End to end delay

Theoretical exercises

Setting variables

In [7]:

```
link1_capacity = 60_000_000
link2_capacity = 25_000_000
link3_capacity = 20_000_000_000

link1_length = 15
link2_length = 250
link3_length = 10_000

link1_velocity = 300_000_000
link2_velocity = 200_000_000
link3_velocity = 250_000_000
```

1) Determine the propagation delay and the transmission delay for the 3 links for packets of size 1500 bytes.

In [8]:

```
package_size = 1_500 * 8

link1_propagation_delay = package_size / link1_capacity
link2_propagation_delay = package_size / link2_capacity
link3_propagation_delay = package_size / link3_capacity

print("The propagation delay for Link 1 is " + \\
      str(link1_propagation_delay * 1000_000) + " micro seconds")
print("The propagation delay for Link 2 is " + \\
      str(link2_propagation_delay * 1000_000) + " micro seconds")
print("The propagation delay for Link 3 is " + \\
      str(link3_propagation_delay * 1000_000) + " micro seconds")
```

```
The transmission delay for Link 1 is 200.0 micro seconds
The transmission delay for Link 2 is 480.0 micro seconds
The transmission delay for Link 3 is 0.6 micro seconds
```

In [9]:

```
link1_transmission_delay = link1_length / link1_velocity
link2_transmission_delay = link2_length / link2_velocity
link3_transmission_delay = link3_length / link3_velocity

print("The transmission delay for Link 1 is " + \\
      str(round(link1_transmission_delay * 1_000_000, 3)) + " micro secon
print("The transmission delay for Link 2 is " + \\
      str(link2_transmission_delay * 1_000_000) + " micro seconds")
print("The transmission delay for Link 3 is " + \\
      str(link3_transmission_delay * 1_000_000) + " micro seconds")
```

```
The propagation delay for Link 1 is 0.04999999999999996 micro seconds
The propagation delay for Link 2 is 1.25 micro seconds
The propagation delay for Link 3 is 40.0 micro seconds
```

2) Determine the logical bus length of the three links. The logical bus length is the number of packets (or the fraction of a packet) that can be simultaneously present on the bus.

In [10]:

```
link1_duration = link1_transmission_delay / package_size
link2_duration = link2_transmission_delay / package_size
link3_duration = link3_transmission_delay / package_size

link1_logical_length = link1_propagation_delay / link1_duration
link2_logical_length = link2_propagation_delay / link2_duration
link3_logical_length = link3_propagation_delay / link3_duration

print("The logical bus length for Link 1 is " + \\
      str(link1_logical_length / package_size) + " packages")
print("The logical bus length for Link 2 is " + \\
      str(link2_logical_length / package_size) + " packages")
print("The logical bus length for Link 3 is " + \\
      str(link3_logical_length / package_size) + " packages")
```

```
The logical bus length for Link 1 is 0.00025 packages
The logical bus length for Link 2 is 0.002604166666666667 packages
The logical bus length for Link 3 is 66.66666666666667 packages
```

3) Determine the end-to-end delay for the transmission of a packet over the 3 links in the order of Link 1-Link 2-Link 3. Does the end-to-end delay depend on the order of the links?

In [11]:

```
end_to_end_delay = \\
    link1_propagation_delay + link1_transmission_delay \\
    + link2_propagation_delay + link2_transmission_delay \\
    + link3_propagation_delay + link3_transmission_delay

print("The end to end delay is " + \\
      str(end_to_end_delay * 1_000_000) + " micro seconds")
```

```
The end to end delay is 721.9000000000001 micro seconds
```

If we only have a single packet and each link can handle the packet independently without queuing delays, then the end-to-end delay would simply be the sum of the individual delays for each link, and the order would not matter.

4) Now consider a packet burst of 20 packets, meaning 20 packets are transmitted directly one after another. What is the total transmission duration for this packet burst if the links are transferred in the order of Link 1-Link 2-Link 3? In this case, does the total transmission duration depend on the order of the links?

In [12]:

```
total_transmission_delay = \\
    end_to_end_delay + (20 - 1) * link2_propagation_delay

print("The total transmission duration for this packet burst is " \\
      + str(total_transmission_delay * 1_000) + "ms")
```

```
The total transmission duration for this packet burst is 9.841899999999999
9ms
```

Even with the packet burst of 20 packets, the order of the links has no influence on the total duration, as it only depends on the slowest link.