

## PART-A

Ans 1:

No.	Time	Source	Destination	Protocol	Length	Info
131	18.731275551	10.0.2.15	128.119.245.12	HTTP	21960	POST /wireshark-labs/lab3-1-reply.htm HTTP/1.1 (text/plain)
149	19.494639401	128.119.245.12	10.0.2.15	HTTP	833	HTTP/1.1 200 OK (text/html)

**Figure 1:** The HTTP POST message that uploaded alice.txt from your computer to gaia.cs.umass.edu

Frame 131: 21960 bytes on wire (175680 bits), 21960 bytes captured (175680 bits) on interface any, Linux cooked capture v1	
Internet Protocol Version 4, Src: 10.0.2.15, Dst: 128.119.245.12	
Transmission Control Protocol, Src Port: 46914, Dst Port: 80, Seq: 131073, Ack: 1, Len: 21904	
Source Port: 46914	
Destination Port: 80	
[Stream index: 1]	
[Conversation completeness: Incomplete, DATA (15)]	
[TCP Segment Len: 21904]	
Sequence Number: 131073 (relative sequence number)	
Sequence Number (raw): 93880349	
[Next Sequence Number: 152977 (relative sequence number)]	
Acknowledgment Number: 1 (relative ack number)	
Acknowledgment number (raw): 311616002	
[Header Length: 20 bytes (5)]	
0020 80 77 f5 0c b7 42 00 50 00 8f 21 fd 12 92 e2 02 .w..B.P...!.....	
0030 50 18 fa f0 d7 3d 00 00 65 73 2c 27 20 79 6f 75 p....=..es,' you	
0040 20 73 65 65 2c 20 62 65 63 61 75 73 65 0d 0a 73 see, be cause..s	
0050 6f 6d 65 20 6f 66 20 74 68 65 6d 20 77 65 72 65 ome of t hem were	
0060 20 61 6e 69 6d 61 6c 73 2c 20 61 6e 64 20 73 6f animals , and so	
0070 6d 65 20 77 65 72 65 20 62 69 72 64 73 2c 29 20 me were birds,)	
0080 60 49 20 73 75 70 70 6f 73 65 20 74 68 65 79 0d 'I suppo se they.	
0090 0a 61 72 65 20 74 68 65 20 6a 75 72 6f 72 73 2e .are the jurors.	
00a0 27 20 20 53 68 65 20 73 61 69 64 20 74 68 69 73 ' She s aid this	
00b0 20 6c 61 73 74 20 77 6f 72 64 20 74 77 6f 20 6f last wo rd two o	
00c0 72 20 74 68 72 65 65 20 74 69 6d 65 73 20 6f 76 r three times ov	
00d0 65 72 0d 0a 74 6f 20 68 65 72 73 65 6c 66 2c 20 er..to h erself,	
00e0 62 65 69 6e 67 20 72 61 74 68 65 72 20 70 72 6f being ra ther pro	
00f0 75 64 20 6f 66 20 69 74 3a 20 20 66 6f 72 20 73 ud of it : for s	
0100 68 65 20 74 68 6f 75 67 68 74 2c 20 61 6e 64 0d he thoug ht, and-	
0110 0a 72 69 67 68 74 6c 79 20 74 6f 6f 2c 20 74 68 .rightly too, th	
Frame (21960 bytes)	Reassembled TCP (152976 bytes)

Frame 131: 21960 bytes on wire (175680 bits), 21960 bytes captured (175680 bits) on interface any, Linux cooked capture v1	
Internet Protocol Version 4, Src: 10.0.2.15, Dst: 128.119.245.12	
Transmission Control Protocol, Src Port: 46914, Dst Port: 80, Seq: 131073, Ack: 1, Len: 21904	
[23 Reassembled TCP Segments (152976 bytes): #17(2920), #18(2920), #19(2920), #20(2920), #21(2920)]	
[Frame: 17, payload: 0-2919 (2920 bytes)]	
[Frame: 18, payload: 2920-5839 (2920 bytes)]	
[Frame: 19, payload: 5840-8759 (2920 bytes)]	
[Frame: 20, payload: 8760-11679 (2920 bytes)]	
[Frame: 21, payload: 11680-14599 (2920 bytes)]	
[Frame: 23, payload: 14600-17519 (2920 bytes)]	
[Frame: 25, payload: 17520-20439 (2920 bytes)]	
[Frame: 27, payload: 20440-23359 (2920 bytes)]	
[Frame: 29, payload: 23360-26279 (2920 bytes)]	
[Frame: 31, payload: 26280-29199 (2920 bytes)]	
00000000 50 4f 53 54 20 2f 77 69 72 65 73 68 61 72 6b 2d POST /wi reshark-	
00000010 6c 61 62 73 2f 6c 61 62 33 2d 31 2d 72 65 70 6c labs/lab 3-1-repl	
00000020 79 2e 68 74 6d 20 48 54 54 50 2f 31 2e 31 0d 0a y.htm HT TP/1.1..	
00000030 48 6f 73 74 3a 20 67 61 69 61 2e 63 73 2e 75 6d Host: ga ia.cs.um	
00000040 61 73 73 2e 65 64 75 0d 0a 55 73 65 72 2d 41 67 ass.edu..User-Ag	
00000050 65 6e 74 3a 20 4d 6f 7a 69 6c 6c 61 2f 35 2e 30 ent: Moz illa/5.0	
00000060 20 28 58 31 31 3b 20 55 62 75 6e 74 75 3b 20 4c (X11; U buntu; L	
00000070 69 6e 75 78 20 78 38 36 5f 36 34 3b 20 72 76 3a inux x86 _64; rv:	
00000080 31 30 39 2e 30 29 20 47 65 63 6b 6f 2f 32 30 31 109.0) G ecko/201	
00000090 30 30 31 30 31 20 46 69 72 65 66 6f 78 2f 31 31 00101 F1 refox/11	
000000a0 36 2e 30 0d 0a 41 63 63 65 70 74 3a 20 74 65 78 6.0..Acc ept: tex	
000000b0 74 2f 68 74 6d 6c 2c 61 70 70 6c 69 63 61 74 69 t/html,a pplicati	
000000c0 6f 6e 2f 70 68 74 6d 6c 2b 78 6d 6c 2c 61 70 70 on/xhtmll +xml,app	
000000d0 6c 69 63 61 74 69 6f 6e 2f 78 6d 6c 30 71 3d 30 lication /xml;q=0	
000000e0 2e 39 2c 69 6d 61 67 65 2f 61 76 69 66 2c 69 6d .0,image /xml;q=0	
000000f0 61 67 65 2f 77 65 62 70 2c 2a 2f 2a 3b 71 3d 30 age/webp , /*;q=0	
Frame (21960 bytes)	Reassembled TCP (152976 bytes)

