Florida Institute of Technology

CSE5231

Fall 2015

Class Project

Due on: **December 8th, 2015** by midnight

Group submission through CANVAS

GROUP Number: _______________

Student: ____________________________________

Student: ____________________________________

Student: ____________________________________

Student: ____________________________________

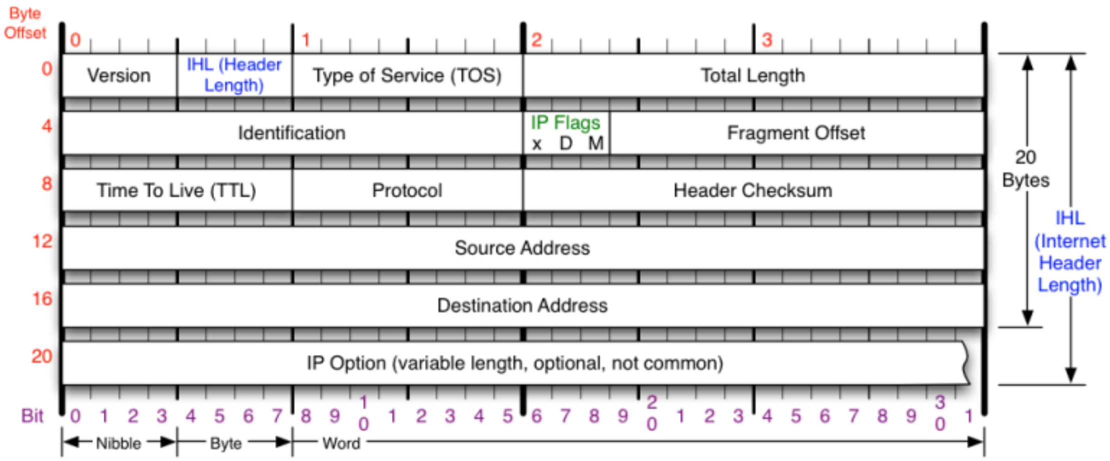
Professor:	**Teaching Assistant:**
Dr. Marco Carvalho	Gireesh Rajulapudi
mcarvalho@fit.edu	grajulapudi2013@my.fit.edu

Florida Institute of Technology
150 W. University Blvd, Melbourne, FL - 32901
http://www.fit.edu

1. Description

Building from your prior programming assignments, complete the simulated network stack you develop to include the following features:

a) Implementation of the IPv4 and Ethernet (Layer 2) headers (shows below).
b) Calculation (and check) of the IPv4 header Checksum
c) Calculation (and check) the checksum of the Ethernet Frame
d) Build the routing tables on each individual nodes based on the provided IP-allocation for each node interface
e) Create a simple de-confliction (medium access control) based on the "busy" signal from the medium.
f) Accept a formatted general input file defining the topology and traffic
g) Generate a formatted output file with the results of the simulation



Version

Version of IP Protocol. 4 and 6 are valid. This diagram represents version 4 structure only.

Header Length

Number of 32-bit words in TCP header, minimum value of 5. Multiply by 4 to get byte count.

Protocol

IP Protocol ID. Including (but not limited to):

1	ICMP	17	UDP	57	SKIP
2	IGMP	47	GRE	88	EIGRP
6	TCP	50	ESP	89	OSPF
9	IGRP	51	AH	115	L2TP

Total Length

Total length of IP datagram, or IP fragment if fragmented. Measured in Bytes.

Fragment Offset

Fragment offset from start of IP datagram. Measured in 8 byte (2 words, 64 bits) increments. If IP datagram is fragmented, fragment size (Total Length) must be a multiple of 8 bytes.

