Prem Raj Oli (@02992540)

traceroute_at_location.py -> take location as input and find the traceroute to the given IP address.

traceroute_prefix.py -> creates active_ip_sets.txt and trace_results.txt file. route_results.txt contains traceroute from prefix.0.0 to prefix.255.255 and active_ip_sets.txt contains sets of traceroute.

all_active_ips.py -> reads trace_results.txt file and creates routes.txt file. routes.txt contains IP addresses to be included in the topology (routers, switches, or other networking devices). Also, traceroute for each IP address is separated with Line End: line.

topology.py -> reads routes.txt and creates the .png and .md file.

Note**:**

Script files are Organized in GitHub at Code/Prem Folder.

Pradeep Lamichhane (@02967686)

chem-sn116_topology.py → Reads .txt file and creates the .png and .md file. Used specifically for chemistry building. Later used topology.py to read routes and create a .png file for different subnets.

trace_ip.py → Create unique_ip.txt file and result.txt file. result.txt file contains traceroute from prefix 0.0 to prefix 255.255. unique_ip.txt file creates a set of traceroute.

trace_ip_updated.py → Updated code to track traceroute from prefix 0 to 255 and set of traceroute.

ugl-sn199_topology.py → Creates the .png and .md file for UGL. Later used topology.py to create a better .png file.

Sanjaya Subedi (@02988775)

Data/Sanjay/start.py → Created to find the private subnet of Howard university iLab.

Code/Sanjay/trace_route.py →Under all of the subnets, this was used to find all of the router ip addresses and created a location_ilab.txt that consists of all the found ip addresses.

Topology/Sanjay/Final_iLab/iLab_all.py → It reads all data of iLab from location_ilab.txt and creates a new file active_ips_from_file.txt. Again from active_ips_from_file.txt, it creates routes.txt that has the final list of router ip addresses.

Topology/Sanjay/Final_iLab/correct_topology.py → This file graphs the topology reading from the routes.txt importing networkx and using spring_layout algorithm.

Ujjwal Adhikari (@02962931)

Data/Ujjwal/trace_ip_stokes_library.py → program to run the trace_route function by changing ipaddr prefix accordingly

Data/Ujjwal/ChemistryBuilding/ChemistryBuildingActiveIPs.py → The code parses traceroute results, filters active IP addresses, and organizes them into routes, enabling structured analysis of network paths.

Data/Ujjwal/ChemistryBuilding/chemistry_building_topology.py / Data/Ujjwal/StokesLibrary/netw.py → This code parses traceroute results, filters specific routers, constructs a network graph, visualizes the topology, saves it as an image, and generates a corresponding Markdown file for documentation.

Data/Ujjwal/ChemistryBuilding/chemistry_building_topology_create.py → Updated code that constructs a graph representing network topology from traceroute data, visualizes it using NetworkX, and saves the resulting graph as an image while generating a Markdown document for documentation.

Manish Niure(@02969739)

Data/Manish/trace_route.py → program to run trace_route function and write files in respective .txt files

Data/ManishSubnet1/make_topologgy → make_topolgy creates the network topology by reading the designated txt files and create md file to

Data/Manish/Uglsubnet26/ugl_topology → Reads data from uglsubne26 and creates topology (later make_topolgy was used which is more effective)