**Figure 2, 3:** expanding the HTTP POST message that uploaded alice.txt from your computer to gaia.cs.umass.edu

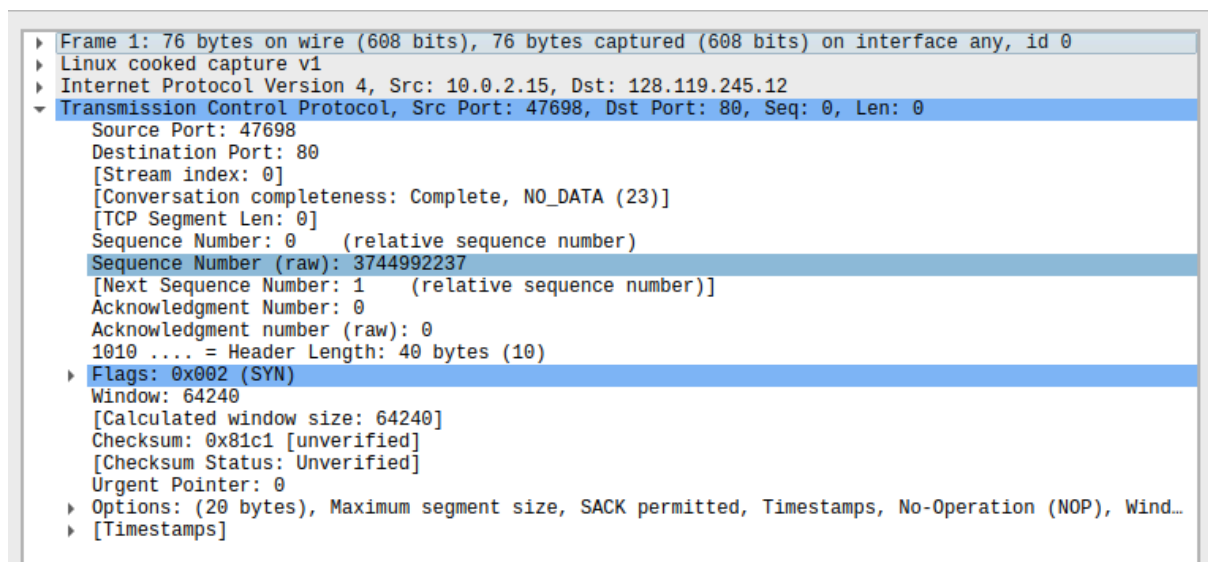
Here IP address of client computer (source) is **10.0.2.15**  
And Source Port is **46914**

**Ans 2:**

Here IP address of the gaia.cs.umass.edu is **128.119.245.12**  
And Destination Port is **80**

**Ans 3:**

Here we can see the sequence Number of the TCP SYN segment that is used to initiate the TCP connection between the client computer and gaia.cs.umass.edu is **3744992237**  
Where we can see that the relative sequence number is 0.



**Figure 4:** expanding the TCP SYN segment your computer to gaia.cs.umass.edu

Because of **Flags 0x002 (SYN)**

Here we can see the client computer (source) in SYN segment in options allow the permission of SACK

were gaia.cs.umass.edu in SYN segment have not given permission for the SACK

So we can not use selective Acknowledgments

```

▶ Frame 1: 76 bytes on wire (608 bits), 76 bytes captured (608 bits) on interface any, id 0
▶ Linux cooked capture v1
▶ Internet Protocol Version 4, Src: 10.0.2.15, Dst: 128.119.245.12
▼ Transmission Control Protocol, Src Port: 47698, Dst Port: 80, Seq: 0, Len: 0
  Source Port: 47698
  Destination Port: 80
  [Stream index: 0]
  [Conversation completeness: Complete, NO_DATA (23)]
  [TCP Segment Len: 0]
  Sequence Number: 0 (relative sequence number)
  Sequence Number (raw): 3744992237
  [Next Sequence Number: 1 (relative sequence number)]
  Acknowledgment Number: 0
  Acknowledgment number (raw): 0
  1010 .... = Header Length: 40 bytes (10)
▶ Flags: 0x002 (SYN)
  Window: 64240
  [Calculated window size: 64240]
  Checksum: 0x81c1 [unverified]
  [Checksum Status: Unverified]
  Urgent Pointer: 0
▼ Options: (20 bytes), Maximum segment size, SACK permitted, Timestamps, No-Operation (NOP)
  ▶ TCP Option - Maximum segment size: 1460 bytes
  ▶ TCP Option - SACK permitted
  ▶ TCP Option - Timestamps
  ▶ TCP Option - No-Operation (NOP)
  ▶ TCP Option - Window scale: 7 (multiply by 128)
  ▶ [Timestamps]

```

**Figure 5:** expanding the TCP SYN segment from your computer to gaia.cs.umass.edu

```

▶ Frame 2: 62 bytes on wire (496 bits), 62 bytes captured (496 bits) on interface any, id 0
▶ Linux cooked capture v1
▶ Internet Protocol Version 4, Src: 128.119.245.12, Dst: 10.0.2.15
▼ Transmission Control Protocol, Src Port: 80, Dst Port: 47698, Seq: 0, Ack: 1, Len: 0
  Source Port: 80
  Destination Port: 47698
  [Stream index: 0]
  [Conversation completeness: Complete, NO_DATA (23)]
  [TCP Segment Len: 0]
  Sequence Number: 0 (relative sequence number)
  Sequence Number (raw): 309312001
  [Next Sequence Number: 1 (relative sequence number)]
  Acknowledgment Number: 1 (relative ack number)
  Acknowledgment number (raw): 3744992238
  0110 .... = Header Length: 24 bytes (6)
▶ Flags: 0x012 (SYN, ACK)
  Window: 65535
  [Calculated window size: 65535]
  Checksum: 0xa449 [unverified]
  [Checksum Status: Unverified]
  Urgent Pointer: 0
▼ Options: (4 bytes), Maximum segment size
  ▶ TCP Option - Maximum segment size: 1460 bytes
    Kind: Maximum Segment Size (2)
    Length: 4
    MSS Value: 1460
  ▶ [Timestamps]
  ▶ [SEQ/ACK analysis]

```

**Figure 6:** expanding the TCP SYNACK segment that to your computer from gaia.cs.umass.edu

**Ans 4:**

In figure 6 we can see sequence Number (raw) for asked segment is **309312001**

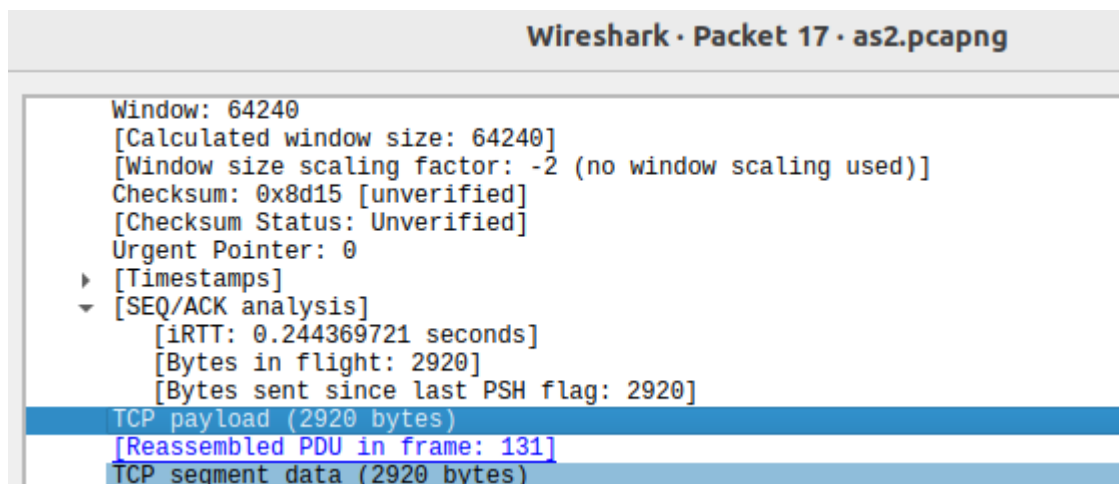
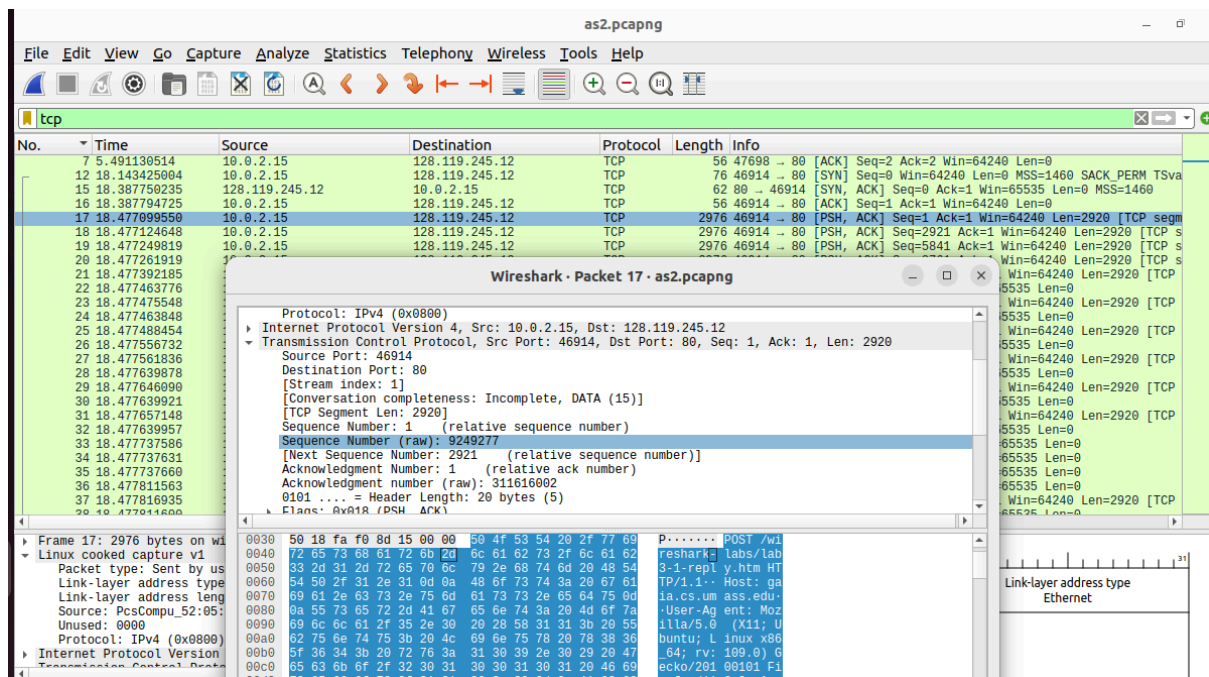
In figure 6 we can see Flags: 0x012 (SYN, ACK) it shows that the segment identifies as a SYNACK

