PSTAT 160B Homework 2

Harrison Hansen, Jiayue Chen, Jack Liu, Evan Pei, Hunter Marshall

February 10, 2023

Exercise 1

For both graphs in the example do the following. Find the adjacency matrix A and compute A^2 and A^3. Confirm that the entry of the matrix A^2 (and A^3) corresponding to vertices v and y equals the number of ways to walk from vertex v to vertex y in 2 (and 3 for A^3) steps, by enumerating these walks (i.e., write each walk as a sequence of vertices).

The adjacency matrix of the directed graph:

$$A_D = \begin{array}{c c} & \begin{array}{cccc} u & v & y & z \end{array} \\ \begin{array}{c} u \\ v \\ y \\ z \end{array} & \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \end{array}$$

$$A_D^2 = A_D * A_D = \begin{array}{c c} & \begin{array}{cccc} u & v & y & z \end{array} \\ \begin{array}{c} u \\ v \\ y \\ z \end{array} & \begin{pmatrix} 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \end{pmatrix} \end{array}$$

$$A_D^3 = A_D^2 * A_D = \begin{array}{c c} & \begin{array}{cccc} u & v & y & z \end{array} \\ \begin{array}{c} u \\ v \\ y \\ z \end{array} & \begin{pmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 \end{pmatrix} \end{array}$$

The adjacency matrix of the undirected graph:

$$A_U = \begin{array}{c c} & \begin{array}{cccc} u & v & y & z \end{array} \\ \begin{array}{c} u \\ v \\ y \\ z \end{array} & \begin{pmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 2 \\ 0 & 0 & 2 & 0 \end{pmatrix} \end{array}$$

$$A_U^2 = A_U * A_U = \begin{array}{c c} & \begin{array}{cccc} u & v & y & z \end{array} \\ \begin{array}{c} u \\ v \\ y \\ z \end{array} & \begin{pmatrix} 2 & 1 & 1 & 2 \\ 1 & 2 & 1 & 2 \\ 1 & 1 & 6 & 0 \\ 2 & 2 & 0 & 4 \end{pmatrix} \end{array}$$

$$A_U^3 = A_U^2 * A_U = \begin{array}{c c} & \begin{array}{cccc} u & v & y & z \end{array} \\ \begin{array}{c} u \\ v \\ y \\ z \end{array} & \begin{pmatrix} 2 & 3 & 7 & 2 \\ 3 & 2 & 7 & 2 \\ 7 & 7 & 2 & 12 \\ 2 & 2 & 12 & 0 \end{pmatrix} \end{array}$$

For any A^k, the entry $(A^k)_{ij}$ is the number of ways to walk from vertex i to vertex j in k steps. We will prove this in Exercise 5.

For the directed graph, the entry of the matrix A_D^2 from vertex v and y is 0, or $(A_D^2)_{vy} = 0$, which is the number of ways to walk from vertex v to vertex y in 2 steps. We will arrive at u or v in 2 steps when we starts at v, so it confirms there is no way to arrive vertex y from v in two steps.

The entry of the matrix A_D^3 from vertex v and y is 1, or $(A_D^3)_{vy} = 1$. The path arriving to y from v in 3 steps is $v \to y \to z \to y$, and it's the only possible walk.

For the undirected graph, the entry of the matrix A_U^2 from vertex v and y is 1, or $(A_U^2)_{vy} = 1$. The path starts from vertex v to arrive at vertex y in 2 steps is as follow: $v \to u \to y$

The entry of the matrix A_U^3 from vertex v and y is 7, or $(A_U^3)_{vy} = 7$. This means there are 7 paths starts from vertex v to arrive at vertex y in 3 steps.

$$v \to y \to z \to y$$

This path includes 4 ways to walk through as vertex y and z are connected by 2 edges, let the first edge numbered as ① and the second edge numbered as ② and the 7 paths can be listed as following:

$$v \to y \xrightarrow{①} z \xrightarrow{①} y$$

$$v \to y \xrightarrow{②} z \xrightarrow{②} y$$

$$v \to y \xrightarrow{①} z \xrightarrow{②} y$$

$$v \to y \xrightarrow{②} z \xrightarrow{①} y$$

$$v \to y \to v \to y$$

$$v \to y \to u \to y$$

$$v \to u \to v \to y$$

Exercise 2

Let D be a diagonal matrix with the ith diagonal entry equal to the sum of the entries of the ith row of the matrix A. For each A in the previous exercise find D and compute $D^{-1}A$. Explain why the latter is a one-step transition probability matrix for a Markov chain on $\{u, b, y, z\}$.

