Assignment 2

Question 1

a. For the 3 hosts we can choose any figure from 0-24 as the port, provided that each host has a different port.
Host 1: 192.168.1.0
Host 2: 192.168.1.1
Host 3: 192.168.1.2

b. To build the NAT translation table we need the endpoints of each TCP connection. In the example, one endpoint is the home network (hosts) on the LAN side and the other endpoint is sent as a destination address to the ISP.

NAT Translation Table	
WAN side	LAN side
128.120.41.85, 80	192.168.1.0
128.120.41.85, 80	192.168.1.0
128.120.41.85, 80	192.168.1.1
128.120.41.85, 80	192.168.1.1
128.120.41.85, 80	192.168.1.2
128.120.41.85, 80	192.168.1.2

Question 2

Note that for this three-node topology, the value of the costliest path (5) is less than the combined cost of the other 2 paths (4+2). The least costly path will always consist of 1 hop.

Node N1 Table

	Cost to		
From	N1	N2	N3
N1	0	4	5
N2	∞	∞	∞
N3	∞	∞	∞

	Cost to		
From	N1	N2	N3
N1	0	4	5
N2	4	0	2
N3	5	2	0

Node N2 Table

	Cost to		
From	N1	N2	N3
N1	∞	∞	∞
N2	4	0	2
N3	∞	∞	∞

	Cost to		
From	N1	N2	N3
N1	0	4	5
N2	4	0	2
N3	5	2	0

Node N3 Table

	Cost to		
From	N1	N2	N3
N1	∞	∞	∞
N2	∞	∞	∞
N3	5	2	0

	Cost to		
From	N1	N2	N3
N1	0	4	5
N2	4	0	2
N3	5	2	0

Question 3

Dijkstra's Algorithm

N' = the subset of nodes with known least-cost path.
D(n) = current least-cost path from source to destination *n*.
c(*w,n*) = link cost from node *w* to node *n*.
p(n) = previous node along current last-cost path from source to *w*.

Initialise:
N' = {*w*}
for each node *n*
    if *n* adjacent to *w*, *D(n)* = *c(w,n)*
    else, *D(n)* = ∞

Loop:
Until all nodes in *N'*:
    Find node *o* with minimum *D(o)* and add that to *N'*.
    Update *D(n)* for each node *n* adjacent to *o* and not in *N'*.
        D(n) = *min(D(n), D(o) + c(n,o))*

Table of Results

Step	N'	*D(x), p(x)*	*D(u), p(u)*	*D(v), p(v)*	*D(t), p(t)*	*D(y), p(y)*	*D(z), p(z)*
0	w	**4,w**	**1,w**	8,w	∞	∞	∞
1	wu			**5,u**	**8,u**	∞	∞
2	wux					**7,x**	**5,x**
2	wuxv						
3	wuxvz						
4	wuxvzy						
5	wuxvzyt						

Forwarding Table in *w*

Destination	**Link**
u	(*w, u*)
x	(*w, x*)
v	(*w, u*)
z	(*w, x*)
y	(*w, x*)
t	(*w, u*)

Question 4

a. The suffixes of the IP addresses of the interfaces can be arbitrarily chosen, as long as they are unique.

Subnet 1:
Router -> Subnet 2 = 101.101.101.001
A = 101.101.101.002
B = 101.101.101.003

Subnet 2:
Router -> Subnet 1 = 102.102.102.001
C = 102.102.102.002
D = 102.102.102.003
Router -> Subnet 3 = 102.102.102.004

Subnet 3:
Router -> Subnet 2 = 103.103.103.001
E = 103.103.103.002
F = 103.103.103.003

b. MAC addresses are 48-bits long – 6 groups of 2 hexadecimal digits. They can also be arbitrarily chosen, as long as they are unique.

Subnet 1:
Left Router – Subnet 1-side = 11-11-11-AA-AA-AA
Host A = 11-11-11-BB-BB-BB
Host B = 11-11-11-CC-CC-CC

Subnet 2:
Left Router – Subnet 2-side = 22-22-22-AA-AA-AA
Host C = 22-22-22-BB-BB-BB
Host D = 22-22-22-CC-CC-CC
Right Router – Subnet 2-side = 22-22-22-DD-DD-DD

Subnet 3:
Right Router – Subnet 3-side = 33-33-33-AA-AA-AA
Host E = 33-33-33-BB-BB-BB
Host F = 33-33-33-CC-CC-CC

c. From host B to host F:

(i) From host B to left router:

