**Social Media Analytics Lab**

**Instructor: Dr. Janani Selvaraj**

**Lab 6. Creating a network graph and perform graph operations**

In this lab you will create a network graph and calculate the centrality measures of a graph

**Create a network graph using the networkx package**

Install the networkx package

import networkx as nx # pip install networkx

# Create a directed graph

g = nx.DiGraph()

# Add an edge to the directed graph from X to Y

g.add\_edge('X', 'Y')

# Print some statistics about the graph

print(nx.info(g))

# Get the nodes and edges from the graph

print("Nodes:", g.nodes())

print("Edges:", g.edges())

print()

# Get node properties

print("X props:", g.node['X'])

print("Y props:", g.node['Y'])

print()

# Get edge properties

print("X=>Y props:", g['X']['Y'])

print()

# Update a node property

g.node['X'].update({'prop1' : 'value1'})

print("X props:", g.node['X'])

print()

# Update an edge property

g['X']['Y'].update({'label' : 'label1'})

print("X=>Y props:", g['X']['Y'])

**Constructing an ego graph of a repository and its stargazers**

g = nx.DiGraph()

g.add\_node(repo.name + '(repo)', type='repo', lang=repo.language, owner=user.login)

for sg in stargazers:

g.add\_node(sg.login + '(user)', type='user')

g.add\_edge(sg.login + '(user)', repo.name + '(repo)', type='gazes')

## Perform handy graph operations

print(nx.info(g))

print(g.node['Mining-the-Social-Web(repo)'])

print(g.node['ptwobrussell(user)'])

print(g['ptwobrussell(user)']['Mining-the-Social-Web(repo)'])

print(g['ptwobrussell(user)'])

print(g['Mining-the-Social-Web(repo)'])

print(g.in\_edges(['ptwobrussell(user)']))

print(g.out\_edges(['ptwobrussell(user)']))

print(g.in\_edges(['Mining-the-Social-Web(repo)']))

print(g.out\_edges(['Mining-the-Social-Web(repo)']))