For the directed graph:

$$D_D = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$$D_D^{-1} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & \frac{1}{2} & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$$D_D^{-1}A_D = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ \frac{1}{2} & 0 & 0 & \frac{1}{2} \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

For the undirected graph:

$$D_U = \begin{pmatrix} 2 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 \\ 0 & 0 & 4 & 0 \\ 0 & 0 & 0 & 2 \end{pmatrix}$$

$$D_U^{-1} = \begin{pmatrix} \frac{1}{2} & 0 & 0 & 0 \\ 0 & \frac{1}{2} & 0 & 0 \\ 0 & 0 & \frac{1}{4} & 0 \\ 0 & 0 & 0 & \frac{1}{2} \end{pmatrix}$$

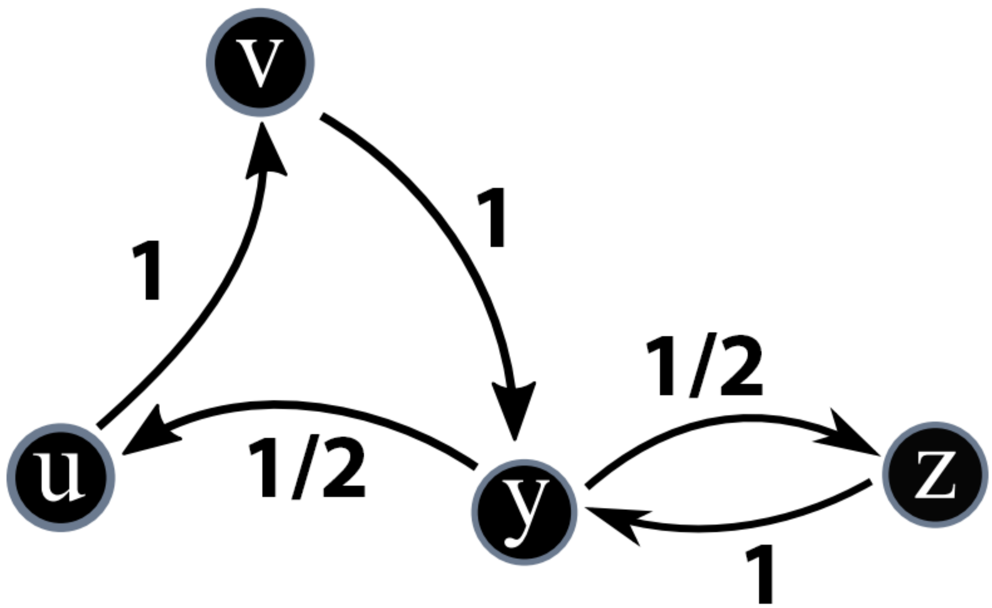
$$D_U^{-1}A_U = \begin{pmatrix} 0 & \frac{1}{2} & \frac{1}{2} & 0 \\ \frac{1}{2} & 0 & \frac{1}{2} & 0 \\ \frac{1}{4} & \frac{1}{4} & 0 & \frac{1}{2} \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

In this exercise, D_D and D_U are the Degree matrices of the graphs, which contain information about the degree of each vertex, the number of edges attached to each vertex. The inverse of a diagonal matrix is given by replacing the main diagonal elements of the matrix with their reciprocals. So, $D^{-1}A$ effectively multiplies each row of A by the reciprocal of its sum, so that it sums to 1. Therefore it becomes a one-step transition probability matrix for a Markov chain on $\{u, b, y, z\}$.

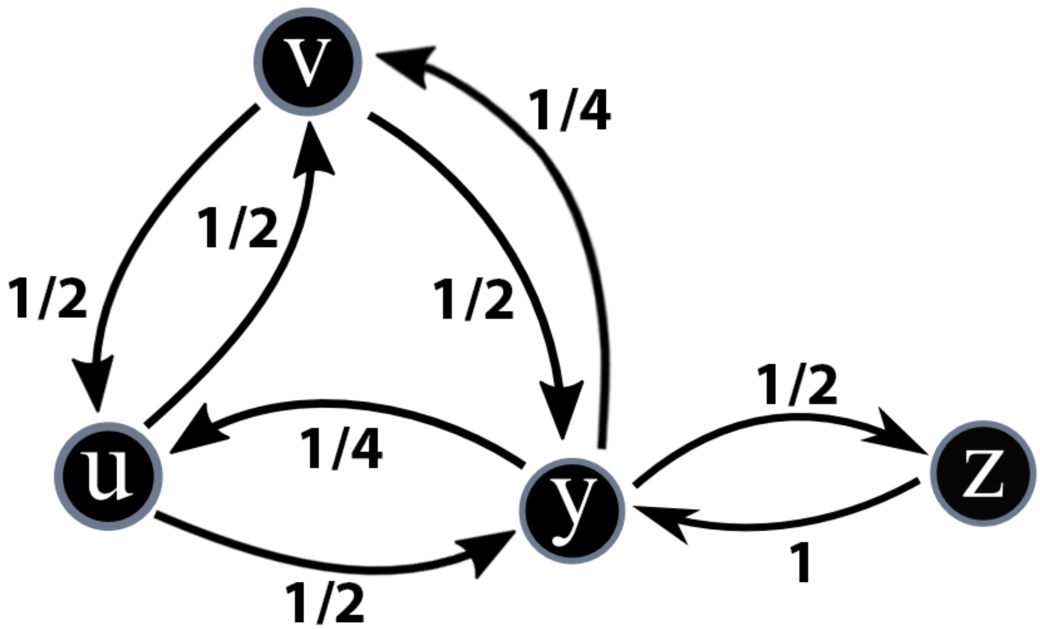
Exercise 3

Draw a transition probability diagram and compute the stationary distribution for each Markov chain of the previous exercise. Classify each of two chains as reversible or nonreversible by showing whether local balance holds or does not.For the nonreversible cases explain how the "playing the movie forward/backward" analogy confirms your result.