Header Checksum

Checksum of entire IP header

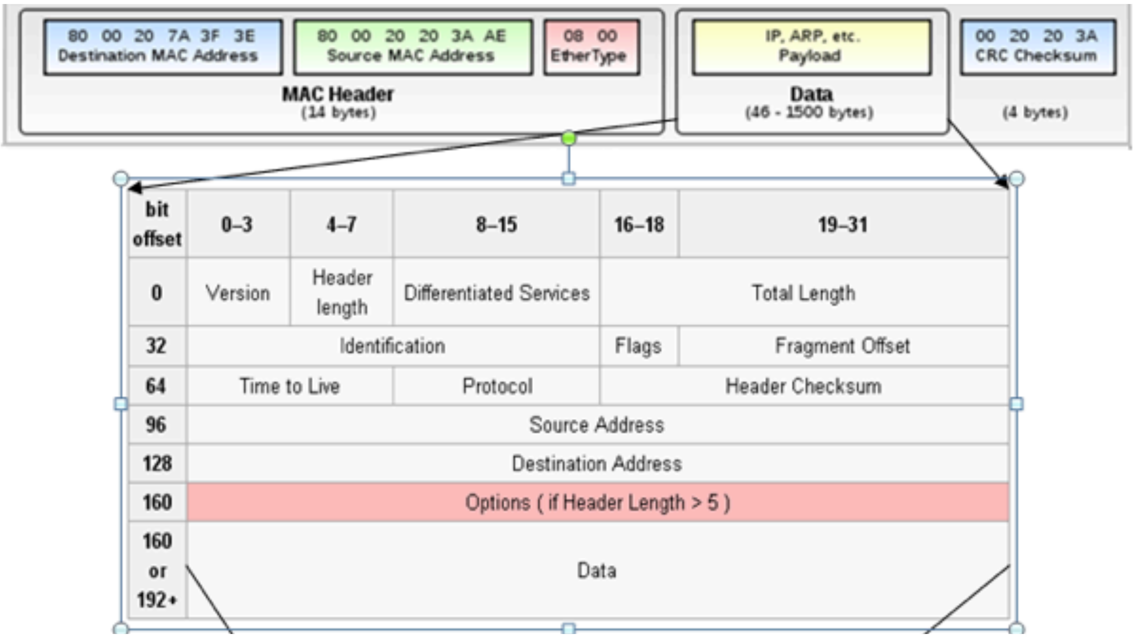
IP Flags

x D M

x 0x80 reserved (evil bit)
D 0x40 Do Not Fragment
M 0x20 More Fragments follow

RFC 791

Please refer to RFC 791 for the complete Internet Protocol (IP) Specification.



bit offset	0–3	4–7	8–15	16–18	19–31
0	Version	Header length	Differentiated Services	Total Length	
32	Identification			Flags	Fragment Offset
64	Time to Live		Protocol	Header Checksum	
96	Source Address				
128	Destination Address				
160	Options (if Header Length > 5)				
160 or 192+	Data				

2. Sample Input File

```
SECTION_MEDIUM_START #-------------------------

NAME: MD1
MD1_MTU: 1400                   #in bytes

NAME: MD2
MD2_MTU: 1400                   #in bytes

NAME: MD3
MD3_MTU: 1500                   #in bytes

SECTION_MEDIUM_END #-------------------------


SECTION_NODE_START #-------------------------

NAME: ND1
ND1_IF1_IP: 10.10.20.10
ND1_IF1_MASK: 255.255.255.0
ND1_IF1_ETHER: 80 00 20 7A 3E
ND1_IF1_BANDWIDTH: 10           #in KBps   (not MBps)
ND1_IF1_CONNECTION: MD1

NAME: ND2
ND2_IF1_IP: 10.10.20.11
ND2_IF1_MASK: 255.255.255.0
ND2_IF1_ETHER: 80 00 10 7A 2E
ND2_IF1_BANDWIDTH: 10           #in KBps   (not MBps)
ND2_IF1_CONNECTION: MD1

ND2_IF2_IP: 10.10.40.12
ND2_IF2_MASK: 255.255.255.0
ND2_IF2_ETHER: 80 00 20 7B 0E
ND2_IF2_BANDWIDTH: 10           #in KBps   (not MBps)
ND2_IF2_CONNECTION: MD2


SECTION_NODE_END #-------------------------


SECTION_DATA_START #-------------------------

# JOB numner, at time 0 ms, node ND1 starts transmitting 100 Kbytes (not MBytes) to node N2
FILE    TIME    NODE_SRC      NODE_DEST    FILE_SIZE (KB)
0       0       ND1           ND2          80
1       100     ND2           ND1          150
2       110     ND1           ND4          200
....
SECTION_DATA_END #-------------------------
```

