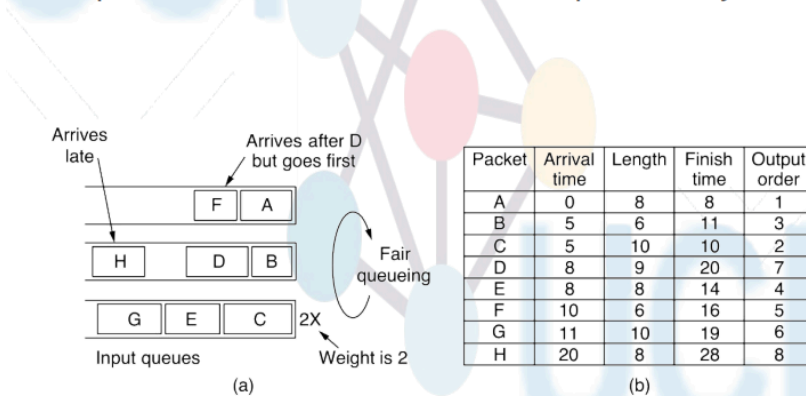
Tema de Congestión y QoS

Queueing exercise

Consider the packet queues shown in Fig. 5-29

What is the finish time and output order of the packets if the **middle** queue, instead of the bottom queue, has a weight of 2?

Order packets with the same finish time alphabetically



Peso por cola:

- Arriba (F, A) → 1
- Medio (H, D, B) → 2
- Abajo (G, E, C) → 1

| Packet | Length | Finish time | Output order |
|--------|--------|--------------------------------|--------------|
| A | 8 | $\max(0, 8) + 8/1 = 16$ | 1 |
| B | 6 | $\max(16, 6) + 6/2 = 19$ | 2 |
| C | 10 | $\max(19, 10) + 10/1 = 29$ | 3 |
| D | 9 | $\max(29, 9) + 9/2 = 33.5$ | 4 |
| E | 8 | $\max(33.5, 8) + 8/1 = 41.5$ | 5 |
| F | 6 | $\max(41.5, 6) + 6/1 = 47.5$ | 6 |
| G | 10 | $\max(47.5, 10) + 10/1 = 57.5$ | 7 |
| H | 8 | $\max(57.5, 8) + 8/2 = 61.5$ | 8 |

Token bucket exercise

A computer on a 6-Mbps network is regulated by a token bucket. The token bucket is filled at a rate of 1 Mbps. It is initially filled to capacity with 8 megabits.

How long can the computer transmit at the full 6 Mbps?

Ancho de banda = 6 Mbps

El balde de tokens se llena a una velocidad de 1 Mbps

El balde se llena al inicio con 8 Mb

Se sacan tokens a una velocidad de 6-1 Mbps

$$8 \text{ Mb} / (6 - 1) \text{ Mbps} = 1.6 \text{ s}$$

Entonces, puede transmitir 1.6 segundos en 6 Mbps.

Tema de Transporte

6-15) In Figure 6-20, suppose that a new flow E is added that takes a path from R1 to R2 to R6. How does the max-min bandwidth allocation change for the five flows?

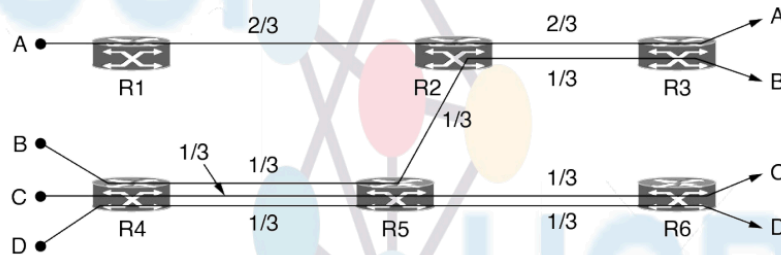
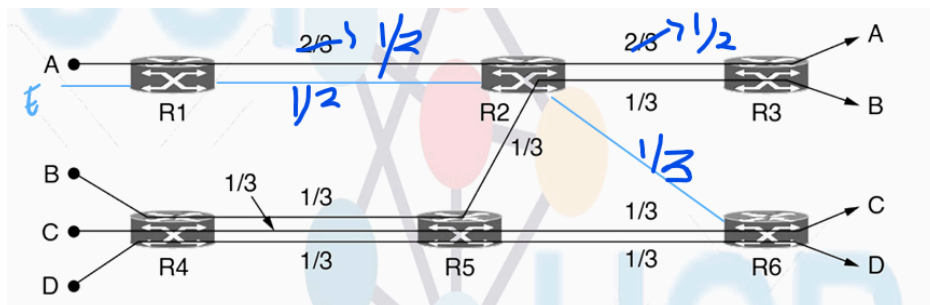


Figure 6-20. Max-min bandwidth allocation for four flows.



6-20) Two hosts simultaneously send data through a network with a capacity of 1 Mbps. Host A uses UDP and transmits a 100 bytes packet every 1 msec. Host B generates data with a rate of 600 kbps and uses TCP. Which host will obtain higher throughput?

$100 \text{ bytes} / 1 \text{ msec} \times 10^3 / 1 \text{ msec} = 100000 \text{ bytes} / \text{segundo}$
 $100000 \text{ bytes} / 1 \text{ segundo} \times 8 \text{ bits} / 1 \text{ byte} = 800000 \text{ bits} / \text{segundo}$
 $800000 \text{ bits} / 1 \text{ segundo} \times 1 \text{ Mbit} / 10^6 \text{ bits} = 0.8 \text{ Mbps} \rightarrow \text{Host A}$
 $600 \text{ kb} / 1 \text{ segundo} \times 1 \text{ Mb} / 10^3 \text{ Kb} = 0.6 \text{ Mbps} \rightarrow \text{Host B}$

El host A tiene mayor throughput entre los 2 ya que $0.8 > 0.6 \text{ Mbps}$.

Tanto UDP como TCP utilizan números de puerto para identificar la entidad de destino al entregar un mensaje. Dé dos razones por las que estos protocolos inventaron un nuevo ID abstracto (números de puerto), en lugar de usar ID de proceso, que ya existían cuando se diseñaron.

Los PID son específicos del sistema operativo, cambian cada vez que un proceso se ejecuta, mientras los puertos son una abstracción a nivel de red, estandarizados. Esto permite que una aplicación se conecte a un servicio conocido, sin saber qué PID tiene el proceso que lo ejecuta.

Además, los PID son locales en la PC, pero los puertos funcionan en la red, TCP/UDP lo tienen en el encabezados de los paquetes para que el sistema operativo del receptor sepa a qué socket o proceso entregarle los datos.