Problem1

1.1:The purpose is to help the routers to decide where the data should go and ensure the data could arrive fast.

1.2:First, it separated the concerns, designers could break a big problem into small segments, so the problem could be more easily to solve. Second, it's more independent for each layer, each layer would concern their own problems and would not affect other layers when displace the protocol etc.

1.3:Link layer convert packets to frame, but physical layer convert frame to bits. Besides Link layer would check the data reliability using checksum or CRC method, however physical layer only transform the bits without checking.

1.4:Physical Layer:"bit" Link Layer:"frame" Network Layer:"packet" Transport Lyaer:"datagram/segment"

1.5:ex1:I log in mysjsu account to view messages.(End -to-end Connection)
Ex2:Inform sender to stop sending data by using some variables, such as rwnd which means receiver window.(Congestion Control)
Ex3:Host runs several applications and send data over internet simultaneously(Multiplexing)

Problem 2

Ex:Connection-Oriented:Using SSH(Secure shell) to upload and download files.

Ex:Connectionless-Oriented: Video chat with other people

For the Connection-Oriented, you have to establish a channel before, then the hosts could communicate, besides the data source is more reliable, because the receiver will check if the data has lost and can ask sender to resend the data.

For the Connectionless-Oriented, hosts don't need to establish the channel, and the data is less reliable than the Connection-Oriented, since it will not do the check process, but the advantage of Connectionless is that it doesn't need a good bandwidth compare with Connection-Oriented.

Problem 3

According to the question, physical transport will take 10+100/40=12.5h=45000 second The upload time will be D/100*2^20 Second, in order to let physical transport make sense we should let D/100*2^20>45000, so the minimum Value of D will be 4718592000000 +1 Which is 47185920000001.

Problem 4

1600*1200*3=5760000 bytes =45000kbit=43.945 Mb=0.0429 Gb

For the (i)45000/56=803 Second.

For the(ii) 43.945 /1=43.945 Second.

For the (iii) 43.945 /10=4.3945Second.

For the (iv) 43.945 /100=0.43953 Second.

For the (v) 0.0429 /1=0.0429 Second.

Problem 5

3.1.1

I guess the reason could be that you send more than 5 packets(Interval is 1 second), so before the packets get to the server the program could be over and returned, therefore we could set the stop time in a very large number to ensure we could see the result. (Actually after the experiment, I see the reason is the client side program stopped before client can receive

the message , so I increased the client side and server side stop time to 40 seconds!! And it works now)

3.1.2

Server:

Packet size\Data rate	512bps	1Kbps	512Kbps	1Mbps
128	4.46975	3.23537	2.00341	2.00221
256	6.46975	4.23538	2.00536	2.00318
1024	18.4698	10.2354	2.01708	2.00904

Client:

Packet size\Data rate	512bps	1Kbps	521Kbps	1Mbps
128	6.9395	4.47075	2.00682	2.00441
256	10.9395	6.47075	2.01073	2.00636
1024	34.9395	18.4707	2.03417	2.01808

3.2.1:

Of course we need to pay attention to the port number, since different application and protocol will use different port number, such as HTTP uses port 80, but Https use 443.

3.2.2

Server: At time which a message is received at the server

	Data Rate(csma)	Delay(csma)	Time
Application1(port9)	100Mbps	6560(Nanao Second)	2.0118
Application2(port10)	100Mbps	6560(Nano Second)	2.0119

Client: At time which a message is received at the Client

	Data Rate(p2p)	Delay(p2p)	Packet Size(p2p)	Time
Application1(port9)	5Mbps	2ms	1024	2.02461
Application2(port10)	5Mbps	2ms	1024	2.02629