in figure 6 we can see Acknowledgement number (raw) 3744992238 (which is 3744992237+1 where 3744992237 is sequence number of SYN segment as shown in figure 5 and 1 because length of the segment is 0 so we want next byte of the message, we can see the length of SYN segment in figure 5)

**Ans 5:**

Here in the given figure 7 we can see that data field contain the ascii 'POST'

Sequence number of this segment is **9249277**



**Figure 7 :** expanding TCP segment containing the header of the HTTP POST command

Here 2920 bytes of data are contained in the payload (data) field of this TCP  
No, all of the data in the transferred file alice.txt fit into this single segment

**Ans 6:**

15	18.387750235	128.119.245.12	10.0.2.15	TCP	62 80 → 46914	[SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460
16	18.387794725	10.0.2.15	128.119.245.12	TCP	56 46914 → 80	[ACK] Seq=1 Ack=1 Win=64240 Len=0
17	18.477099550	10.0.2.15	128.119.245.12	TCP	2976 46914 → 80	[PSH, ACK] Seq=1 Ack=1 Win=64240 Len=2920 [TCP s
18	18.477124648	10.0.2.15	128.119.245.12	TCP	2976 46914 → 80	[PSH, ACK] Seq=2921 Ack=1 Win=64240 Len=2920 [TCP s
19	18.477249819	10.0.2.15	128.119.245.12	TCP	2976 46914 → 80	[PSH, ACK] Seq=5841 Ack=1 Win=64240 Len=2920 [TCP s
20	18.477261919	10.0.2.15	128.119.245.12	TCP	2976 46914 → 80	[PSH, ACK] Seq=8761 Ack=1 Win=64240 Len=2920 [TCP s
21	18.477392185	10.0.2.15	128.119.245.12	TCP	2976 46914 → 80	[PSH, ACK] Seq=11681 Ack=1 Win=64240 Len=2920 [TCP s
22	18.477463776	128.119.245.12	10.0.2.15	TCP	62 80 → 46914	[ACK] Seq=1 Ack=1461 Win=65535 Len=0
23	18.477475548	10.0.2.15	128.119.245.12	TCP	2976 46914 → 80	[PSH, ACK] Seq=14601 Ack=1 Win=64240 Len=2920 [TCP s
24	18.477463848	128.119.245.12	10.0.2.15	TCP	62 80 → 46914	[ACK] Seq=1 Ack=2921 Win=65535 Len=0
25	18.477488454	10.0.2.15	128.119.245.12	TCP	2976 46914 → 80	[PSH, ACK] Seq=17521 Ack=1 Win=64240 Len=2920 [TCP s
26	18.477556732	128.119.245.12	10.0.2.15	TCP	62 80 → 46914	[ACK] Seq=1 Ack=4381 Win=65535 Len=0
27	18.477561836	10.0.2.15	128.119.245.12	TCP	2976 46914 → 80	[PSH, ACK] Seq=20441 Ack=1 Win=64240 Len=2920 [TCP s
28	18.477639878	128.119.245.12	10.0.2.15	TCP	62 80 → 46914	[ACK] Seq=1 Ack=5841 Win=65535 Len=0
29	18.477646090	10.0.2.15	128.119.245.12	TCP	2976 46914 → 80	[PSH, ACK] Seq=23361 Ack=1 Win=64240 Len=2920 [TCP s
30	18.477639921	128.119.245.12	10.0.2.15	TCP	62 80 → 46914	[ACK] Seq=1 Ack=7301 Win=65535 Len=0

At what time was the first segment (the one containing the HTTP POST) in the data-transfer part of the TCP connection sent?

**Ans: 18.477099550 sec.**

At what time was the ACK for this first data-containing segment received?  
What is the RTT for this first data-containing segment?

**Ans: 18.477463848 sec.** and RTT for this first data containing segment is **0.000364298 sec.**

**Ans: 18.477639878-18.477124648=0.0005130 sec.**

18	18.477124648	10.0.2.15	128.119.245.12	TCP	2976 46914 → 80	[PSH, ACK] Seq=2921 Ack=1 Win=64240 Len=2920 [TCP s
19	18.477249819	10.0.2.15	128.119.245.12	TCP	2976 46914 → 80	[PSH, ACK] Seq=5841 Ack=1 Win=64240 Len=2920 [TCP s
20	18.477261919	10.0.2.15	128.119.245.12	TCP	2976 46914 → 80	[PSH, ACK] Seq=8761 Ack=1 Win=64240 Len=2920 [TCP s
21	18.477392185	10.0.2.15	128.119.245.12	TCP	2976 46914 → 80	[PSH, ACK] Seq=11681 Ack=1 Win=64240 Len=2920 [TCP s
22	18.477463776	128.119.245.12	10.0.2.15	TCP	62 80 → 46914	[ACK] Seq=1 Ack=1461 Win=65535 Len=0
23	18.477475548	10.0.2.15	128.119.245.12	TCP	2976 46914 → 80	[PSH, ACK] Seq=14601 Ack=1 Win=64240 Len=2920 [TCP s
24	18.477463848	128.119.245.12	10.0.2.15	TCP	62 80 → 46914	[ACK] Seq=1 Ack=2921 Win=65535 Len=0
25	18.477488454	10.0.2.15	128.119.245.12	TCP	2976 46914 → 80	[PSH, ACK] Seq=17521 Ack=1 Win=64240 Len=2920 [TCP s
26	18.477556732	128.119.245.12	10.0.2.15	TCP	62 80 → 46914	[ACK] Seq=1 Ack=4381 Win=65535 Len=0
27	18.477561836	10.0.2.15	128.119.245.12	TCP	2976 46914 → 80	[PSH, ACK] Seq=20441 Ack=1 Win=64240 Len=2920 [TCP s
28	18.477639878	128.119.245.12	10.0.2.15	TCP	62 80 → 46914	[ACK] Seq=1 Ack=5841 Win=65535 Len=0
29	18.477646090	10.0.2.15	128.119.245.12	TCP	2976 46914 → 80	[PSH, ACK] Seq=23361 Ack=1 Win=64240 Len=2920 [TCP s
30	18.477639921	128.119.245.12	10.0.2.15	TCP	62 80 → 46914	[ACK] Seq=1 Ack=7301 Win=65535 Len=0
31	18.477657148	10.0.2.15	128.119.245.12	TCP	2976 46914 → 80	[PSH, ACK] Seq=26281 Ack=1 Win=64240 Len=2920 [TCP s
32	18.477639957	128.119.245.12	10.0.2.15	TCP	62 80 → 46914	[ACK] Seq=1 Ack=8761 Win=65535 Len=0
33	18.477737586	128.119.245.12	10.0.2.15	TCP	62 80 → 46914	[ACK] Seq=1 Ack=10221 Win=65535 Len=0

