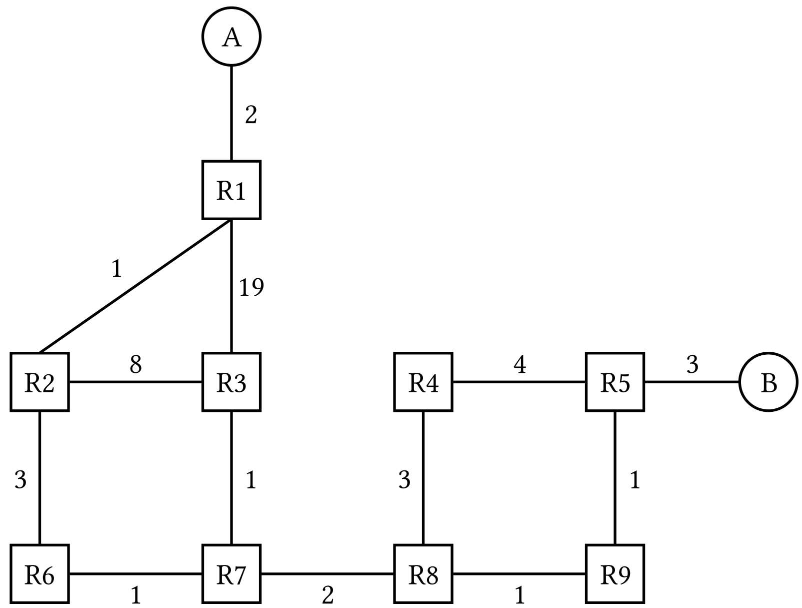
1 Link-State Routing



For this problem, assume the network is running a link-state routing protocol, minimizing total route latency. The following questions indicate events that happen consecutively.

1.1 After convergence, what route does **R7** think its packet will take to **Host B**?

R7 ⟶ R8 ⟶ R9 ⟶ R5 ⟶ B

EVENT: *The R8-to-R9 link goes down.*

1.2 **R8** and **R9** have recomputed their routes but have not yet sent updates to other routers. What route does **R7** think its packet will take to **Host B**?

Same, R7 ⟶ R8 ⟶ R9 ⟶ R5 ⟶ B

1.3 What route does it actually take?

R7 ⟶ R8 ⟶ R4 ⟶ R5 ⟶ B

1.4 Assume all nodes are now aware of the new network state and have recomputed their routes. What route does a packet take from **R3** to **Host A**?

R3 ⟶ R7 ⟶ R6 ⟶ R2 ⟶ R1 ⟶ A

EVENT: *The cost of the R1-to-R2 link increases to 100.*

1.5 **R2** and **R1** recompute their routes but have not yet sent updates to other routers. What route does **R2** think its packet will take to **Host A**?

R2 ⟶ R6 ⟶ R7 ⟶ R3 ⟶ R1 ⟶ A

1.6 What route does it actually take?

R2 ⟶ R6 ⟶ R2 *(loop)*

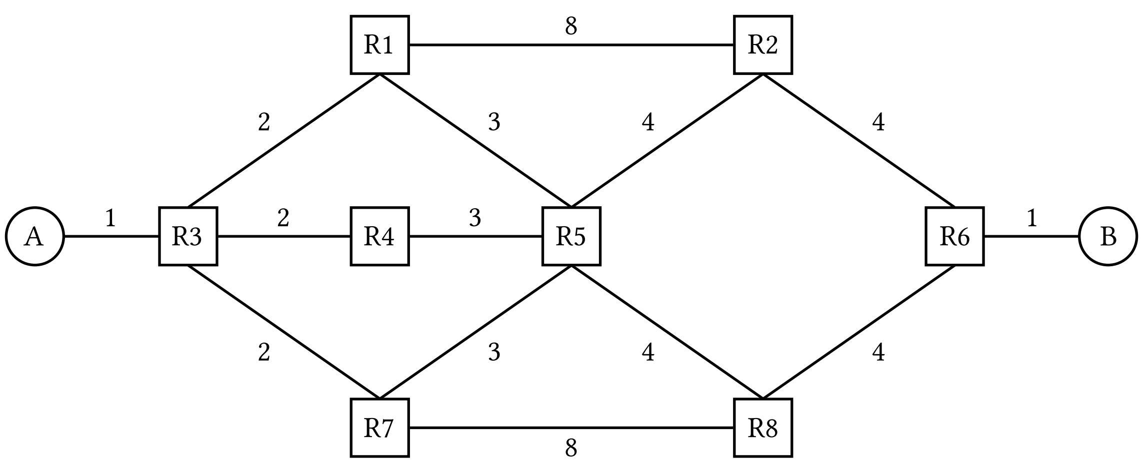
1.7 Which additional routers must receive the routing updates and recompute their routes for all routers to be able to successfully send packets to **Host A**?

R3, **R6**, **R7**

1.8 All routers except **R3** have received the routing updates and recomputed their routes. Which routers can successfully send packets to **Host A**?