MAC addresses
Source: 11-11-11-CC-CC-CC
Destination: 11-11-11-AA-AA-AA

IP addresses
Source: 101.101.101.003
Destination: 101.101.101.001

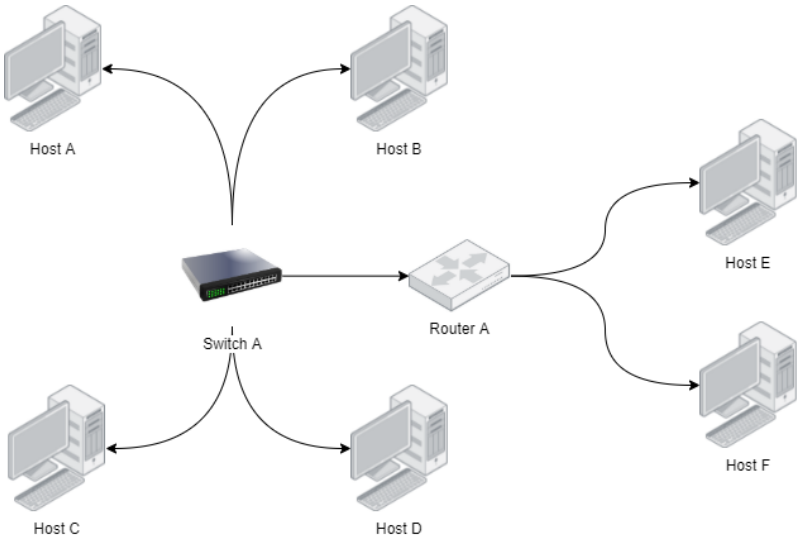
(ii) From left router to right router:

MAC addresses

Source: 22-22-22-AA-AA-AA
Destination: 22-22-22-DD-DD-DD
IP addresses
Source: 102.102.102.001
Destination: 102.102.102.004

(iii) From right router to host F:
MAC addresses
Source: 33-33-33-AA-AA-AA
Destination: 33-33-33-CC-CC-CC
IP addresses
Source: 103.103.103.001
Destination: 103.103.103.003

d. Switches and routers work similarly as both use a store-and-forward model. However, switches are used as a bridge rather than a defined endpoint. They lack IP addresses and operate entirely using MAC addresses.
Updating the model diagram gives us:



Assuming all MAC and IP address remain the same:

(iv) Host B to router:
MAC addresses
Source: 11-11-11-CC-CC-CC
Destination: 22-22-22-DD-DD-DD
IP addresses
Source: 101.101.101.003
Destination: 102.102.102.004

(v) Router to Host F:
MAC addresses
Source: 33-33-33-AA-AA-AA
Destination: 33-33-33-CC-CC-CC
IP addresses
Source: 103.103.103.001
Destination: 103.103.103.003

Question 5

Physical Layer

The first step is to ensure the host PC is connected to a network. In this case we use an Ethernet cable to connect to the router.

From now the host PC will be referred to as the **client** and the owner of the Webpage will be the **server**.

The host requires a Media Access Control (MAC) address to receive frames/datagrams from other hosts, and an Internet Protocol (IP) address to send and receive requests.

When the host joins the network, it will be dynamically assigned an IP address using the Dynamic Host Configuration Protocol (DHCP). This process consists of 4 messages. Note that if a DHCP server is already known to the host, the first 2 steps can be skipped:

- *DHCP discover:* The host searches for a DNCP server
- *DHCP offer:* A valid DNCP server replies with an offered IP address.
- *DHCP request:* The host requests the address.
- *DHCP ack:* The server sends the address.

MAC addresses will NOT be generated with DHCP, but they can be manually added.

It is important to note that datagrams sent from a local network will have the same IP address. Individual devices are effectively invisible to foreign networks, which is a security boon. The network router/switch will direct responses to the appropriate host.

Before data is exchanged between the endpoints, a connection is established using a transport protocol. The TCP protocol is used for HTTP requests/responses.
First, the client sends a connection request. If the address is accurate and the request is acknowledged, the server will return an "accept" notification to the client. Then the client will send the HTTP request.

The client generates the HTTP request. This request contains information such as the destination address, the request type (GET, POST, PUT, etc.)

The user will input the address of the web page they want to download. For this to be usable, the address name will need to be converted to an IP address. This is where the Domain Name System (DNS) comes into play. The DNS is an application-layer protocol that translates the address entered by humans into the IP address used for address datagrams. It does this by searching server-by-server, either through an iterated or recursive query. If the correct server is tracked down, the IP address of that server is added to the request.

The request will be sent to the destination address through the router/switch. The server will generate a HTTP response containing, amongst other things, the source/destination addresses, response status (such as Success-200, Forbidden-401, etc) and the audio/visual/textual objects that make up the web page.

The response is passed back to the network router. The server containing the webpage does not know the exact port of the recipient, so the router determines the correct host and

sends the reply to them. The client will now be able to load the webpage if the response was successful.

To acquire the MAC address of a host, a broadcast message is sent to the LAN. Using the Address Resolution Protocol (ARP), an ARP table will be consulted. This table matches MAC addresses to IP addresses, allowing datagrams to be sent from the router/switch to specific hosts.