**Ans:**

RTT for first packet : 18.477463848-18.477099550 = 0.000366

RTT for second packet : 18.477639878-18.477124648 = 0.000513

EstimatedRTT = (1-alfa) \* Previously estimated RTT + alfa \* Sample RTT  
= (1-0.125) \* 0.000366 + (0.125) \* 0.000513  
= 0.00032025 + 0.000064125 = **0.00038375**



Ans 7:

17	18.477099550	10.0.2.15	128.119.245.12	TCP	2976	46914	→ 80	[PSH, ACK]	Seq=1 Ack=1 Win=64240 Len=2920 [TCP segm
18	18.477124648	10.0.2.15	128.119.245.12	TCP	2976	46914	→ 80	[PSH, ACK]	Seq=2921 Ack=1 Win=64240 Len=2920 [TCP s
19	18.477249819	10.0.2.15	128.119.245.12	TCP	2976	46914	→ 80	[PSH, ACK]	Seq=5841 Ack=1 Win=64240 Len=2920 [TCP s
20	18.477261919	10.0.2.15	128.119.245.12	TCP	2976	46914	→ 80	[PSH, ACK]	Seq=8761 Ack=1 Win=64240 Len=2920 [TCP s
21	18.477392185	10.0.2.15	128.119.245.12	TCP	2976	46914	→ 80	[PSH, ACK]	Seq=11681 Ack=1 Win=64240 Len=2920 [TCP

Here we can see 5 segments and their relative sequence number,

To calculate payload of the first 4 packets we can do relative sequence of 5th packet -1

= 11681-1

=11680

Now we add header length of each packets

=11680+80

=11760

Or we can calculate other way is : = 4(2920+20) because each packet have

payload=2920 bytes and header length is 20 bytes

=11760

Ans.8:

```

sequence number: 1      (relative sequence number)
Sequence Number (raw): 311616002
[Next Sequence Number: 1      (relative sequence number)]
Acknowledgment Number: 2921      (relative ack number)
Acknowledgment number (raw): 9252197
0101 .... = Header Length: 20 bytes (5)
▶ Flags: 0x010 (ACK)
Window: 65535
[Calculated window size: 65535]
[Window size scaling factor: -2 (no window scaling used)]
Checksum: 0x5428 [unverified]
[Checksum Status: Unverified]

```

the minimum amount of available buffer space advertised to the client by

gaia.cs.umass.edu among these first data-carrying TCP segments is 65535

And it is the same for all next five segments and there is no throttle. But after 45 packets we can see it started to decrease

70	18.481788063	128.119.245.12	10.0.2.15	TCP	62	80	→ 46914	[ACK]	Seq=1 Ack=50289 Win=65535 Len=0
71	18.481788116	128.119.245.12	10.0.2.15	TCP	62	80	→ 46914	[ACK]	Seq=1 Ack=51749 Win=65535 Len=0
72	18.481788161	128.119.245.12	10.0.2.15	TCP	62	80	→ 46914	[ACK]	Seq=1 Ack=53209 Win=65535 Len=0
73	18.481833985	128.119.245.12	10.0.2.15	TCP	62	80	→ 46914	[ACK]	Seq=1 Ack=54689 Win=65535 Len=0
74	18.481834038	128.119.245.12	10.0.2.15	TCP	62	80	→ 46914	[ACK]	Seq=1 Ack=56129 Win=65535 Len=0
75	18.481834080	128.119.245.12	10.0.2.15	TCP	62	80	→ 46914	[ACK]	Seq=1 Ack=57589 Win=65535 Len=0
76	18.481834125	128.119.245.12	10.0.2.15	TCP	62	80	→ 46914	[ACK]	Seq=1 Ack=59049 Win=65535 Len=0
77	18.481834168	128.119.245.12	10.0.2.15	TCP	62	80	→ 46914	[ACK]	Seq=1 Ack=60509 Win=65535 Len=0
78	18.481834210	128.119.245.12	10.0.2.15	TCP	62	80	→ 46914	[ACK]	Seq=1 Ack=61969 Win=65535 Len=0
79	18.481834260	128.119.245.12	10.0.2.15	TCP	62	80	→ 46914	[ACK]	Seq=1 Ack=63429 Win=65535 Len=0
80	18.481834303	128.119.245.12	10.0.2.15	TCP	62	80	→ 46914	[ACK]	Seq=1 Ack=64889 Win=65535 Len=0
82	18.481853287	128.119.245.12	10.0.2.15	TCP	62	80	→ 46914	[ACK]	Seq=1 Ack=66349 Win=65535 Len=0
83	18.481853331	128.119.245.12	10.0.2.15	TCP	62	80	→ 46914	[ACK]	Seq=1 Ack=67809 Win=64240 Len=0
84	18.481853371	128.119.245.12	10.0.2.15	TCP	62	80	→ 46914	[ACK]	Seq=1 Ack=69269 Win=62780 Len=0
85	18.481853417	128.119.245.12	10.0.2.15	TCP	62	80	→ 46914	[ACK]	Seq=1 Ack=70729 Win=61320 Len=0
86	18.481853461	128.119.245.12	10.0.2.15	TCP	62	80	→ 46914	[ACK]	Seq=1 Ack=72189 Win=59860 Len=0
87	18.481853493	128.119.245.12	10.0.2.15	TCP	62	80	→ 46914	[ACK]	Seq=1 Ack=73649 Win=58400 Len=0

```

Acknowledgment Number: 131073 (relative ack number)
Acknowledgment number (raw): 9380349
0101 .... = Header Length: 20 bytes (5)
Flags: 0x010 (ACK)
Window: 976
[Calculated window size: 976]
[Window size scaling factor: -2 (no window scaling used)]
Checksum: 0x5bbe [unverified]
[Checksum Status: Unverified]
Urgent Pointer: 0
[Timestamps]
[SEQ/ACK analysis]
0000 00 00 00 01 00 06 52 54 00 12 35 02 00 00 08 00 .....RT...5.....
0010 45 00 00 28 4f 72 00 00 40 06 a9 cb 80 77 f5 0c E..(Or...@...w...
0020 0a 00 02 0f 00 50 b7 42 12 92 e2 02 00 8f 21 fd .....P..B.....!..
0030 50 10 03 d0 5b be 00 00 00 00 00 00 00 00 00 00 P...[.....

