**PRACTICAL NO 8**

**AIM:** Link Analysis and PageRank

* Implement the PageRank algorithm to rank web pages based on link analysis.
* Apply the PageRank algorithm to a small web graph and analyze the results.

**INPUT:**

import numpy as np

def page\_rank(graph, damping\_factor=0.85, max\_iterations=100, tolerance=1e-6):

num\_nodes = len(graph)

page\_ranks = np.ones(num\_nodes) / num\_nodes

for \_ in range(max\_iterations):

prev\_page\_ranks = np.copy(page\_ranks)

for node in range(num\_nodes):

incoming\_links = [i for i, v in enumerate(graph) if node in v]

if not incoming\_links:

continue

page\_ranks[node] = (1 - damping\_factor) / num\_nodes + \

damping\_factor \* sum(prev\_page\_ranks[link] / len(graph[link]) for link in incoming\_links)

if np.linalg.norm(page\_ranks - prev\_page\_ranks, 2) < tolerance:

break

return page\_ranks

if \_\_name\_\_ == "\_\_main\_\_":

web\_graph = [

[1, 2],

[0, 2],

[0, 1],

[1, 2],

]

result = page\_rank(web\_graph)

for i, pr in enumerate(result):

print(f"Page {i}: {pr}")

**OUTPUT:**

