

What is the value of each of the header fields ? Explain why the value is what it is .

Version(4 bits) value =4. The value 4 means that the ip version is 4 i.e. IPv4.

Header Length (4 bits) value =5 . Indicates the number of 32-bit words in the header.
5 means the header is 20 bytes long (5 * 4 bytes).

DSCP (6 bits) Value= 0x80 . Value 0x80 indicates a DSCP value corresponds to a DSCP value of 32, which is associated with the Expedited Forwarding (EF) Per-Hop Behavior.

ECN (2 bits) Value =00 , This value indicates that the packet is not using ECN. It is not ECN-capable, and any network devices should not set or interpret ECN-related bits for this packet.

Total Length (2 bytes) Value =1228 , This value indicates that the entire packet, including the header and data is 1228 bytes long .

Identification (2 bytes) Value =0x0000, This value indicates that the identification value of the packet is 0 .

Flags (3 bits) Value=0x2, Don't fragment. The value 0x2 identifies as Don't fragment.

Fragment Offset(13 bits) , Value = 0, Indicates where in the datagram this fragment belongs , 0 means not fragmented.

TTL <Time to live> (1 byte) , Value=58 , Indicates it can pass through 58 routers before getting discarded. MacOS or GNU/Linux often use a default TTL of 64 which means packet has already passed through 6 routers in my case.

Protocol (1 byte) , Value : UDP(17) , It means User Datagram Protocol (UDP) is being used in the data portion of the IP packet.

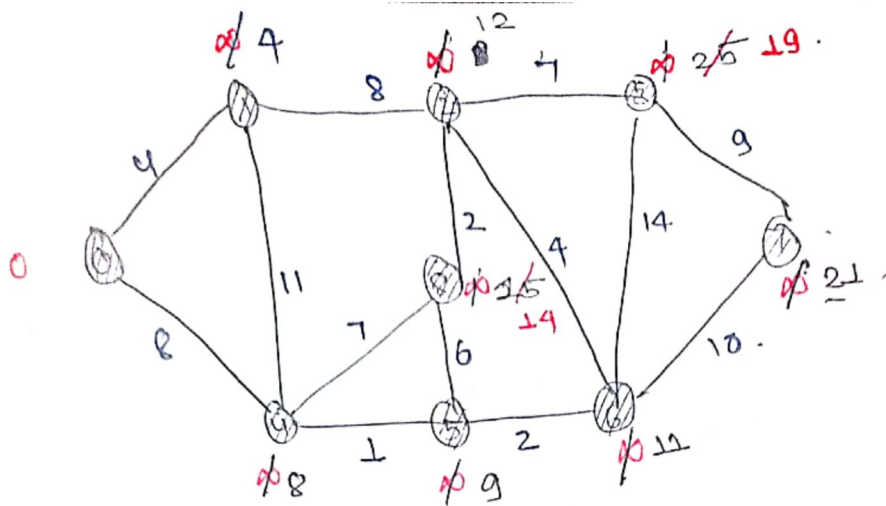
Header Checksum (2 bytes) , value : 0x27b1, It indicates whether the sent packets and recieved packets are same or not , if not same then some data is lost.

Source Address (4 bytes) value: 192.168.0.114 , This is the IP address of the device that sent the packet

Destination Address (4 bytes) value: 52.113.194.132, This is the IP address of device receiving the packet .

2)On leetcode find a problem that can be solved with Dijkstra's algorithm and solve it

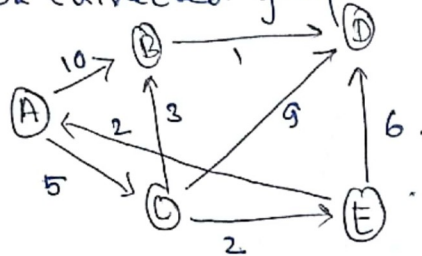
problem:



$$\text{if } \text{dis}(u,v) = \text{dis}(u) + c(u,v) < \text{dis}(v)$$

$$\text{then } \text{dis}(v) = \text{dis}(u) + c(u,v)$$

ex (directed graph).



visited vertex	A	B	C	D	E
source vertex = A	0	∞	∞	∞	∞
C		10	5	∞	∞
E		8		14	7
B		8		13	
				9	

Shortest dist

$$A-D = 9$$

$$A-E = 7$$

$$A-C = 5$$

$$A-B = 8$$

3

$$\text{path} = DBCA. = 5 + 3 + 1 = 9.$$

$$(A-D)$$

so

$$\text{shortest path} = ACBD.$$

$$\text{shortest path } A-B = BCA = ACB.$$

$$= 5 + 3 = 8.$$

$$\text{if } (\text{dis}(u) + c(u,v) < \text{dis}(v))$$

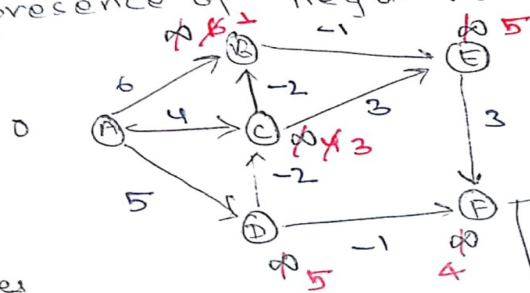
$$\text{dist}(v) = \text{dis}(u) + c(u,v)$$

0.0

3) On leetcode find a problem that can be solved with Bellman-Ford algorithm and solve it. For each of your solution build a flowchart and a table that shows all the updates to all your variables

Bellman ford algorithm:-

(single source shortest path algorithm).
 → confirms that your shortest path is correct even in presence of - negative weight.



go on relaxing all edges $(n-1)$ times.
 $n = \text{no of vertices}$.

if $d[u] + c(u,v) < d[v]$ then
 $d[v] = d[u] + c(u,v)$

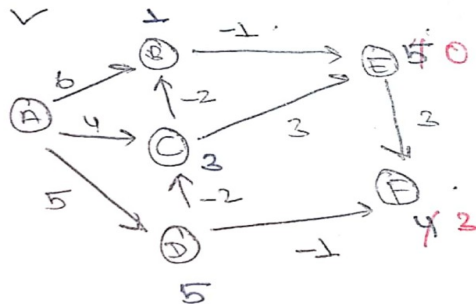
edges

(A,B), (A,C), (A,D), (B,E), (C,E), (C,D), (D,E), (D,F), (E,F), (C,B), (D,C).

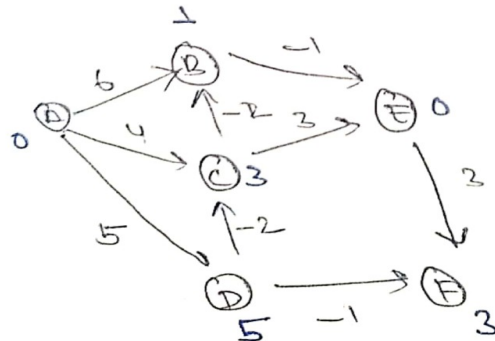
relaxing

1st :- v

2nd :-



3rd :-



(no change)
 so over.