```

128	18.482232125	128.119.245.12	10.0.2.15	TCP	62	80	→	46914	[ACK]	Seq=1	Ack=130589	Win=1460	Len=0
129	18.482232156	128.119.245.12	10.0.2.15	TCP	62	80	→	46914	[ACK]	Seq=1	Ack=131073	Win=976	Len=0
130	18.731256574	128.119.245.12	10.0.2.15	TCP	62	80	→	46914	[ACK]	Seq=1	Ack=131073	Win=6553	Len=0
132	18.731619928	128.119.245.12	10.0.2.15	TCP	62	80	→	46914	[ACK]	Seq=1	Ack=132533	Win=64240	Len=0
133	18.731620015	128.119.245.12	10.0.2.15	TCP	62	80	→	46914	[ACK]	Seq=1	Ack=133993	Win=62780	Len=0
134	18.731620047	128.119.245.12	10.0.2.15	TCP	62	80	→	46914	[ACK]	Seq=1	Ack=135453	Win=61320	Len=0
135	18.731620078	128.119.245.12	10.0.2.15	TCP	62	80	→	46914	[ACK]	Seq=1	Ack=136913	Win=59860	Len=0
136	18.731620107	128.119.245.12	10.0.2.15	TCP	62	80	→	46914	[ACK]	Seq=1	Ack=138373	Win=58400	Len=0

Here we see one of the the minimum amount of available buffer space advertised by the receiver to client is 976 and after it shows **TCP Window Update**

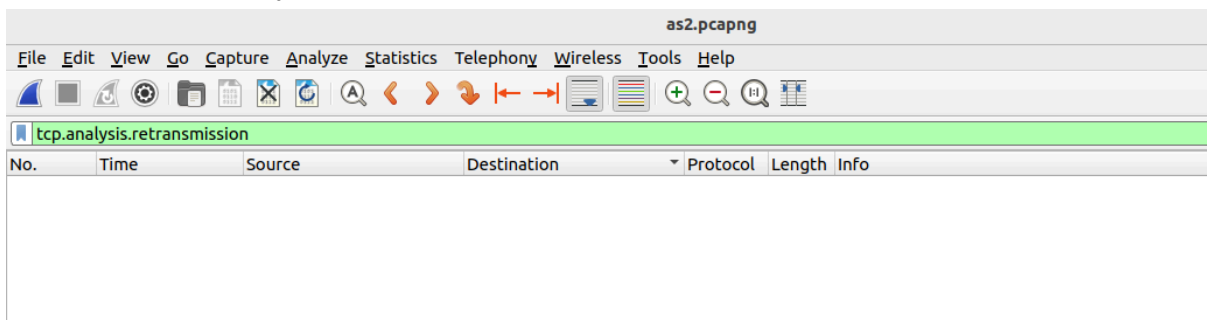
#### Ans 9:

the minimum amount of available buffer space advertised by the client to Gaia.cs.umass.edu is 64240 and it remains same for all the packets but in question 8 we can say that the window size of Gain.cs.umass.edu will not remain the same.

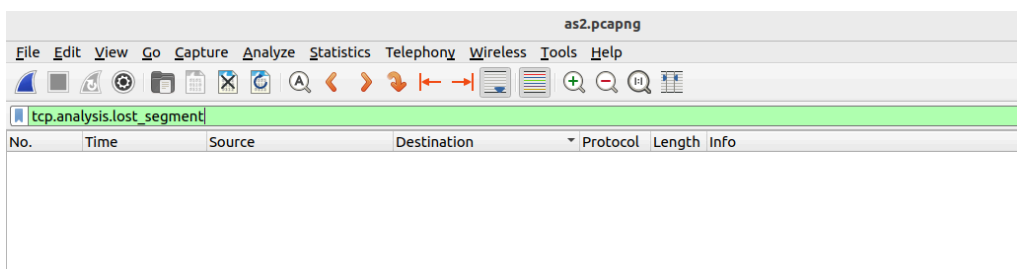
#### Ans 10:

Here we can see that there is no duplicate packets

We can use tcp.analysis.retransmission to filter the packets



We can also use tcp.analysis.lost\_segment to show any packers that lost and result in retransmission.



**Ans 11:**

13	18.146465215	192.168.36.53	10.0.2.15	DNS	143	Standard query response 0x91c9 AAAA gaia.cs.umass.edu SOA unix
15	18.387750235	128.119.245.12	10.0.2.15	TCP	62	80 → 46914 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460
22	18.477463776	128.119.245.12	10.0.2.15	TCP	62	80 → 46914 [ACK] Seq=1 Ack=1461 Win=65535 Len=0
24	18.477463848	128.119.245.12	10.0.2.15	TCP	62	80 → 46914 [ACK] Seq=1 Ack=2921 Win=65535 Len=0
26	18.477556732	128.119.245.12	10.0.2.15	TCP	62	80 → 46914 [ACK] Seq=1 Ack=4381 Win=65535 Len=0
28	18.477639878	128.119.245.12	10.0.2.15	TCP	62	80 → 46914 [ACK] Seq=1 Ack=5841 Win=65535 Len=0
30	18.477639921	128.119.245.12	10.0.2.15	TCP	62	80 → 46914 [ACK] Seq=1 Ack=7381 Win=65535 Len=0
32	18.477639957	128.119.245.12	10.0.2.15	TCP	62	80 → 46914 [ACK] Seq=1 Ack=8761 Win=65535 Len=0
33	18.477737586	128.119.245.12	10.0.2.15	TCP	62	80 → 46914 [ACK] Seq=1 Ack=10221 Win=65535 Len=0
34	18.477737631	128.119.245.12	10.0.2.15	TCP	62	80 → 46914 [ACK] Seq=1 Ack=11681 Win=65535 Len=0
35	18.477737660	128.119.245.12	10.0.2.15	TCP	62	80 → 46914 [ACK] Seq=1 Ack=13141 Win=65535 Len=0
36	18.477811563	128.119.245.12	10.0.2.15	TCP	62	80 → 46914 [ACK] Seq=1 Ack=14601 Win=65535 Len=0
38	18.477811600	128.119.245.12	10.0.2.15	TCP	62	80 → 46914 [ACK] Seq=1 Ack=16061 Win=65535 Len=0
39	18.477811636	128.119.245.12	10.0.2.15	TCP	62	80 → 46914 [ACK] Seq=1 Ack=17521 Win=65535 Len=0
40	18.477811672	128.119.245.12	10.0.2.15	TCP	62	80 → 46914 [ACK] Seq=1 Ack=18981 Win=65535 Len=0
41	18.477811703	128.119.245.12	10.0.2.15	TCP	62	80 → 46914 [ACK] Seq=1 Ack=20441 Win=65535 Len=0
42	18.477811733	128.119.245.12	10.0.2.15	TCP	62	80 → 46914 [ACK] Seq=1 Ack=21901 Win=65535 Len=0
43	18.477811763	128.119.245.12	10.0.2.15	TCP	62	80 → 46914 [ACK] Seq=1 Ack=23361 Win=65535 Len=0
44	18.477811793	128.119.245.12	10.0.2.15	TCP	62	80 → 46914 [ACK] Seq=1 Ack=24821 Win=65535 Len=0
45	18.477840921	128.119.245.12	10.0.2.15	TCP	62	80 → 46914 [ACK] Seq=1 Ack=26281 Win=65535 Len=0
46	18.477840956	128.119.245.12	10.0.2.15	TCP	62	80 → 46914 [ACK] Seq=1 Ack=27741 Win=65535 Len=0
48	18.477840986	128.119.245.12	10.0.2.15	TCP	62	80 → 46914 [ACK] Seq=1 Ack=29201 Win=65535 Len=0

1460 bytes of data does the receiver typically acknowledge in an ACK among the first ten data-carrying segments sent from the client to gaia.cs.umass.edu, I can't identify cases where the receiver is ACKing every other received segment among these first ten data-carrying segments.

