A. Implementing PageRank Algorithm :-

1. Implementing Column Stochastic Matrix:-

   Import generateMatrix from stochasticMatrix.py to create a column stochastic matrix(probability matrix). It takes a nested list of tuples with directed edges between the links with number of website links and number of edges to form a column stochastic matrix in form of a 2D numpy array.

Code:-

```
from schotasticMatrix import generateMatrix
edges = [(1,2),(2,1),(3,2),(2,3),(3,4),(4,3)]
numberOfLinks = 4
numberOfEdges = 6
print(generateMatrix(edges,numberOfLinks,numberOfEdges))
```

Result:-

```
PS C:\Users\rishi\Desktop\pagerank> python -u "c:\Users\rishi\Desktop\pagerank\main.py"
[[0.  0.5 0.  0. ]
 [1.  0.  0.5 0. ]
 [0.  0.5 0.  1. ]
 [0.  0.  0.5 0. ]]
```

2. Implementing PageRank algorithm using direct method (i.e solving for the eigen vector/rank vector using python libraries) :-

   Import directMethod from Pagerank.py to implement PageRank algorithm using direct method. It takes a Probability matrix and returns ranking vector in form of numpy array.

Ex:-

Code:-

```
from Pagerank import directMethod
from schotasticMatrix import generateMatrix
import numpy as np
probabilityMatrix = generateMatrix([(1,2),(2,1),(3,2),(2,3),(3,4),(4,3)],4,6)
print(directMethod(probabilityMatrix))
```

Result:-

```
[16.66666667 33.33333333 33.33333333 16.66666667]
```

3. Implementing PageRank algorithm using power iteration method :-

   Import powerIteration from PageRank.py to implement PageRank algorithm using power Iteration method. It takes a Probability matrix and probability to follow a link at random by a random surfer and returns ranking vector in form of numpy array.

   Code:-

```
from Pagerank import powerIteration
from schotasticMatrix import generateMatrix
import numpy as np
probabilityMatrix = generateMatrix([(1,2),(2,1),(3,2),(2,3),(3,4),(4,3)],4,6)
print(powerIteration(probabilityMatrix,0.9))
```

   Result:-

```
PS C:\Users\rishi\Desktop\pagerank> python -u "c:\Users\rishi\Desktop\pagerank\main.py"
[17.24019048 32.75980952 32.75980952 17.24019048]
```