Problem 1

Suppose an application generates chunks of 60 bytes of data, each chunk gets encapsulated in a TCP segment, and then an IP datagram.

1. What percentage of each datagram will be overhead, and what percentage will be application data?
2. What would be the overhead if each TCP segment include 100 of application chunks (i.e., 100×60 bytes), assuming the maximum size of an IP packet is 500 bytes and sending such big TCP payload would require fragmentation.

1. Each 60 byte datagram will have a 20 byte TCP header and a 20 byte IP header. This means that $\frac{40}{100} = 40\%$ of each datagram is overhead and $\frac{60}{100} = 60\%$ of each datagram is application data.
2. 100 application chunks in a single TCP segment have a single TCP header attached to 6000 bytes of data. However, IP must fragment these pieces of data such that each IP packet (including its header) must not exceed 500 bytes.

   The first IP datagram has a 20 byte IP header, the 20 byte TCP header, and 460 bytes of data (totaling 500 bytes). This results in this IP datagram having $\frac{20+20}{460} = 8.70\%$ overhead.

   The next 11 IP datagrams have a 20 byte IP header and 480 bytes of data. This results in each IP datagram of this structure to have $\frac{20}{480} = 4.17\%$ overhead.

   The last IP datagram has a 20 byte IP header and 260 bytes of data to account for the last chunk so that all 6000 data bytes are accounted for. This results in this IP datagram to have $\frac{20}{260} = 7.69\%$ overhead.

   The overhead of sending 100 application chunks in a TCP segment fragmented into IP datagrams can be calculated with the equation:

   $\frac{\sum TCP_{Overhead} + \sum IP_{Overhead}}{TotalBytesSent} = \frac{20+(13\times20)}{6280} = 4.46\%$ of overhead.

   This means that the overhead per application chunk is $\frac{4.46\%}{100} = 0.0446\%$.

Problem 2

Consider the router trying to send the following IP packet:

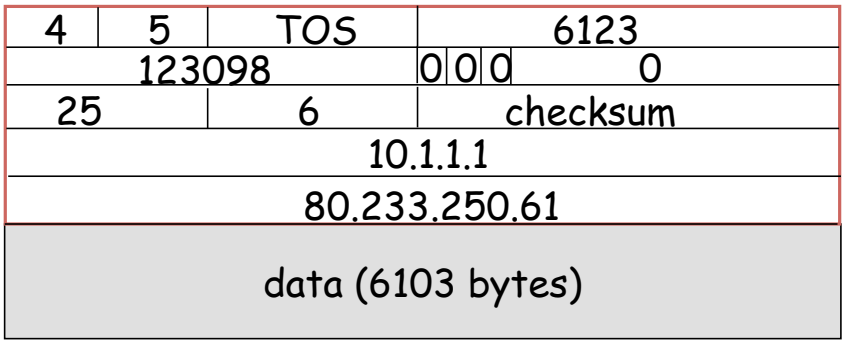

Figure 1: An IP packet.

Assuming that the maximum transmission unit that can be transferred over the link is 1400 bytes. For each of the fragment show the header length, total length, identification, flags, fragment offset, TTL, protocol fields, and IP payload size.

Write your answer using the table below

Header length	Total length	Identification	Flags	Fragment offset	TTL	Protocol	Data size
5	1396	123098	001	0	25	6	1376
5	1396	123098	001	172	25	6	1376
5	1396	123098	001	344	25	6	1376
5	1396	123098	001	516	25	6	1376
5	619	123098	000	688	25	6	599

Problem 3

Calculate the network mask, the number of bits of the network, the number of endpoint addresses in the network (excluding special addresses), the network address, and the broadcast address of the network for the following:

1. 131.179.196.0/24
2. 169.232.34.48/30
3. 196.22.136.0/21
4. 93.181.192.0, netmask 255.255.224.0
5. 10.128.0.0, netmask 255.192.0.0

1.
Network mask: 11111111 11111111 11111111 00000000 (255.255.255.0)
Number of network bits: 24
Number of endpoint addresses (without special addresses): $2^{32-24} - 2 = 2^8 - 2 = 254$
Network address:
11111111 11111111 11111111 00000000 &
10000011 10110011 11000100 00000000
10000011 10110011 11000100 00000000
131.179.196.0
Broadcast address:
10000011 10110011 11000100 00000000 ||
00000000 00000000 00000000 11111111
10000011 10110011 11000100 11111111
131.179.196.255
2.
Network mask: 11111111 11111111 11111111 11111100 (255.255.255.252)
Number of network bits: 30
Number of endpoint addresses (without special addresses): $2^{32-30} - 2 = 2^2 - 2 = 2$
Network address:
11111111 11111111 11111111 11111100 &
10101001 11101000 00100010 00110000
10101001 11101000 00100010 00110000
169.232.34.48
Broadcast address:
10101001 11101000 00100010 00110000 ||
00000000 00000000 00000000 00000011
10101001 11101000 00100010 00110011
169.232.34.51
3.
Network mask: 11111111 11111111 11111000 00000000 (255.255.248.0)
Number of network bits: 21
Number of endpoint addresses (without special addresses): $2^{32-21} - 2 = 2^{11} - 2 = 2046$
Network address:
11111111 11111111 11111000 00000000 &
11000100 00010110 10001000 00000000
11000100 00010110 10001000 00000000
196.22.136.0
Broadcast address:
11000100 00010110 10001000 00000000 ||
00000000 00000000 00000111 11111111
11000100 00010110 10001111 11111111
196.22.143.255