3.3.1

A:throughput from the echo client to server:0.840153Mbps

```
/NodeList/6/$ns3::MobilityModel/CourseChange x = 3.95127, y = -1.58857
/NodeList/6/$ns3::MobilityModel/CourseChange x = 3.64777, y = -0.635735
/NodeList/6/$ns3::MobilityModel/CourseChange x = 3.6617, y = -1.63564
/NodeList/6/$ns3::MobilityModel/CourseChange x = 3.73417, y = -2.63301
/NodeList/6/$ns3::MobilityModel/CourseChange x = 3.94601, y = -3.61031
/NodeList/6/$ns3::MobilityModel/CourseChange x = 3.41179, y = -2.76497
/NodeList/6/$ns3::MobilityModel/CourseChange x = 2.41233, y = -2.73219
/NodeList/6/$ns3::MobilityModel/CourseChange x = 1.79802, y = -3.52125
/NodeList/6/$ns3::MobilityModel/CourseChange x = 2.36079, y = -4.34787
/NodeList/6/$ns3::MobilityModel/CourseChange x = 1.51012, y = -4.87357
/NodeList/6/$ns3::MobilityModel/CourseChange x = 0.655258, y = -5.39241
/NodeList/6/$ns3::MobilityModel/CourseChange x = 0.260733, y = -6.3113
/NodeList/6/$ns3::MobilityModel/CourseChange x = -0.700195, y = -6.03451
/NodeList/6/$ns3::MobilityModel/CourseChange x = -1.08427, y = -5.1112
/NodeList/6/$ns3::MobilityModel/CourseChange x = -0.108713, y = -5.33097
/NodeList/6/$ns3::MobilityModel/CourseChange x = -0.296186, y = -6.31324
Flow 1 (10.1.3.3 -> 10.1.3.2)
  Tx Packets: 1
  Tx Bytes:   1052
  Rx Bytes:   1052
  Throughput: 0.840153 Mbps
Flow 2 (10.1.3.2 -> 10.1.3.3)
  Tx Packets: 1
  Tx Bytes:   1052
```

```
Flow 1 (10.1.3.3 -> 10.1.3.2)
  Tx Packets: 1
  Tx Bytes:   1052
  Rx Bytes:   1052
  Throughput: 0.840153 Mbps
```

B: throughput from the echo server to client:1.18866Mbps

```
Flow 2 (10.1.3.2 -> 10.1.3.3)
  Tx Packets: 1
  Tx Bytes:   1052
  Rx Bytes:   1052
  Throughput: 1.18866 Mbps
HaoRans-MacBook-Pro:ns-3.30.1 haoranchen$
```

3.3.2

Since my ID is 012494781 so ,I change the wifi devices to 10.

A:Final position of the server(2.10029,16.6314)

```
/NodeList/13/$ns3::MobilityModel/CourseChange x = 2.10029, y = 16.6314
```

B:final position of the Client(0.240376,24.6561)

```
/NodeList/14/$ns3::MobilityModel/CourseChange x = 0.240376, y = 24.6561
Flow 1 (10.1.3.10 -> 10.1.3.9)
```

C: throughput from the echo server to client:0.76203Mbps

```
Flow 1 (10.1.3.10 -> 10.1.3.9)
  Tx Packets: 1
  Tx Bytes:   1052
  Rx Bytes:   1052
  Throughput: 0.76203 Mbps
```

D: throughput from the echo server to client:0.754042Mbps

```
Flow 2 (10.1.3.9 -> 10.1.3.10)
  Tx Packets: 1
  Tx Bytes:   1052
  Rx Bytes:   1052
  Throughput: 0.754042 Mbps
HaoRans-MacBook-Pro:ns-3.30.1 haoranchen$
```

E:Ip address of the echo server is 10.1.3.9

F:Ip address of the echo client is 10.1.3.10

Reference :

1.TCP/IP Illustrated Volum1 The protocols (Kevin R. Fall W. Richard Stevens).

2.GeeksForGeeks(https://www.geeksforgeeks.org/transport-layer-responsibilities/
).

3.Difference between connectionless and
Connection(http://rishabhcs206.blogspot.com/p/difference-between-connectionless-and.html
)

4.Quora(https://www.quora.com/How-many-bit-in-1kb
)

5.Class PPT.

6.Ns3-Tutorial