None of the routers (except **R1**)

2 More L3 Link State



2.1 After convergence, what is the path cost from **A** to **B**, and what are all the possible paths with this cost?

The cost is 15. All paths are A ⟶ R3 ⟶ [R1, R4, R7] ⟶ R5 ⟶ [R2, R8] ⟶ R6 ⟶ B.

2.2 Suppose that a control message (a message used by the routing algorithm) takes 1 second to propagate along a link, regardless of link cost. What individual link failure inside the network would cause the longest delay to reconvergence, and what is that delay?

A link **R3-to-R4** failure would cause a 3-second delay. The last node that learns about the failure is node **R6**, and the update must traverse through at least 3 links to get to **R6**, which will take 3 seconds. Any other link failure within the network can propagate to every other node in less than 3 seconds. Host links could also take 3 seconds to propagate.

2.3 Suppose you have the ability to take down individual nodes. Which nodes would you take down in order to partition the network? If you can't partition the network, which nodes would you take down to increase path costs from **A** to **B** maximally?

In each part, suppose you can take down:

(a) A single node (excluding **R3** and **R6**).

R5. Can't partition, and removing **R5** increases path costs the most (by 1).

(b) Two nodes (excluding **R3** and **R6**).

R2 and **R8**. Removing them will partition **R6** from the network.

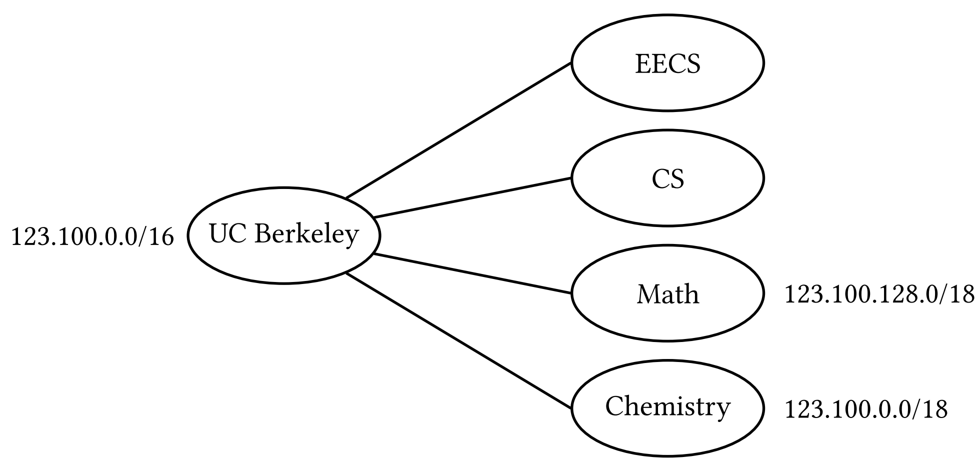
2.4 Which single link's cost (if any) should you double in order to increase the path cost from **A** to **B**?

Doubling the cost of **A-R3** or **R6-B** will increase the total path cost by 1. It's not possible with any other link, since there is always an alternate, cheaper path (see part 1).

3 IP Addressing

Suppose **berkeley.edu** is the Provider AS for EECS, CS, Math, and Chemistry, and needs to assign IPv4 addresses to them.

Assume that the CIDR (Classless InterDomain Routing) addressing scheme is used, and that **berkeley.edu** has the `123.100.0.0/16` prefix reserved.



3.1 Which addresses are included in the Math department's prefix? How many addresses are in this range?

Math's address prefix, in binary (with the prefix **bolded**), is:

01111011 . 01100100 . 10 000000 . 00000000

The range of allowed addresses is thus:

01111011 . 01100100 . 10 000000 . 00000000

...

01111011 . 01100100 . 10 111111 . 11111111

which in decimal is **[123.100.128.0, 123.100.191.255]**.

If Math's prefix is 18 bits long, and IPv4 addresses are 32 bits long, then there are **32 - 18 = 14** bits available to uniquely identify hosts, so this prefix contains $\mathbf{2^{14}}$ addresses.

3.2 `123.100.192.0/18` is reserved for EECS and CS. Assign equal halves of this address space to the two departments.

This address space has the prefix: **01111011 . 01100100 . 11** 000000 . 00000000

To divide this `/18` prefix in two, we can assign one `/19` prefix to each department:

EECS: 01111011 . 01100100 . 11 **0** 00000 . 00000000 **123.100.192.0/19**

CS: 01111011 . 01100100 . 11 **1** 00000 . 00000000 **123.100.224.0/19** (or vice versa)

3.3 You want to start a new department, Floriology, and assign it an unused address range. You foresee that no more than 50 people will enroll. Assuming one address per person, what prefix would you assign to it?

A **64-address prefix** will hold enough addresses for Floriology, which requires **6 bits** for hosts. Thus, we need a **(32 - 6)** = **/26** prefix.

We have to assign a prefix that is currently unclaimed. If EECS/CS claimed **123.100.192.0/18**, Math claimed **123.100.128.0/18**, and Chemistry claimed **123.100.0.0/18**, then only the **123.100.64.0/18** prefix is unclaimed.

Any `/26` prefix in **123.100.64.0/18** will work, such as **123.100.64.0/26**.