NAME :

: HIMANSHU NETAM (2019BITE039)

: MOHITH (2019BITE038)

: VISHAL KUMAR (2019BITE006)

COMPUTER NETWORKS PROJECT ON
NETWORK SIMULATOR (USING PHYSICAL AND DATA LINK LAYER)

Language used : PYTHON

Platform : PyCharm , Jupyter Notebook

Library used : Python libraries like Time, Random, Threading

What is our approach :

We created devices using objects by creating classes (blueprints) for every device namely Hubs, Switches etc. We interconnected the devices and created a topology by creating a Topology class(It is a kind of head class where we can connect different devices.

Protocols implemented:

Access Control : Token Passing

Flow Control : Stop and Wait and still working on Go Back N

INPUT:

```
topology1 = Topology()


D0 = devices(topology1.td , create_mac_address(),0)
```

```
topology1.add_device_device(D0)  #0

D1 = devices(topology1.td,create_mac_address(),0)
topology1.add_device_device(D1)  #1


D2 = devices(topology1.td,create_mac_address(),1)
topology1.add_device_device(D2)  #2


D3 = devices(topology1.td,create_mac_address(),1)
topology1.add_device_device(D3)  #3


D4 = devices(topology1.td,create_mac_address(),2)
topology1.add_device_device(D4)  #4


D5 = devices(topology1.td,create_mac_address(),2)
topology1.add_device_device(D5)  #5

topology1.add_device_hub(hub_device(topology1.td, 0))  #6
topology1.add_device_hub(hub_device(topology1.td, 1))  #7
topology1.add_device_hub(hub_device(topology1.td, 2))  #8
```

```
S1 = Switch(topology1.td,create_mac_address(),[0,1,2])
topology1.add_device_switch(S1)   #9

token_gen = threading.Thread(target=get_token, args=(topology1.num_devices,))
token_gen.start()

#making connections betweeen the  devs
topology1.make_connection_between(7, 2)
topology1.make_connection_between(3, 7)
topology1.make_connection_between(1, 6)
topology1.make_connection_between(0, 6)
topology1.make_connection_between(4,8)
topology1.make_connection_between(5,8)

#switching the switch
topology1.make_connection_between(6,9)
topology1.make_connection_between(7,9)
topology1.make_connection_between(9,8)

topology1.stop_and_wait(D2,D4,"UwU OwO >w< wakuwak")
```