Transition probability diagram for the directed graph:



Transition probability diagram for the undirected graph:



The stationary distribution for direct graph can be computed as follows, where $P_D = D_D^{-1}A_D$:

$$\pi * P_D = \pi$$

$$\begin{pmatrix}\pi(u) & \pi(v) & \pi(y) & \pi(z)\end{pmatrix} \begin{array}{c c} & \begin{array}{cccc} u & v & y & z \end{array} \\ \begin{array}{c} u \\ v \\ y \\ z \end{array} & \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ \frac{1}{2} & 0 & 0 & \frac{1}{2} \\ 0 & 0 & 1 & 0 \end{pmatrix} \end{array} = \begin{pmatrix}\pi(u) & \pi(v) & \pi(y) & \pi(z)\end{pmatrix}$$

We can solve this linear system directly:

$$\frac{1}{2}\pi(y) = \pi(u);\ \pi(u) = \pi(v); \pi(v) + \pi(z) = \pi(y); \frac{1}{2}\pi(y) = \pi(z)$$

$$\pi(u) = \pi(u); \pi(v) = \pi(u); \pi(y) = 2\pi(u); \pi(z) = \pi(u)$$

Given $\pi(u) + \pi(v) + \pi(y) + \pi(z) = 1$, We can get: $\pi(u) = \frac{1}{5}; \pi(v) = \frac{1}{5}; \pi(y) = \frac{2}{5}; \pi(z) = \frac{1}{5}$

Thus the stationary distribution for the direct graph is $\pi = [\frac{1}{5}, \frac{1}{5}, \frac{2}{5}, \frac{1}{5}]$

The direct graph is non-reversible because the chain between state v,y,u is rotating clockwise while it will rotate counterclockwise if we reverse the chain. Taking the analogy of "playing the movie", if chain v, y, u is playing the movie forward, then when we reverse the chain, the movie can be discerned as playing backward.
Also The directed graph is non-reversible since it does not satisfy the local balance equation.

$$\pi(u)p(u, v) = \pi(v)p(v, u)$$

$$1/5 * 1 \neq 1/5 * 0$$

To get the stationary distribution for the undirected graph, we can solve $\pi = \pi P_U$, where $P_U = D_U^{-1} A_U$ using guassian elimination.

$$\begin{aligned}
0\pi_u + \frac{1}{2}\pi_v + \frac{1}{4}\pi_y + 0\pi_z &= \pi_u \\
\frac{1}{2}\pi_u + \ 0\pi_v + \frac{1}{4}\pi_y + 0\pi_z &= \pi_v \\
\frac{1}{2}\pi_u + \frac{1}{2}\pi_v + \ 0\pi_y + 1\pi_z &= \pi_y \\
0\pi_u + \ 0\pi_v + \frac{1}{2}\pi_y + 0\pi_z &= \pi_z \\
\pi_u + \quad \pi_v + \quad \pi_y + \quad \pi_z &= 1
\end{aligned}$$

$$\implies \left(\begin{array}{cccc|c} -1 & \frac{1}{2} & \frac{1}{4} & 0 & 0 \\ \frac{1}{2} & -1 & \frac{1}{4} & 0 & 0 \\ 0 & 0 & \frac{1}{2} & -1 & 0 \\ 1 & 1 & 1 & 1 & 1 \end{array}\right)$$

$$R_2 = 2R_2 + R_1 \left(\begin{array}{cccc|c} -1 & \frac{1}{2} & \frac{1}{4} & 0 & 0 \\ 0 & -\frac{3}{4} & \frac{3}{8} & 0 & 0 \\ 0 & 0 & \frac{1}{2} & -1 & 0 \\ 1 & 1 & 1 & 1 & 1 \end{array}\right)$$

$$R_4 = R_4 + R_1 \left(\begin{array}{cccc|c} -1 & \frac{1}{2} & \frac{1}{4} & 0 & 0 \\ 0 & -\frac{3}{4} & \frac{3}{8} & 0 & 0 \\ 0 & 0 & \frac{1}{2} & -1 & 0 \\ 0 & \frac{3}{2} & \frac{5}{4} & 1 & 1 \end{array}\right)$$