**Ans 12:**

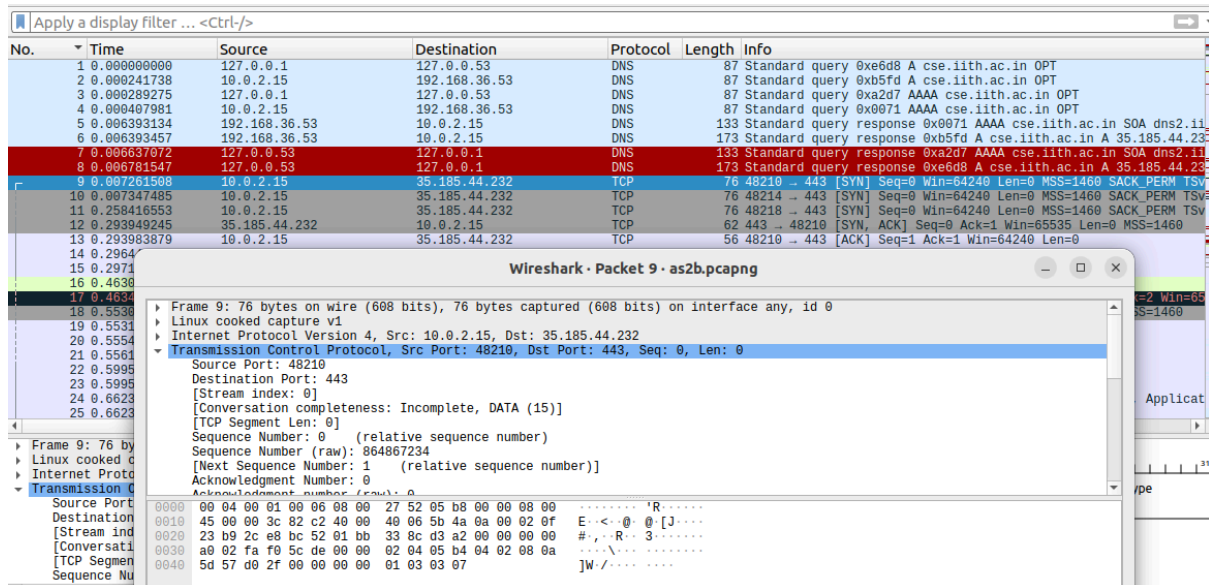
Total data transferred = 152977-1 = 152976 (because 152977 is ack of last segment)  
 Know let time t is time difference between first packet send from client and last ack received at client = 18.731644940-18.477099550 = 0.254546390

Throughput = Total data transferred / time taken to transfer the data  
 = 152976/0.254546390  
 = 6009754.63  
 = 600 KBps



## PART-B

Ans 1:



Here IP address of client computer (source) is **10.0.2.15**

And Source Port is **48210**

Ans 2:

Here IP address of the <https://cse.iith.ac.in/> is **35.185.44.232**

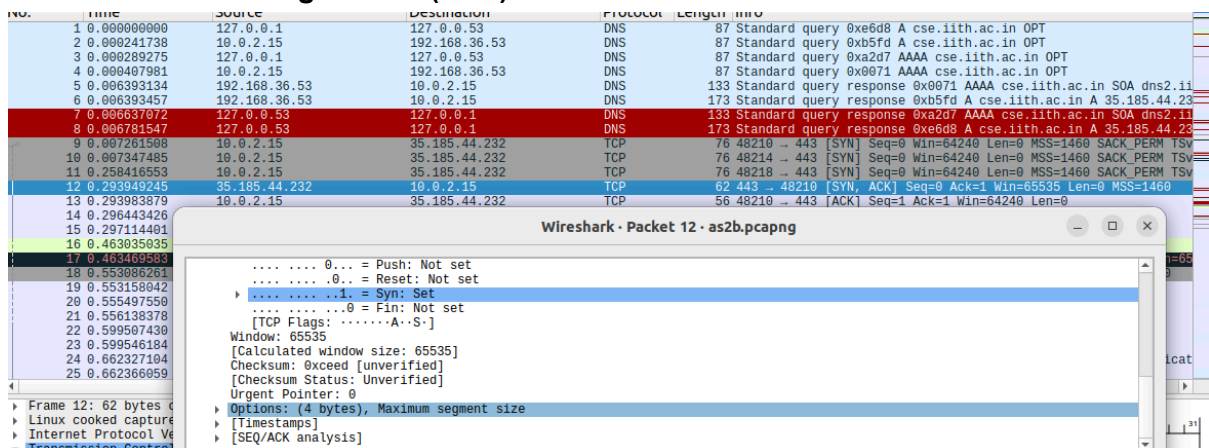
And Destination Port is **443**

Ans 3:

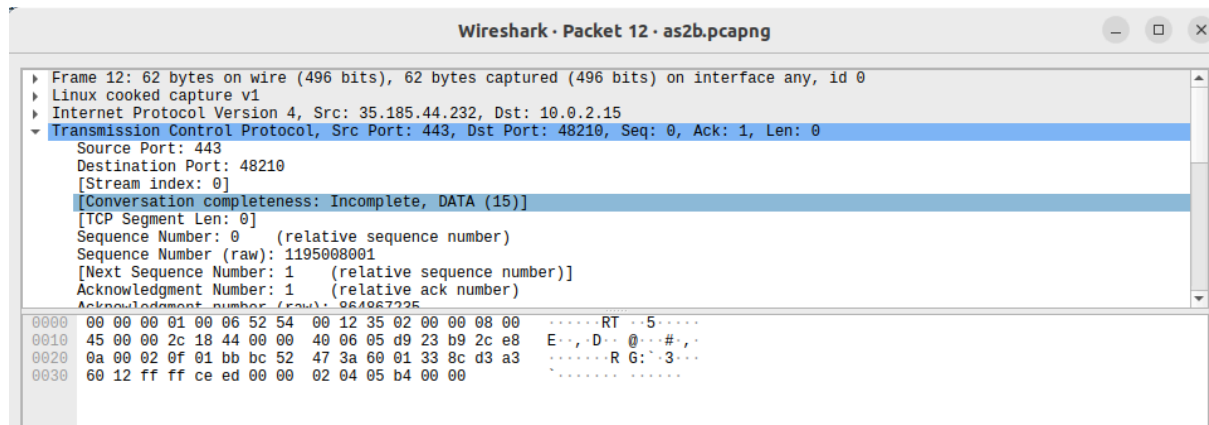
Here we can see the sequence Number of the TCP SYN segment that is used to initiate the TCP connection between the client computer and <https://cse.iith.ac.in/> is **864867234**

Where we can see that the relative sequence number is 0.

Because of control **Flags 0x002 (SYN)**



Here we can see the client computer (source) in SYN segment in options allow the permission of SACK  
 were <https://cse.iith.ac.in/> in SYN segment have not given permission for the SACK  
 So we can not use selective Acknowledgments



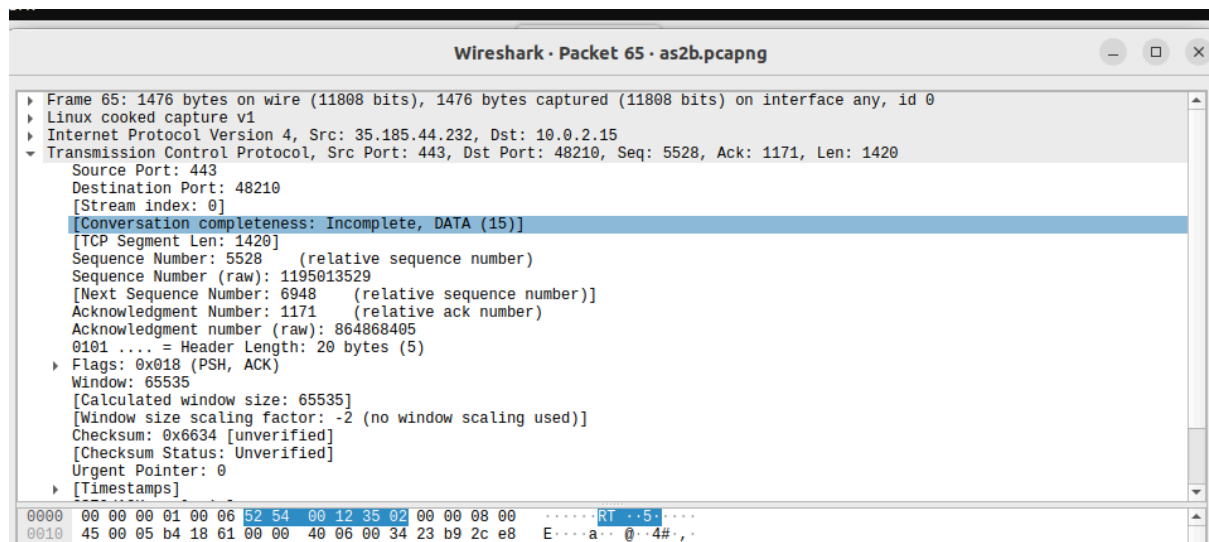
**Ans 4:**

In figure 6 we can see sequence Number (raw) for asked segment is **1195008001**

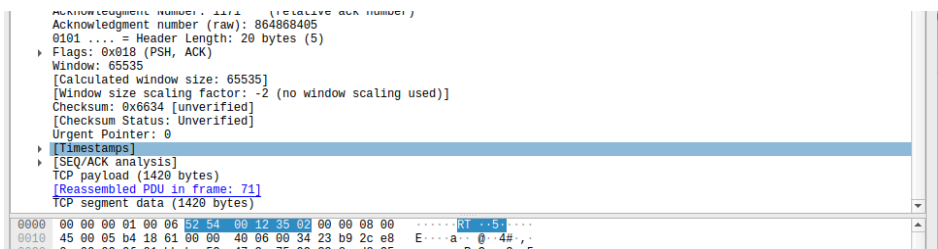
In above figure we can see Flags: 0x012 (SYN, ACK) it shows that the segment identifies as a SYNACK

