ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Distributed Computing

FS 2018 Prof. R. Wattenhofer

Computer Engineering II

Exercise Sheet Chapter 2

We categorize questions into four different categories:

Quiz Short questions which we will solve rather interactively at the start of the exercise sessions.

Basic Improve the basic understanding of the lecture material.

Advanced Test your ability to work with the lecture content. This is the typical style of questions which appear in the exam.

Mastery Beyond the essentials, more interesting, but also more challenging. These questions are **optional**, and we do not expect you to solve such exercises during the exam.

Questions marked with $^{(g)}$ may need some research on Google.

Quiz

1 Quiz Questions

a) What is the weight of the shortest path from s to t in the following graph?

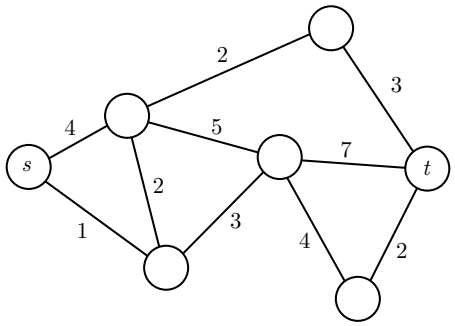


Figure 1: A simple graph.

b) What is the total cost of the MST of the graph in Figure 1?

$^{(g)}$**c)** Can I assign the IPv4 address `127.0.0.1` to a computer in my network?

$^{(g)}$**d)** How many IPv6 addresses can a single network interface have?

$^{(g)}$**e)** What are use-cases in which a single computer might advertise multiple addresses on a single network interface?

Basic

2 Basic IP Networking

You have bought a new "home router" to connect your flat to the Internet, and you connected your PC to the "home router".

(g) **a)** We wrote the term "home router" with quotation marks. How does the device which is colloquially called "[home] router" differ from an actual router as discussed in the lecture?

(g) **b)** How can you determine the IPv4 address assigned to the network interface card of your PC?

(g) **c)** You know that the "home router" can be configured using a web interface. How can you determine the IPv4 address of the "home router" to which you need to connect to from your PC?

(g) **d)** Everyone seems to be using the same boring `192.168.0.*` addresses for their local network. Is it a good idea to be more fancy and choose the nicer address space `8.8.8.*` for your home network?

(g) **e)** You establish a connection to a remote server on the Internet. Will the server see the IPv4 address of your PC (answer to **c)**) as the sender address? If not, which address will the server see, and how can you determine this address?

3 Using an SSH Tunnel (g)

In some environments you might not be able to connect to the services that you want to use. However, it might be possible that SSH traffic is not blocked. In those cases you can setup an SSH tunnel to a server (which you can access using SSH), and tunnel your traffic to that server to bypass the restrictions.

Setup an SSH tunnel via `slab1.ethz.ch` such that you can browse the web using that tunnel. Note that you might need to change the settings in your browser to use the local SOCKS proxy.

Check if the tunnel works, by checking your public IP address, e.g. using a website like `https://www.whatismyip.com/`.

Advanced

4 Simple Routing

You have to implement a routing solution for the given tree graph. As you are in charge for the routing, you managed to get the permission to assign addresses to nodes in your network in any way you want. As in most networks, packets can be sent in both directions over each edge. Hence, if there is an edge between two nodes u and v, we say that there is an *outgoing edge* (u, v) from u, and an outgoing edge (v, u) from v. Thus, a node with degree 3 has three outgoing edges.

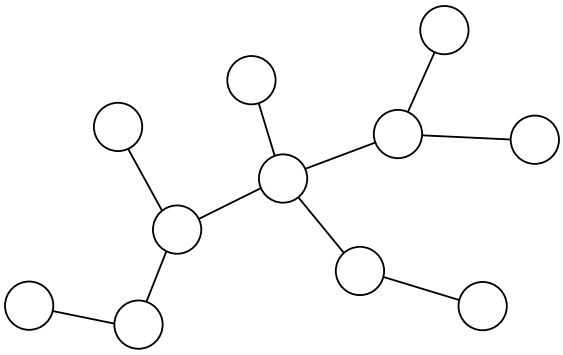
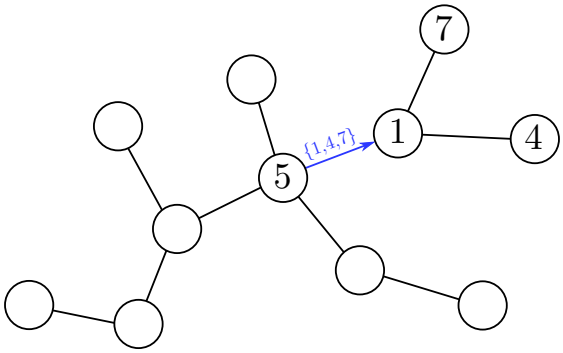


Figure 2: A tree network.

Your task is to come up with an assignment of addresses to nodes in such a way, that you can store the information *where* the other nodes are *efficiently* on each node (i.e., that you have simple routing tables on each node). Since the degrees are limited, you decide that you want to store for each outgoing edge the information about which nodes that can be reached using this edge.

If you assign addresses randomly, it can happen that you have to store many addresses on a single edge. In the following example, Node 5 needs to store the three individual addresses $\{1, 4, 7\}$ for the blue edge.



Depending of the number of nodes which are reachable through an edge, this list could grow massively. It would therefore be a lot more efficient to assign the addresses in such a way, that the only information which needs to be stored for an outgoing edge is an interval, e.g.: "this edge leads to all addresses in $[4, 7]$". However, if you need a lot of intervals per edge, nothing is gained.

a) Find an assignment of addresses from 1 to 11 to the nodes in Figure 2 in such a way, that you need to store at most two intervals on each outgoing edge.

b) Develop an algorithm (high-level description or pseudocode) which, for any given tree, creates an assignment of addresses to nodes such that every node needs to store at most two intervals for each edge.

5 Routing Loops

Each node in a network knows how to forward incoming packets using the routing table. If a link fails, it possible that (temporarily) packets can not be delivered even though there still exists a physical path between the two nodes. The packets get sent back and forth between two or more nodes. E.g., v_1 forwards a packet to v_2 who then forwards it again to v_1, etc. A "Short-Term Routing Loop" (STRL) has formed. The directly neighboring nodes see the failure of a link or of a node immediately.

a) The RIP protocol gets used in networks with few nodes. After the failure of a link, it can take up to a few minutes until a node knows about the change in the network. Why does it take so long?

b) How can a node that is part of the STRL detect the STRL?

c) When does an STRL get resolved in a link-state protocol?

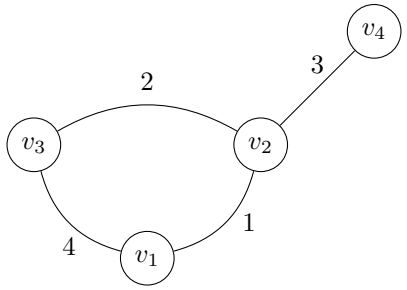


Figure 3: Network with four nodes. Values indicate the costs of the links.

d) Show how an STRL can form in the graph in Figure 3 when a single link fails while the graph is still connected. Give the routing tables of the nodes that are part of the STRL immediately after the failure of the link. Assume that the routing table has been created using a link-state protocol.

e) Can an STRL also form when a node fails in a network? If no, explain why. If yes, give an example graph.