$$R_4 = R_4 + 2R_2 \left(\begin{array}{cccc|c} -1 & \frac{1}{2} & \frac{1}{4} & 0 & 0 \\ 0 & -\frac{3}{4} & \frac{3}{8} & 0 & 0 \\ 0 & 0 & \frac{1}{2} & -1 & 0 \\ 0 & 0 & 2 & 1 & 1 \end{array}\right)$$

$$R_4 = R_4 - 4R_3 \left(\begin{array}{cccc|c} -1 & \frac{1}{2} & \frac{1}{4} & 0 & 0 \\ 0 & -\frac{3}{4} & \frac{3}{8} & 0 & 0 \\ 0 & 0 & \frac{1}{2} & -1 & 0 \\ 0 & 0 & 0 & 5 & 1 \end{array}\right)$$

$$5\pi(z) = 1 \implies \pi(z) = \frac{1}{5}$$

$$\frac{1}{2}\pi(y) - \frac{1}{5} = 0 \implies \pi(y) = \frac{2}{5}$$

$$-\frac{3}{4}\pi(v) + \frac{3}{8}(\frac{2}{5}) = 0 \implies \frac{3}{4}\pi(v) = \frac{3}{20} \implies \pi(v) = \frac{1}{5}$$

$$\pi(u) + \pi(v) + \pi(y) + \pi(z) = 1 \implies \pi(u) = 1 - \frac{1}{5} - \frac{1}{5} - \frac{2}{5}, \pi(u) = \frac{1}{5}$$

SO, we get $\pi = [\frac{1}{5}, \frac{1}{5}, \frac{2}{5}, \frac{1}{5}]$. Or, $\pi(u) = \frac{1}{5}; \pi(v) = \frac{1}{5}; \pi(y) = \frac{2}{5}; \pi(z) = \frac{1}{5}$

The undirected graph is reversible as it satisfies the local balance equation, $\pi(x)p(x,y) = \pi(y)p(y,x)$. Using the $D_U^{-1}A_U$ matrix from Question 2, we can verify if the local balance holds.

$$\pi(u)p(u,v) = \pi(v)p(v,u) \implies \frac{1}{5} * \frac{1}{2} = \frac{1}{5} * \frac{1}{2} \checkmark$$

$$\pi(u)p(u,y) = \pi(y)p(y,u) \implies \frac{1}{5} * \frac{1}{2} = \frac{2}{5} * \frac{1}{4} \checkmark$$

$$\pi(u)p(u,z) = \pi(z)p(z,u) \implies \frac{1}{5} * 0 = \frac{1}{5} * 0 \checkmark$$

$$\pi(v)p(v,y) = \pi(y)p(y,v) \implies \frac{1}{5} * \frac{1}{2} = \frac{2}{5} * \frac{1}{4} \checkmark$$

$$\pi(v)p(v,z) = \pi(z)p(z,v) \implies \frac{1}{5} * 0 = \frac{1}{5} * 0 \checkmark$$

$$\pi(y)p(y,z) = \pi(z)pi(z,y) \implies \frac{2}{5} * \frac{1}{2} = \frac{1}{5} * 1 \checkmark$$

The undirected graph is reversible because we could film the Markov chain and we wouldn't be able to tell if it is playing forward or backward since its undirected and could be going either way.

Exercise 4

Suppose the two graph examples represent web pages / links of the world wide web. Compute the PageRank of each web page (vertex) by using the teleportation constant $q = 0.1$ in each case. Discuss whether the rankings are reasonable in the context of an internet search engine.

For the directed graph, the PageRank is calculated as follows:

First, we must look at our original directed transition probability matrix: $\begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ \frac{1}{2} & 0 & 0 & \frac{1}{2} \\ 0 & 0 & 1 & 0 \end{pmatrix}$

Then, using our teleportation constant q = 0.1, and the equation

$$p_q(x, y) = \frac{q}{n} + (1 - q)p(x, y)$$

our transition probability matrix becomes: $\begin{pmatrix} \frac{1}{40} & \frac{37}{40} & \frac{1}{40} & \frac{1}{40} \\ \frac{1}{40} & \frac{1}{40} & \frac{37}{40} & \frac{1}{40} \\ \frac{19}{40} & \frac{1}{40} & \frac{1}{40} & \frac{19}{40} \\ \frac{1}{40} & \frac{1}{40} & \frac{37}{40} & \frac{1}{40} \end{pmatrix}$

Now, we must solve for the PageRank of each vertex. The steps are as follows, we take our transition probability matrix A and transpose it, then subtract the eigenvalue one from the diagonal columns. From there, we RREF the resulting matrix to get the stationary measure, which we can then normalize to arrive at the stationary distribution.