in figure 6 we can see Acknowledgement number (raw) **864867234** (which is **864867233**+1 where **864867234** is sequence number of SYN segment as shown in figure and 1 because length of the segment is 0 so we want next byte of the message, we can see the length of SYN segment in figure)

**Ans 5:**



Here in the given Sequence number raw of this segment is **1195013529**



Here 1420 bytes of data are contained in the payload (data) field of this TCP  
No, all of the data not transferred into this single segment

Ans 6:

65	1.037344352	35.185.44.232	10.0.2.15	TCP	1476	443 → 48210	[PSH, ACK]	Seq=5528 Ack=1171 Win=65535 Len=1420	[T
67	1.049473025	35.185.44.232	10.0.2.15	TCP	2896	443 → 48210	[PSH, ACK]	Seq=6948 Ack=1171 Win=65535 Len=2840	[T
69	1.052194364	35.185.44.232	10.0.2.15	TCP	1476	443 → 48210	[PSH, ACK]	Seq=9788 Ack=1171 Win=65535 Len=1420	[T
71	1.065615972	35.185.44.232	10.0.2.15	TLSv1.3	1476	Application Data			
73	1.072557233	35.185.44.232	10.0.2.15	TCP	1476	443 → 48210	[PSH, ACK]	Seq=12628 Ack=1171 Win=65535 Len=1420	[T
83	1.082905829	35.185.44.232	10.0.2.15	TCP	1476	443 → 48210	[PSH, ACK]	Seq=14048 Ack=1171 Win=65535 Len=1420	[T
101	1.093122362	35.185.44.232	10.0.2.15	TCP	1476	443 → 48210	[PSH, ACK]	Seq=15468 Ack=1171 Win=65535 Len=1420	[T
103	1.108120649	35.185.44.232	10.0.2.15	TCP	1476	443 → 48210	[PSH, ACK]	Seq=16888 Ack=1171 Win=65535 Len=1420	[T

-> 1.037344352 sec.

64	1.034550990	10.0.2.15	35.185.44.232	TCP	56	48210 → 443	[ACK]	Seq=1171 Ack=5528 Win=62780 Len=0	
66	1.037382264	10.0.2.15	35.185.44.232	TCP	56	48210 → 443	[ACK]	Seq=1171 Ack=6948 Win=62780 Len=0	
68	1.049502853	10.0.2.15	35.185.44.232	TCP	56	48210 → 443	[ACK]	Seq=1171 Ack=9788 Win=62780 Len=0	
70	1.052234488	10.0.2.15	35.185.44.232	TCP	56	48210 → 443	[ACK]	Seq=1171 Ack=12628 Win=62780 Len=0	
72	1.065649238	10.0.2.15	35.185.44.232	TCP	56	48210 → 443	[ACK]	Seq=1171 Ack=14048 Win=62780 Len=0	
74	1.072580444	10.0.2.15	35.185.44.232	TCP	56	48210 → 443	[ACK]	Seq=1171 Ack=15468 Win=62780 Len=0	
84	1.082928866	10.0.2.15	35.185.44.232	TCP	56	48210 → 443	[ACK]	Seq=1171 Ack=16888 Win=62780 Len=0	
102	1.093143509	10.0.2.15	35.185.44.232	TCP	56	48210 → 443	[ACK]	Seq=1171 Ack=18308 Win=62780 Len=0	
104	1.108145978	10.0.2.15	35.185.44.232	TCP	56	48210 → 443	[ACK]	Seq=1171 Ack=18308 Win=62780 Len=0	
106	1.145649825	10.0.2.15	35.185.44.232	TLSv1.3	194	Application Data			
108	1.147741959	10.0.2.15	35.185.44.232	TLSv1.3	139	Application Data			
114	1.153548268	10.0.2.15	35.185.44.232	TLSv1.3	167	Application Data			
121	1.162839873	10.0.2.15	35.185.44.232	TLSv1.3	146	Application Data			
123	1.167020235	10.0.2.15	35.185.44.232	TLSv1.3	146	Application Data			
126	1.175195848	10.0.2.15	35.185.44.232	TCP	56	48210 → 443	[RST]	Seq=794 Win=0 Len=0	
127	1.182906834	10.0.2.15	35.185.44.232	TLSv1.3	146	Application Data			
128	1.183218996	10.0.2.15	35.185.44.232	TLSv1.3	147	Application Data			
147	1.289883663	10.0.2.15	35.185.44.232	TCP	56	48210 → 443	[ACK]	Seq=1864 Ack=21858 Win=62780 Len=0	
149	1.297021456	10.0.2.15	35.185.44.232	TCP	56	48210 → 443	[RST]	Seq=1 Win=0 Len=0	

At what time was the ACK for this first data-containing segment received?

What is the RTT for this first data-containing segment?

Ack of the first segment is got at time **1.037382264 sec.** and RTT for this first data containing segment is **0.000038 sec.**

Rtt for the second segment will be **1.049502853-1.049473025 =0.000029 sec.**

RTT for first packet : **0.000038**

RTT for second packet : **0.000029**

Estimated RTT = (1-alfa) \* Previously estimated RTT + alfa \* Sample RTT

= (1-0.125) \* **0.000038** + (0.125) \* **0.000029**

= **0.000036875**

Ans 7:

63	1.034516430	35.185.44.232	10.0.2.15	TLSv1.3	287	Application Data			
65	1.037344352	35.185.44.232	10.0.2.15	TCP	1476	443 → 48210	[PSH, ACK]	Seq=5528 Ack=1171 Win=65535 Len=1420	[T
67	1.049473025	35.185.44.232	10.0.2.15	TCP	2896	443 → 48210	[PSH, ACK]	Seq=6948 Ack=1171 Win=65535 Len=2840	[T
69	1.052194364	35.185.44.232	10.0.2.15	TCP	1476	443 → 48210	[PSH, ACK]	Seq=9788 Ack=1171 Win=65535 Len=1420	[T
71	1.065615972	35.185.44.232	10.0.2.15	TLSv1.3	1476	Application Data			
73	1.072557233	35.185.44.232	10.0.2.15	TCP	1476	443 → 48210	[PSH, ACK]	Seq=12628 Ack=1171 Win=65535 Len=1420	[T
83	1.082905829	35.185.44.232	10.0.2.15	TCP	1476	443 → 48210	[PSH, ACK]	Seq=14048 Ack=1171 Win=65535 Len=1420	[T
101	1.093122362	35.185.44.232	10.0.2.15	TCP	1476	443 → 48210	[PSH, ACK]	Seq=15468 Ack=1171 Win=65535 Len=1420	[T