3. Sample Output File

You're free to make adjustments to provide more information, if necessary – but please make sure the file is well structured and easy to read.

```
Time      Node/Medium         Event
0         N1                  Starts JOB1 80 KB
0         N1                  Fragments JOB1 into 4 parts (25 KB, 25 KB, 25 KB, 5 KB)
0         N1                  Starts Transmitting J1F1 (Frame 1 of Job 1) to N2
                              [Print Frame Header in Hex]
                              -------------------------------------------------------------------------------------
                              | 6 Bytes (SRC MAC)                                                                 |
                              -------------------------------------------------------------------------------------
                              | 6 Bytes (DEST MAC)                                                                |
                              -------------------------------------------------------------------------------------
                              | 2 Bytes (TYPE)                      |
                              --------------------------------------
                              -------------------------------------------------------------------------
                              | 4-bit   | 4-bit header |8-bit "type | 16-bit total length (in bytes)  |
                              | version | length       |of service" |                                 |
                              -------------------------------------------------------------------------
                              | 16-bit identification               | 3-bit | 13-bit fragment offset  |
                              |                                     | flag  |                         |
                              -------------------------------------------------------------------------
                              | 8-bit "time to live"    | 8-bit       |  16-bit header checksum        |
                              |                        | protocol    |                                 |
                              -------------------------------------------------------------------------
                              |         32-bit source IP address                                      |
                              -------------------------------------------------------------------------
                              |         32-bit destination IP address                                 |
                              -------------------------------------------------------------------------

                              -------------------------------------------------------------------------
                              |         CRC                                                           |
                              -------------------------------------------------------------------------

0         M1                  Enters BUSY state
105       N1                  Finishes Transmitting 100 KB
105       N3                  Receives J1F1 (Dropped Frame – N2's MAC address)
105       N2                  Receives J1F1 (Accepted Frame – N2's MAC address, Checksum: Check, CRC: Check))
                              [Print Frame Header in Hex]
                              -------------------------------------------------------------------------------------
                              | 6 Bytes (SRC MAC)                                                                 |
                              -------------------------------------------------------------------------------------
                              | 6 Bytes (DEST MAC)                                                                |
                              -------------------------------------------------------------------------------------
                              | 2 Bytes (TYPE)                      |
                              --------------------------------------
                              -------------------------------------------------------------------------
                              | 4-bit   | 4-bit header |8-bit "type | 16-bit total length (in bytes)  |
                              | version | length       |of service" |                                 |
                              -------------------------------------------------------------------------
                              | 16-bit identification               | 3-bit | 13-bit fragment offset  |
                              |                                     | flag  |                         |
                              -------------------------------------------------------------------------
                              | 8-bit "time to live"    | 8-bit       |  16-bit header checksum        |
                              |                        | protocol    |                                 |
                              -------------------------------------------------------------------------
                              |         32-bit source IP address                                      |
                              -------------------------------------------------------------------------
                              |         32-bit destination IP address                                 |
                              -------------------------------------------------------------------------

                              -------------------------------------------------------------------------
                              |         CRC                                                           |
                              -------------------------------------------------------------------------

105       M1                  Enters FREE state

…
```

4. Deliverables

a) Documented Program code (on GITLab) – each deliverable should be tagged in the repository.

b) Brief report submitted through CANVAS (one per group), including:

   a. Instructions on how to checkout, compile, and run the program.

   b. A description of the program. Please use a diagram to describe the functionality of your application, and explain the output.

   c. A link to the code repository and the TAG name used for the specific deliverable.

5. Requirements

a) Application will be built and executed on a Linux (Ubuntu) server. All student-groups will have access to server to test their code.

b) The ONLY argument to the application should be a path to the configuration file. The format of the configuration file will be the same for all groups.

c) The application should provide detailed logging of all the activities performed by the threads (nodes), and the communications medium. A common time references (in milliseconds) kept by the main application is recommended, and timestamps should be referential to the start of the program.

Florida Institute of Technology
150 W. University Blvd, Melbourne, FL - 32901
http://www.fit.edu