$$A = \begin{pmatrix} \frac{1}{40} & \frac{37}{40} & \frac{1}{40} & \frac{1}{40} \\ \frac{1}{40} & \frac{1}{40} & \frac{37}{40} & \frac{1}{40} \\ \frac{19}{40} & \frac{1}{40} & \frac{1}{40} & \frac{19}{40} \\ \frac{1}{40} & \frac{1}{40} & \frac{37}{40} & \frac{1}{40} \end{pmatrix} \qquad A^T = \begin{pmatrix} \frac{1}{40} & \frac{1}{40} & \frac{19}{40} & \frac{1}{40} \\ \frac{37}{40} & \frac{1}{40} & \frac{1}{40} & \frac{1}{40} \\ \frac{1}{40} & \frac{37}{40} & \frac{1}{40} & \frac{37}{40} \\ \frac{1}{40} & \frac{1}{40} & \frac{19}{40} & \frac{1}{40} \end{pmatrix}$$

$$[A^T - I] = \begin{pmatrix} \frac{-39}{40} & \frac{1}{40} & \frac{19}{40} & \frac{1}{40} \\ \frac{37}{40} & \frac{-39}{40} & \frac{1}{40} & \frac{1}{40} \\ \frac{1}{40} & \frac{37}{40} & \frac{-39}{40} & \frac{37}{40} \\ \frac{1}{40} & \frac{1}{40} & \frac{19}{40} & \frac{-39}{40} \end{pmatrix}$$

We can now start using Reduced Row Echelon Form as follows:

$$R_1 = \frac{-40R_1}{39} \begin{pmatrix} 1 & \frac{1}{-39} & \frac{-19}{39} & \frac{-1}{39} \\ \frac{37}{40} & \frac{-39}{40} & \frac{1}{40} & \frac{1}{40} \\ \frac{1}{40} & \frac{37}{40} & \frac{-39}{40} & \frac{37}{40} \\ \frac{1}{40} & \frac{1}{40} & \frac{19}{40} & \frac{-39}{40} \end{pmatrix}$$

$$R_2 = R_2 - \frac{37R_1}{40} \begin{pmatrix} 1 & \frac{1}{-39} & \frac{-19}{39} & \frac{-1}{39} \\ 0 & \frac{-371}{390} & \frac{371}{780} & \frac{19}{390} \\ \frac{1}{40} & \frac{37}{40} & \frac{-39}{40} & \frac{37}{40} \\ \frac{1}{40} & \frac{1}{40} & \frac{19}{40} & \frac{-39}{40} \end{pmatrix}$$

$$R_3 = R_3 - \frac{R_1}{40} \begin{pmatrix} 1 & \frac{1}{-39} & \frac{-19}{39} & \frac{-1}{39} \\ 0 & \frac{-371}{390} & \frac{371}{780} & \frac{19}{390} \\ 0 & \frac{361}{390} & \frac{-751}{780} & \frac{361}{390} \\ \frac{1}{40} & \frac{1}{40} & \frac{19}{40} & \frac{-39}{40} \end{pmatrix}$$

$$R_4 = R_4 - \frac{R_1}{40} \begin{pmatrix} 1 & \frac{1}{-39} & \frac{-19}{39} & \frac{-1}{39} \\ 0 & \frac{-371}{390} & \frac{371}{780} & \frac{19}{390} \\ 0 & \frac{361}{390} & \frac{-751}{780} & \frac{361}{390} \\ 0 & \frac{1}{39} & \frac{19}{39} & \frac{-38}{39} \end{pmatrix}$$

$$R_2 = \frac{-390R_2}{371} \begin{pmatrix} 1 & \frac{1}{-39} & \frac{-19}{39} & \frac{-1}{39} \\ 0 & 1 & \frac{-1}{2} & \frac{-19}{371} \\ 0 & \frac{361}{390} & \frac{-751}{780} & \frac{361}{390} \\ 0 & \frac{1}{39} & \frac{19}{39} & \frac{-38}{39} \end{pmatrix}$$

$$R_3 = R_3 - \frac{361R_2}{390} \begin{pmatrix} 1 & \frac{1}{-39} & \frac{-19}{39} & \frac{-1}{39} \\ 0 & 1 & \frac{-1}{2} & \frac{-19}{371} \\ 0 & 0 & \frac{-1}{2} & \frac{361}{371} \\ 0 & \frac{1}{39} & \frac{19}{39} & \frac{-38}{39} \end{pmatrix}$$

$$R_4 = R_4 - \frac{R_2}{39} \begin{pmatrix} 1 & \frac{1}{-39} & \frac{-19}{39} & \frac{-1}{39} \\ 0 & 1 & \frac{-1}{2} & \frac{-19}{371} \\ 0 & 0 & \frac{-1}{2} & \frac{361}{371} \\ 0 & 0 & \frac{1}{2} & \frac{-361}{371} \end{pmatrix}$$

$$R_3 = -2R_3 \begin{pmatrix} 1 & \frac{1}{-39} & \frac{-19}{39} & \frac{-1}{39} \\ 0 & 1 & \frac{-1}{2} & \frac{-19}{371} \\ 0 & 0 & 1 & \frac{-722}{371} \\ 0 & 0 & \frac{1}{2} & \frac{-361}{371} \end{pmatrix}$$

$$R_1 = R_1 + \frac{R_3}{2} \begin{pmatrix} 1 & 0 & 0 & -1 \\ 0 & 1 & \frac{-1}{2} & \frac{-19}{371} \\ 0 & 0 & 1 & \frac{-722}{371} \\ 0 & 0 & \frac{1}{2} & \frac{-361}{371} \end{pmatrix}$$

$$R_2 = R_2 + \frac{R_3}{2} \begin{pmatrix} 1 & 0 & 0 & -1 \\ 0 & 1 & 0 & \frac{-380}{371} \\ 0 & 0 & 1 & \frac{-722}{371} \\ 0 & 0 & \frac{1}{2} & \frac{-361}{371} \end{pmatrix}$$