as2b.pcapng						
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help						
tcp.analysis.retransmission						
No.	Time	Source	Destination	Protocol	Length	Info
64	1.034550990	10.0.2.15	35.185.44.232	TCP	56	48210 → 443 [ACK] Seq=1171 Ack=5528 Win=62780 Len=0
66	1.037382264	10.0.2.15	35.185.44.232	TCP	56	48210 → 443 [ACK] Seq=1171 Ack=6948 Win=62780 Len=0
68	1.049502853	10.0.2.15	35.185.44.232	TCP	56	48210 → 443 [ACK] Seq=1171 Ack=9788 Win=62780 Len=0
70	1.052234488	10.0.2.15	35.185.44.232	TCP	56	48210 → 443 [ACK] Seq=1171 Ack=1208 Win=62780 Len=0
72	1.065649238	10.0.2.15	35.185.44.232	TCP	56	48210 → 443 [ACK] Seq=1171 Ack=12628 Win=62780 Len=0
74	1.072580444	10.0.2.15	35.185.44.232	TCP	56	48210 → 443 [ACK] Seq=1171 Ack=14048 Win=62780 Len=0
84	1.082928866	10.0.2.15	35.185.44.232	TCP	56	48210 → 443 [ACK] Seq=1171 Ack=15468 Win=62780 Len=0
102	1.093143509	10.0.2.15	35.185.44.232	TCP	56	48210 → 443 [ACK] Seq=1171 Ack=16888 Win=62780 Len=0
104	1.108145978	10.0.2.15	35.185.44.232	TCP	56	48210 → 443 [ACK] Seq=1171 Ack=18308 Win=62780 Len=0

Ans 11:

64	1.034550990	10.0.2.15	35.185.44.232	TCP	56	48210 → 443 [ACK] Seq=1171 Ack=5528 Win=62780 Len=0
66	1.037382264	10.0.2.15	35.185.44.232	TCP	56	48210 → 443 [ACK] Seq=1171 Ack=6948 Win=62780 Len=0
68	1.049502853	10.0.2.15	35.185.44.232	TCP	56	48210 → 443 [ACK] Seq=1171 Ack=9788 Win=62780 Len=0
70	1.052234488	10.0.2.15	35.185.44.232	TCP	56	48210 → 443 [ACK] Seq=1171 Ack=1208 Win=62780 Len=0
72	1.065649238	10.0.2.15	35.185.44.232	TCP	56	48210 → 443 [ACK] Seq=1171 Ack=12628 Win=62780 Len=0
74	1.072580444	10.0.2.15	35.185.44.232	TCP	56	48210 → 443 [ACK] Seq=1171 Ack=14048 Win=62780 Len=0
84	1.082928866	10.0.2.15	35.185.44.232	TCP	56	48210 → 443 [ACK] Seq=1171 Ack=15468 Win=62780 Len=0
102	1.093143509	10.0.2.15	35.185.44.232	TCP	56	48210 → 443 [ACK] Seq=1171 Ack=16888 Win=62780 Len=0
104	1.108145978	10.0.2.15	35.185.44.232	TCP	56	48210 → 443 [ACK] Seq=1171 Ack=18308 Win=62780 Len=0

1420 bytes of data does the receiver typically acknowledge in an ACK among the first ten data-carrying segments sent from the <https://cse.iith.ac.in/> to client I can't identify cases where the receiver is ACKing every other received segment among these first ten data-carrying segments.

Ans 12:

as2b.pcapng						
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help						
ip.addr == 35.185.44.232						
No.	Time	Source	Destination	Protocol	Length	Info
9	0.007261508	10.0.2.15	35.185.44.232	TCP	76	48210 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM TSV
10	0.007347485	10.0.2.15	35.185.44.232	TCP	76	48210 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM TSV
11	0.258416553	10.0.2.15	35.185.44.232	TCP	76	48210 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM TSV
12	0.293949245	35.185.44.232	10.0.2.15	TCP	62	443 → 48210 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460
13	0.293983879	10.0.2.15	35.185.44.232	TCP	56	48210 → 443 [ACK] Seq=1 Ack=1 Win=64240 Len=0
14	0.296443426	10.0.2.15	35.185.44.232	TLSv1.3	573	Client Hello
15	0.297114401	35.185.44.232	10.0.2.15	TCP	62	443 → 48210 [ACK] Seq=1 Ack=518 Win=65535 Len=0
18	0.553080201	35.185.44.232	10.0.2.15	TCP	62	443 → 48210 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460
19	0.553158042	10.0.2.15	35.185.44.232	TCP	56	48210 → 443 [ACK] Seq=1 Ack=1 Win=64240 Len=0
20	0.555497559	10.0.2.15	35.185.44.232	TLSv1.3	573	Client Hello
21	0.556138378	35.185.44.232	10.0.2.15	TCP	62	443 → 48210 [ACK] Seq=1 Ack=518 Win=65535 Len=0
22	0.595907430	35.185.44.232	10.0.2.15	TLSv1.3	2896	Server Hello, Change Cipher Spec, Application Data
23	0.595946184	10.0.2.15	35.185.44.232	TCP	56	48210 → 443 [ACK] Seq=518 Ack=2841 Win=62780 Len=0
24	0.662327104	35.185.44.232	10.0.2.15	TLSv1.3	2385	Application Data, Application Data, Application Data, Applicat
25	0.662366059	10.0.2.15	35.185.44.232	TCP	56	48210 → 443 [ACK] Seq=518 Ack=5170 Win=62780 Len=0
881	3.293647682	35.185.44.232	10.0.2.15	TLSv1.3	9996	Application Data, Application Data
883	3.310542193	35.185.44.232	10.0.2.15	TCP	2896	443 → 48210 [PSH, ACK] Seq=898063 Ack=2565 Win=65535 Len=2840
885	3.313077042	35.185.44.232	10.0.2.15	TLSv1.3	14256	Application Data, Application Data
887	3.322586090	35.185.44.232	10.0.2.15	TCP	4316	443 → 48210 [PSH, ACK] Seq=915103 Ack=2565 Win=65535 Len=4260
889	3.327611723	35.185.44.232	10.0.2.15	TCP	4316	443 → 48210 [PSH, ACK] Seq=919363 Ack=2565 Win=65535 Len=4260
891	3.331730136	35.185.44.232	10.0.2.15	TLSv1.3	8576	Application Data, Application Data
893	3.339360608	35.185.44.232	10.0.2.15	TCP	9996	443 → 48210 [PSH, ACK] Seq=932143 Ack=2565 Win=65535 Len=9940
895	3.341138408	35.185.44.232	10.0.2.15	TCP	1476	443 → 48210 [PSH, ACK] Seq=942083 Ack=2565 Win=65535 Len=1420
881	3.293647682	35.185.44.232	10.0.2.15	TLSv1.3	9996	Application Data, Application Data
883	3.310542193	35.185.44.232	10.0.2.15	TCP	2896	443 → 48210 [PSH, ACK] Seq=898063 Ack=2565 Win=65535 Len=2840
885	3.313077042	35.185.44.232	10.0.2.15	TLSv1.3	14256	Application Data, Application Data
887	3.322586090	35.185.44.232	10.0.2.15	TCP	4316	443 → 48210 [PSH, ACK] Seq=915103 Ack=2565 Win=65535 Len=4260
889	3.327611723	35.185.44.232	10.0.2.15	TCP	4316	443 → 48210 [PSH, ACK] Seq=919363 Ack=2565 Win=65535 Len=4260
891	3.331730136	35.185.44.232	10.0.2.15	TLSv1.3	8576	Application Data, Application Data
893	3.339360608	35.185.44.232	10.0.2.15	TCP	9996	443 → 48210 [PSH, ACK] Seq=932143 Ack=2565 Win=65535 Len=9940
895	3.341138408	35.185.44.232	10.0.2.15	TCP	1476	443 → 48210 [PSH, ACK] Seq=942083 Ack=2565 Win=65535 Len=1420

Total data transferred = 942083-5170-1 = 936912 byte

Know let time t is time difference between first packet sent from client and last ack received at client = 3.341138408 - 0.007261 = 3.33 sec.

Throughput = Total data transferred / time taken to transfer the data

$$= 936912/3.33$$

$$= 281354.95$$

$$= 281.3 \text{ KBps}$$