2. The main function of the data plane is packet forwarding, which is to forward datagrams from their input links to their output links. For example, the data plane's input ports perform physical layer function of terminating an incoming physical link at a router, perform link-layer function to interoperate with the link layer at the other side of the incoming link, and perform lookup function at the input ports.
The main function of the control plane is routing, which is to determine the paths a packet takes from its source to its destination. A control plane is responsible for executing routing protocols, responding to attached links that go up or down, communicating with remote controllers, and performing management functions.

5. The service model of the Internet's network layer is best-effort service. With this service model, there is no guarantee that packets will be received in the order in which they were sent, no guarantee of their eventual delivery, no guarantee on the end-to-end delay, and no minimal bandwidth guarantee.

6. Input port, switching fabric, and output ports are implemented in hardware, because their datagram-processing functionality is far too fast for software implementation. A routing processor inside a traditional router uses software for executing routing protocols, maintaining routing tables and attached link state information, and computing the forwarding table of a router. In addition, a routing processor in a SDN router also relies on software for communication with a remote controller in order to receive forwarding table entries and install them in the router's input ports.
Data plane is usually implemented in hardware due to the requirement of fast processing, e.g., at nanosecond time scale. Control plane is usually implemented in software and operates at the millisecond or second timescale, for example, for executing routing protocols, responding to attached links that go up or down, communicating with remote controllers, and performing management functions.

9. A router uses longest prefix matching to determine which link interface a packet will be forwarded to if the packet's destination address matches two or more entries in the forwarding table. That is, the packet will be forwarded to the link interface that has the longest prefix match with the packet's destination.

13. HOL blocking: Sometimes a packet that is first in line at an input port queue must wait because there is no available buffer space at the output port to which it wants to be forwarded. When this occurs, all the packets behind the first packet are blocked, even if their output queues have room to accommodate them. HOL blocking occurs at the input port.

16. (A typo in the question: different✉ difference)
With RR, all service classes are treated equally, i.e., no service class has priority over any other service class. With WFQ, service classes are treated differently, i.e., each class may receive a differential amount of service in any interval of time. When a WFQ's all classes have the same amount of service weight, the WFQ is identical to RR.

19. No. IP header checksum only computes the checksum of an IP packet's IP header fields, which share no common bytes with the IP datagram's transport-layer segment part.

21. Yes. They have one address for each interface.

25. 50% overhead.

26. Typically the wireless router includes a DHCP server. DHCP is used to assign IP addresses to the 5 PCs and to the router interface. Yes, the wireless router also uses NAT as it obtains only one IP address from the ISP.

27. Route aggregation means that an ISP uses a single prefix to advertise multiple networks. Route aggregation is useful because an ISP can use this technique to advertise to the rest of the Internet a single prefix address for the multiple networks that the ISP has.

28. A plug-and-play or zeroconf protocol means that the protocol is able to automatically configure a host's network-related aspects in order to connect the host into a network.

33. Each entry in the forwarding table of a destination-based forwarding contains only an IP header field value and the outgoing link interface to which a packet (that matches the IP header field value) is to be forwarded. Each entry of the flow table in OpenFlow includes a set of header field values to which an incoming packet will be matched, a set of counters that are updated as packets are matched to flow table entries, and a set of actions to be taken when a packet matches a flow table entry.

35. Three example header fields in an IP datagram that can be matched in OpenFlow 1.0 generalized forwarding are IP source address, TCP source port, and source MAC address. Three fields that cannot be matched are: TTL field, datagram length field, header checksum (which depends on TTL field).

Problem 5

a)

Prefix Match	Link Interface
11100000 00	0
11100000 01000000	1
1110000	2
11100001 1	3
otherwise	3

b)   Prefix match for first address is 5th entry: link interface 3
Prefix match for second address is 3nd entry: link interface 2
Prefix match for third address is 4th entry: link interface 3

Problem 9

Destination Address	Link Interface
200.23.16/21	0
200.23.24/24	1
200.23.24/21	2
otherwise	3

Problem 11

Any IP address in range 128.119.40.128 to 128.119.40.191

Four equal size subnets: 128.119.40.64/28, 128.119.40.80/28, 128.119.40.96/28, 128.119.40.112/28

Problem 14

The maximum size of data field in each fragment = 680 (because there are 20 bytes IP header). Thus the number of required fragments $= \left\lceil \dfrac{2400-20}{680} \right\rceil = 4$

Each fragment will have Identification number 422. Each fragment except the last one will be of size 700 bytes (including IP header). The last datagram will be of size 360 bytes (including IP header). The offsets of the 4 fragments will be 0, 85, 170, 255. Each

Problem 15

MP3 file size = 5 million bytes. Assume the data is carried in TCP segments, with each TCP segment also having 20 bytes of header. Then each datagram can carry 1500-40=1460 bytes of the MP3 file

Number of datagrams required $= \left\lceil \dfrac{5\times 10^6}{1460} \right\rceil = 3425$. All but the last datagram will be 1,500 bytes; the last datagram will be 960+40 = 1000 bytes. Note that here there is no fragmentation – the source host does not create datagrams larger than 1500 bytes, and these datagrams are smaller than the MTUs of the links.

Problem 16

a) Home addresses: 192.168.1.1, 192.168.1.2, 192.168.1.3 with the router interface being 192.168.1.4

b)

NAT Translation Table

WAN Side	LAN Side
24.34.112.235, 4000	192.168.1.1, 3345
24.34.112.235, 4001	192.168.1.1, 3346
24.34.112.235, 4002	192.168.1.2, 3445
24.34.112.235, 4003	192.168.1.2, 3446
24.34.112.235, 4004	192.168.1.3, 3545
24.34.112.235, 4005	192.168.1.3, 3546

Problem 18

It is not possible to devise such a technique. In order to establish a direct TCP connection between Arnold and Bernard, either Arnold or Bob must initiate a connection to the other. But the NATs covering Arnold and Bob drop SYN packets arriving from the WAN side. Thus neither Arnold nor Bob can initiate a TCP connection to the other if they are both behind NATs.

Problem 19

S2 Flow Table	
Match	**Action**
Ingress Port = 1; IP Src = 10.3.*.*; IP Dst = 10.1.*.*	Forward (2)
Ingress Port = 2; IP Src = 10.1.*.*; IP Dst = 10.3.*.*	Forward (1)
Ingress Port = 1; IP Dst = 10.2.0.3	Forward (3)
Ingress Port = 2; IP Dst = 10.2.0.3	Forward (3)
Ingress Port = 1; IP Dst = 10.2.0.4	Forward (4)
Ingress Port = 2; IP Dst = 10.2.0.4	Forward (4)
Ingress Port = 4	Forward (3)
Ingress Port = 3	Forward (4)

Problem 20

S2 Flow Table	
Match	**Action**
Ingress Port = 3; IP Dst = 10.1.*.*	Forward (2)
Ingress Port = 3; IP Dst = 10.3.*.*	Forward (2)
Ingress Port = 4; IP Dst = 10.1.*.*	Forward (1)
Ingress Port = 4; IP Dst = 10.3.*.*	Forward (1)