This last step will give the RREF form:

$$R_4 = R_4 - \frac{R_3}{2} \begin{pmatrix} 1 & 0 & 0 & -1 \\ 0 & 1 & 0 & \frac{-380}{371} \\ 0 & 0 & 1 & \frac{-722}{371} \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

$$\implies \pi_1 = \pi_4,\ \pi_2 = \frac{380}{371} \cdot \pi_4,\ \pi_3 = \frac{722}{371} \cdot \pi_4,\ \pi_4 = \pi_4$$

$$\mu^T = \begin{pmatrix} 1 \\ \frac{380}{371} \\ \frac{722}{371} \\ 1 \end{pmatrix} \text{ normalizes to } \implies \pi^T = \begin{pmatrix} \frac{371}{1844} \\ \frac{380}{1844} \\ \frac{722}{1844} \\ \frac{371}{1844} \end{pmatrix}$$

We can verify our answer with the following Python code:

```
np.set_printoptions(suppress=True)
#P = [[1,0,0,0],[0.07,0.63,0.3,0],[0.07,0.63,0,0.3],[0,0,0,1]]
P = [[1/40,37/40,1/40,1/40],[1/40,1/40,37/40,1/40],[19/40,1/40,1/40,19/40],[1/40,1/40,37/40,1/40]]

print("Original Transition Probability Matrix")
print()
print(np.dot(P,np.identity(4)))
print()
newP = np.identity(4)
for x in range(100000):
    newP = np.dot(newP,P)
print("Estimated Stationary Matrix after 100000 time steps")
print()
print(newP)
```

```
Original Transition Probability Matrix
[[0.025 0.925 0.025 0.025]
 [0.025 0.025 0.925 0.025]
 [0.475 0.025 0.025 0.475]
 [0.025 0.025 0.925 0.025]]

Estimated Stationary Matrix after 100000 time steps
[[0.20119306 0.20607375 0.39154013 0.20119306]
 [0.20119306 0.20607375 0.39154013 0.20119306]
 [0.20119306 0.20607375 0.39154013 0.20119306]
 [0.20119306 0.20607375 0.39154013 0.20119306]]
```

The ranking for the directed Markov chain is as follows: vertex y is the most important, followed by v, and then vertices u and z which share the same rank. This makes sense since vertex y clearly has the most influence, with the most amount of pages linked towards it with greater total probability than other vertices. y also leads to page z, which has a high probability to lead back to y. It also makes sense here that page v has the 2nd greatest rank, as being in vertex u at any point leads to vertex v with the highest probability, but vertex y moving to vertex u is comparatively less likely as it is equally likely to flow towards vertex z.

For the undirected graph, the PageRank is calculated as follows:

First, we must look at our original undirected transition probability matrix:

$$\begin{pmatrix} 0 & \frac{1}{2} & \frac{1}{2} & 0 \\ \frac{1}{2} & 0 & \frac{1}{2} & 0 \\ \frac{1}{4} & \frac{1}{4} & 0 & \frac{1}{2} \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

Then, using our teleportation constant q = 0.1, and the equation:

$$p_q(x,y) = \frac{q}{n} + (1-q)p(x,y)$$

our transition probability matrix becomes:

$$P = \begin{pmatrix} \frac{1}{40} & \frac{19}{40} & \frac{19}{40} & \frac{1}{40} \\ \frac{19}{40} & \frac{1}{40} & \frac{19}{40} & \frac{1}{40} \\ \frac{9.75}{40} & \frac{9.75}{40} & \frac{1}{40} & \frac{19.5}{40} \\ \frac{1}{40} & \frac{1}{40} & \frac{37}{40} & \frac{1}{40} \end{pmatrix}$$

Now, we must solve for the PageRank of each vertex.

