Communication Networks – HW3
Oz Cabiri & Guy Harem

Question 1 – IP Packets:

1.

Field	Bits (binary of hex)	Description
Version (4)	0100	IPv4 uses 0100
IHL (4)	0101	No additional options – using 5 DWORDS
Total Length (16)	0x03FC	1000 bytes of payload + 20 bytes of IP header – 1020 bytes
Identification (16)	0x0001	Stated in question
Flags (3)	000	Sender is fine with fragmentation
Fragment Offset (13)	0x0000	No fragmentation has occurred
TTL (8)	0x80	Stated in question – 128
Protocol (8)	0x11	Stated in question – UDP
Source address (32)	0xBE1E0A07	190.30.10.7 in Hex
Destination address (32)	0x96490A11	150.73.10.17 in Hex

2. All fields will remain the same except the TTL. The packet has passed 2 routers along the way, so 2 hops will be reduced from the TTL to 126. **New TTL: 0x7E**.
3. The MTU is 500B and the IP header is 20B, so in each packet the payload cannot exceed 480B.
   Since we have 1000B payload it will be fragmented into 3 packets: 480B, 480B, 40B.
   The fields Version, IHL, Identification, Protocol, Source and Destination address will not change. Let's describe the fields that change:

Packet 1:

Field	Bits (binary of hex)	Description
Total Length (16)	0x01F4	480 bytes of payload + 20 bytes of IP header – 500 bytes
Flags (3)	001	Not the last fragment
Fragment Offset (13)	0x0000	1st fragment

Packet 2:

Field	Bits (binary of hex)	Description
Total Length (16)	0x01F4	480 bytes of payload + 20 bytes of IP header – 500 bytes
Flags (3)	001	Not the last fragment
Fragment Offset (13)	0x003C	Offset 480 / 8 = 60 Byte units

Packet 3:

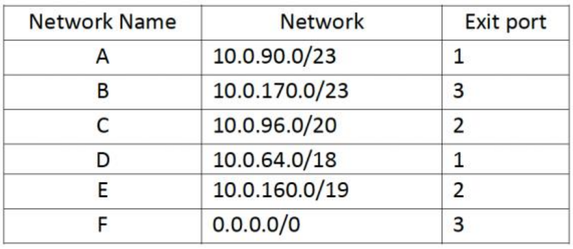
Field	Bits (binary of hex)	Description
Total Length (16)	0x003C	40 bytes of payload + 20 bytes of IP header – 60 bytes
Flags (3)	000	Last fragment
Fragment Offset (13)	0x0078	Offset 960 / 8 = 120 Byte units

TTL for all packets will be *0x80* – $N_{p,i}$ where *p* is the number of packet and *i* is the current hop. This is because with each hop the TTL will decrease by 1 (importance of *i*), and each packet can go through a different route (importance of *p*).

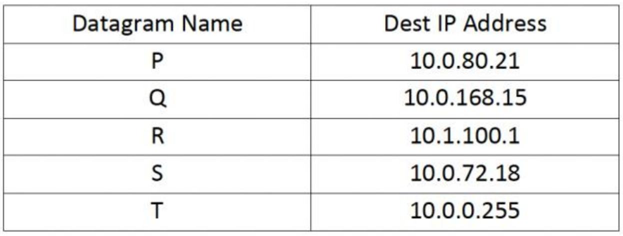
4. By checking the *Identification* field the receiver knows which fragments belong to the same original packet. Now it's all a matter of reconstructing it. When receiving the last fragment (marked with *MF=0*) the receiver can know the entire payload size by checking the *Fragment Offset* field. It can now fill in the 'blanks', by checking each fragment's offset value.
   If no 'blank' spots remain – it received the entire original packet.

5. In order to prevent the packet's fragmentation the sender should have set the *DF* field to 1. When the packet will pass through a router, the router will check if it's size is under the specified *MTU*. If it is, it will forward it, otherwise it will check the *DF* flag.
   In our case the *DF* flag is set, so the router cannot fragment the packet, requiring it to send an ICMP message back to the sender; Specifically, Type 3 Code 4.