4.
Network mask: 11111111 11111111 11100000 00000000 (255.255.224.0)
Number of network bits: 19
Number of endpoint addresses (without special addresses): $2^{32-19} - 2 = 2^{13} - 2 = 8190$
Network address:
11111111 11111111 11100000 00000000 &
01011101 10110101 11000000 00000000
01011101 10110101 11000000 00000000
93.181.192.0
Broadcast address:
01011101 10110101 11000000 00000000 ||
00000000 00000000 00011111 11111111
01011101 10110101 11011111 11111111
93.181.223.255
5.
Network mask: 11111111 11000000 00000000 00000000 (255.192.0.0)
Number of network bits: 10
Number of endpoint addresses (without special addresses): $2^{32-10} - 2 = 2^{22} - 2 = 4194302$
Network address:
11111111 11000000 00000000 00000000 &
00001010 10000000 00000000 00000000
00001010 10000000 00000000 00000000
10.128.0.0
Broadcast address:
00001010 10000000 00000000 00000000 ||
00000000 00111111 11111111 11111111
00001010 10111111 11111111 11111111
10.191.255.255

Problem 4

Why is the IP header checksum recalculated at every router?

The IP header checksum is recalculated at every router because the TTL field along with the options field may change at each router. Thus, the old checksum performed over the header would no longer match the new header. So a new checksum recomputed at each router is necessary to keep up with the header as it changes throughout its lifetime.

Problem 5

Install *Wireshark* (`https://www.wireshark.org/`). Then, (i) start capturing a packet trace from your network interface, (ii) open a web browser, (iii) go to `https://www.cs.ucla.edu/`, (iv) and then stop capturing the trace.

Investigate any TCP packet from your network interface to the UCLA CS web server. What is the IP address of your network interface? What is the IP address of www.cs.ucla.edu? Provide the screenshot that shows the IP addresses in the investigated packet.

IP address of my network interface: 192.168.0.105
IP address of www.cs.ucla.edu: 164.67.100.181

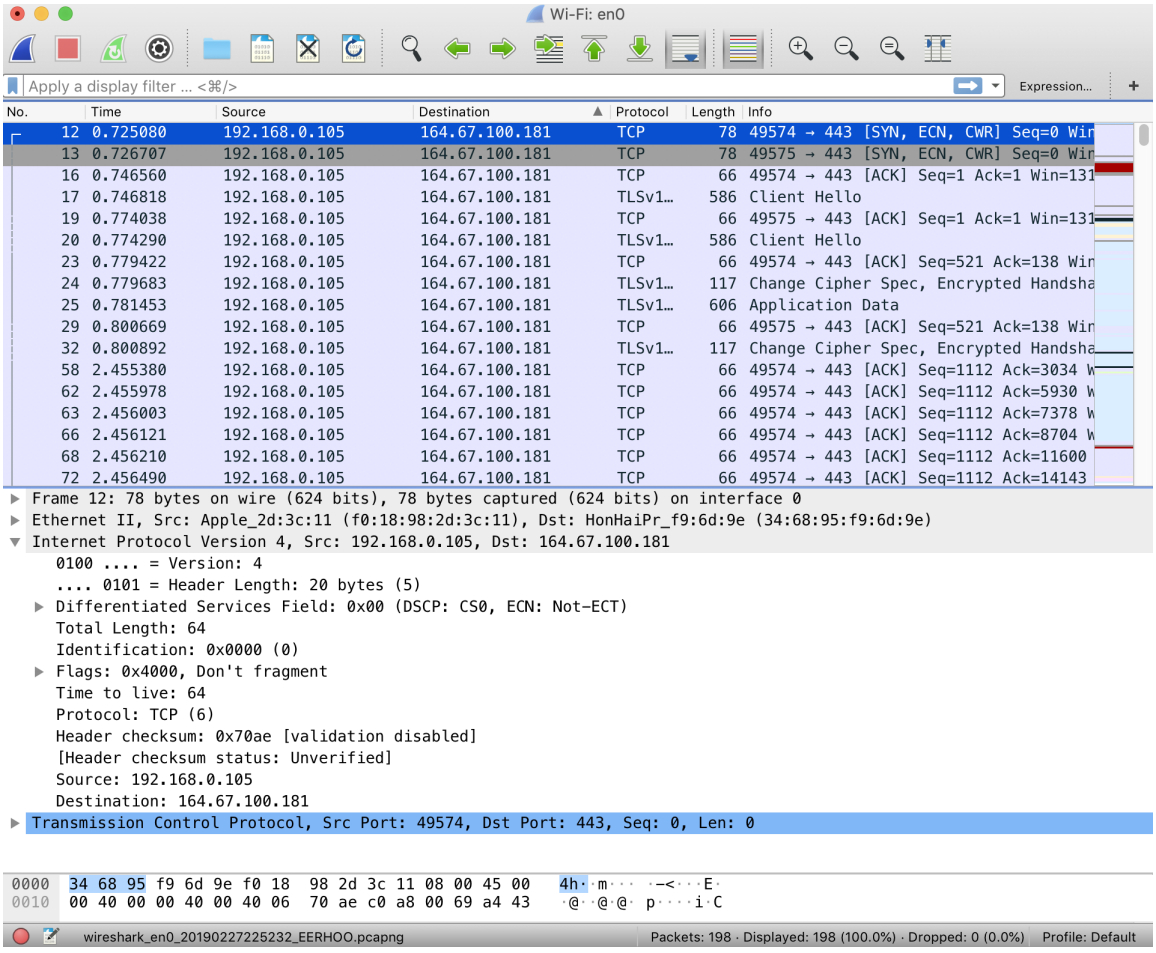

Figure 2: Screenshot of selected packet trace from using a browser to visit https://www.cs.ucla.edu/.