$$P = \begin{pmatrix} \frac{1}{40} & \frac{19}{40} & \frac{19}{40} & \frac{1}{40} \\ \frac{19}{40} & \frac{1}{40} & \frac{19}{40} & \frac{1}{40} \\ \frac{9.75}{40} & \frac{9.75}{40} & \frac{1}{40} & \frac{19.5}{40} \\ \frac{1}{40} & \frac{1}{40} & \frac{37}{40} & \frac{1}{40} \end{pmatrix} \quad P^T = \begin{pmatrix} \frac{1}{40} & \frac{19}{40} & \frac{9.75}{40} & \frac{1}{40} \\ \frac{19}{40} & \frac{1}{40} & \frac{9.75}{40} & \frac{1}{40} \\ \frac{9.75}{40} & \frac{9.75}{40} & \frac{1}{40} & \frac{37}{40} \\ \frac{1}{40} & \frac{1}{40} & \frac{19.5}{40} & \frac{1}{40} \end{pmatrix}$$

$$[P^T - I] = \begin{pmatrix} \frac{-39}{40} & \frac{19}{40} & \frac{9.75}{40} & \frac{1}{40} \\ \frac{19}{40} & \frac{-39}{40} & \frac{9.75}{40} & \frac{1}{40} \\ \frac{39}{4} & \frac{39}{40} & \frac{-39}{40} & \frac{37}{40} \\ \frac{1}{40} & \frac{1}{40} & \frac{19.5}{40} & \frac{-39}{40} \end{pmatrix}$$

RREF process:

$$R_1 = \frac{40R_1}{39} \begin{pmatrix} 1 & \frac{-19}{40} & \frac{-1}{4} & \frac{-1}{39} \\ \frac{19}{40} & \frac{-39}{40} & \frac{9.75}{40} & \frac{1}{40} \\ \frac{39}{4} & \frac{39}{40} & \frac{-39}{40} & \frac{37}{40} \\ \frac{1}{40} & \frac{1}{40} & \frac{19.5}{40} & \frac{-39}{40} \end{pmatrix}$$

$$R_2 = R_2 - \frac{19R_1}{40} \begin{pmatrix} 1 & \frac{-19}{40} & \frac{-1}{4} & \frac{-1}{39} \\ 0 & \frac{-29}{39} & \frac{29}{80} & \frac{29}{780} \\ \frac{39}{4} & \frac{39}{40} & \frac{-39}{40} & \frac{37}{40} \\ \frac{1}{40} & \frac{1}{40} & \frac{19.5}{40} & \frac{-39}{40} \end{pmatrix}$$

$$R_3 = R_3 - \frac{19R_1}{40} \begin{pmatrix} 1 & \frac{-19}{40} & \frac{-1}{4} & \frac{-1}{39} \\ 0 & \frac{-29}{39} & \frac{29}{80} & \frac{29}{780} \\ 0 & \frac{551}{780} & \frac{-137}{160} & \frac{731}{780} \\ \frac{1}{40} & \frac{1}{40} & \frac{19.5}{40} & \frac{-39}{40} \end{pmatrix}$$

$$R_4 = R_4 - \frac{R_1}{40} \begin{pmatrix} 1 & \frac{-19}{40} & \frac{-1}{4} & \frac{-1}{39} \\ 0 & \frac{-29}{39} & \frac{29}{80} & \frac{29}{780} \\ 0 & \frac{551}{780} & \frac{-137}{160} & \frac{731}{780} \\ 0 & \frac{29}{780} & \frac{79}{160} & \frac{-38}{39} \end{pmatrix}$$

$$R_2 = \frac{-39R_2}{29} \begin{pmatrix} 1 & \frac{-19}{40} & \frac{-1}{4} & \frac{-1}{39} \\ 0 & 1 & \frac{-39}{80} & \frac{-1}{20} \\ 0 & \frac{551}{780} & \frac{-137}{160} & \frac{731}{780} \\ 0 & \frac{29}{780} & \frac{79}{160} & \frac{-38}{39} \end{pmatrix}$$

$$R_1 = R_1 + \frac{19R_2}{39} \begin{pmatrix} 1 & 0 & \frac{-39}{80} & \frac{-1}{20} \\ 0 & 1 & \frac{-39}{80} & \frac{-1}{20} \\ 0 & \frac{551}{780} & \frac{-137}{160} & \frac{731}{780} \\ 0 & \frac{29}{780} & \frac{79}{160} & \frac{-38}{39} \end{pmatrix}$$

$$R_3 = R_3 + \frac{551R_3}{780} \begin{pmatrix} 1 & 0 & \frac{-39}{80} & \frac{-1}{20} \\ 0 & 1 & \frac{-39}{80} & \frac{-1}{20} \\ 0 & 0 & \frac{-819}{1600} & \frac{389}{400} \\ 0 & \frac{29}{780} & \frac{79}{160} & \frac{-38}{39} \end{pmatrix}$$

$$R_4 = R_4 - \frac{29R_2}{780} \begin{pmatrix} 1 & 0 & \frac{-39}{80} & \frac{-1}{20} \\ 0 & 1 & \frac{-39}{80} & \frac{-1}{20} \\ 0 & 0 & \frac{-819}{1600} & \frac{389}{400} \\ 0 & 0 & \frac{819}{1600} & \frac{-389}{400} \end{pmatrix}$$