Question 2:



Network Name	Network	Exit port
A	10.0.90.0/23	1
B	10.0.170.0/23	3
C	10.0.96.0/20	2
D	10.0.64.0/18	1
E	10.0.160.0/19	2
F	0.0.0.0/0	3



Datagram Name	Dest IP Address
P	10.0.80.21
Q	10.0.168.15
R	10.1.100.1
S	10.0.72.18
T	10.0.0.255

1. The role of line F in table A, is to be the Default Gateway of this network to the internet. Any packet that does not match the networks in the previous lines, would be directed to the default gateway, this line is there to "catch" every other IP that does not fall under any other of the known networks – meaning if a packet is addressed to an IP that is not part of the A-E networks, it would be directed to network F (Default gateway).

2. First, let's calculate all the networks ranges:
A: network mask is 23, so 24-23=1, 2^1 = 2, so the network's range is from 10.0.90.0 up to 10.0.91.255
B: network mask is 23, like A, network's range is 10.0.170.0 up to 10.0.171.255
C: network mask is 20, so 24-20 = 4, 2^4 = 16, so the network's range is from 10.0.96.0 up to 10.0.111.255
D: network mask is 18, so 24-18 = 6, 2^6 = 64, so the network's range is from 10.0.64.0 up to 10.0.127.255
E: network mas is 19, so 24-19 = 5, 2^5 = 32, so the network's range is from 10.0.160.0 up to 10.0.191.255
F: Default Gateway, All ranges

   Now, for the Packets:
Packet P(10.0.80.21), would be addressed to **network D (exit port 1)**, only suitable beside F, longest prefix match (18>0).
Packet Q(10.0.168.15), would be addressed to **network E (exit port 2)**, only suitable beside F, longest prefix match (19>0).
Packet R(10.1.100.1), would be addressed to **network F (exit port 3)**, only suitable network (Default Gateway).
Packet S(10.0.72.18), would be addressed to **network D (exit port 1)**, only suitable beside F, longest prefix match (18>0).
Packet T(10.0.0.255), would be addressed to **network F (exit port 3)**, only suitable network (Default Gateway).

3. Yes, we can delete network A line from the routing table, because the only other network that "contains network A", is being exited to the same port (port 1), so any packet addressed to this network, would be directed to network D, which exists on the same port.

Question 3 – ICMP:

On a given route, the sender initially sends a large packet (larger than the reasonable MTU) and sets the DF flag to 0. When the packet will arrive at a node that has an MTU limit, it will receive an ICMP message (Type 3 Code 4) stating the packet is too large and the maximum MTU.

The sender will then adjust the packet size to MTU/2 and send it again.

If the packet is dropped again the process repeats itself, otherwise, the packet will be received at the destination (by using TCP the sender will be notified) and then the packet size will be adjusted to 3MTU/4.

This is essentially a binary search upon the route's MTU, yielding$O(\log MTU)$messages.

Question 4 – Routing Algorithms:

1. Start of process for A:

Destination	Distance	Next Hop
B	4	B
C	1	C
D	inf	-
E	inf	-
F	inf	-

End of process for A:

Destination	Distance	Next Hop
B	4	B
C	1	C
D	4	C
E	4	C
F	2	C

2. Yes, the algorithm will stabilize. The graph is connected therefore there will be stabilization at some point and no count to infinity will occur.

Destination	Distance	Next Hop
B	4	B
C	1	C
D	4	C
E	5	B
F	2	C

3. The algorithm will not stabilize because the graph is not connected.
   Every node is aware that it's outgoing edges are disconnected – B is aware that (B,E) is deleted and F is aware that (E,F) is deleted.
   After the deletion of (E,F) the distance table of D does not change because it initially reached E through B, but F will update its distance from E to 4.
   After the deletion of (B,E), B notices it cannot reach E and starts looking for an alternative route. Not knowing (E,F) is deleted, it believes it can reach E through D (B>D>F>E) with distance=5.
   Now D looks at the new distances and calculates it can reach E (through B or F) with distance=6.
   B and F receive this new information from D and update their distances to 7 and 8 respectively.
   Since no node is aware of the deletion of both edges, this process will continue indefinitely.

Question 5:

1. From AS5 to any other AS:

   AS1: AS5 → AS2 → AS1
   AS2: AS5 → AS2
   AS3: AS5 → AS2 → AS1 → AS3
   AS4: AS5 → AS4
   AS6: AS5 → AS6

2. From AS1 to any other AS:

   AS2: AS1 → AS2
   AS3: AS1 → AS3
   AS4: AS1 → AS4
   AS5: AS1 → AS4 → AS5 (preferred over AS1 → AS2 → 5 due to BGP policy)
   AS6: AS1 → AS3 → AS6