$$R_3 = \frac{-1600R_3}{819} \begin{pmatrix} 1 & 0 & \frac{-39}{80} & \frac{-1}{20} \\ 0 & 1 & \frac{-39}{80} & \frac{-1}{20} \\ 0 & 0 & 1 & \frac{-1556}{819} \\ 0 & 0 & \frac{819}{1600} & \frac{-389}{400} \end{pmatrix}$$

$$R_1 = R_1 + \frac{39R_3}{819} \begin{pmatrix} 1 & 0 & 0 & \frac{-41}{42} \\ 0 & 1 & \frac{-39}{80} & \frac{-1}{20} \\ 0 & 0 & 1 & \frac{-1556}{819} \\ 0 & 0 & \frac{819}{1600} & \frac{-389}{400} \end{pmatrix}$$

$$R_2 = R_2 + \frac{39R_3}{80} \begin{pmatrix} 1 & 0 & 0 & \frac{-41}{42} \\ 0 & 1 & 0 & \frac{-41}{42} \\ 0 & 0 & 1 & \frac{-1556}{819} \\ 0 & 0 & \frac{819}{1600} & \frac{-389}{400} \end{pmatrix}$$

$$\text{Reduced Row Echelon Form: } R_4 = R_4 - \tfrac{819R_3}{1600} \begin{pmatrix} 1 & 0 & 0 & \frac{-41}{42} \\ 0 & 1 & 0 & \frac{-41}{42} \\ 0 & 0 & 1 & \frac{-1556}{819} \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

$$\mu^T = \begin{pmatrix} \frac{41}{42} \\ \frac{41}{42} \\ \frac{1556}{819} \\ 1 \end{pmatrix} \text{ normalizes to } \implies \pi^T = \begin{pmatrix} \frac{41}{42} * \frac{819}{3974} \\ \frac{41}{42} * \frac{819}{3974} \\ \frac{1556}{3974} \\ \frac{819}{3974} \end{pmatrix}$$

We can verify our answer with the following Python code:

```
np.set_printoptions(suppress=True)
#P = [[1,0,0,0],[0.07,0.63,0.3,0],[0.07,0.63,0,0.3],[0,0,0,1]]
#P = [[1/40,37/40,1/40,1/40],[1/40,1/40,37/40,1/40],[19/40,1/40,1/40,19/40],[1/40,1/40,37/40,1/40]]
P = [[1/40,19/40,19/40,1/40],[19/40,1/40,19/40,1/40],[9.75/40,9.75/40,1/40,19.5/40],[1/40,1/40,37/40,1/40]]

print("Original Transition Probability Matrix")
print()
print(np.dot(P,np.identity(4)))
print()
newP = np.identity(4)
for x in range(100000):
    newP = np.dot(newP,P)
print("Estimated Stationary Matrix after 100000 time steps")
print()
print(newP)
```

```
Original Transition Probability Matrix

[[0.025   0.475   0.475   0.025  ]
 [0.475   0.025   0.475   0.025  ]
 [0.24375 0.24375 0.025   0.4875 ]
 [0.025   0.025   0.925   0.025  ]]

Estimated Stationary Matrix after 100000 time steps

[[0.20118269 0.20118269 0.39154504 0.20608958]
 [0.20118269 0.20118269 0.39154504 0.20608958]
 [0.20118269 0.20118269 0.39154504 0.20608958]
 [0.20118269 0.20118269 0.39154504 0.20608958]]
```

The ranking for the undirected Markov Chain is as follows: vertex y is the most important, followed by z, and then vertices u and v which share the same rank. Intuitively, this is also makes sense as vertrx y has the most pages linking towards it. In this matrix, vertex Y moving to vertex z has a higher probability than moving towards u or v, which implies that vertex z would be larger than u and v in the long run, especially as z has a higher probability to flow back to y relative to teleporting away.

Exercise 5

Consider a (possibly directed) graph with a $n \times n$ adjacency matrix A. Let A^k be the kth power of A. Use induction to prove that $(A^k)_{ij}$ is the number of ways to walk from vertex i to vertex j in k steps.

We need to prove $(A^k)_{ij}$ is the number of ways to walk from vertex i to vertex j in k steps for any A^k.

Base Case: $k = 1$

$$A_{ij}^{k=1} = A_{ij}^{1} = A_{ij}$$

By definition of the adjacency matrix, A_{ij} gives the number of vertices from state i to j. Thus it is trivially proven for the base case where k=1.

Induction Step: $k = k + 1$

$$A_{ij}^{k+1} = A_{ij}^{k} \cdot A_{ij}^{1} = A_{ij}^{k} \cdot A_{ij}$$

By matrix multiplication:

$$A_{ij}^{k} \cdot A_{ij} = \sum_{x=1}^{n} A_{ix}^{k} \cdot A_{xj}$$

where A_{ij}^{k} is proven to $=$ the number of paths from i to x in k steps from the base case, and $A_{xj} =$ number of paths from x to j. Therefore $\sum_{x=1}^{n} A_{ix}^{k} \cdot A_{xj} =$ all possible paths from i to j in $k+1$ steps, as we are summing over all possible middle states of hitting some state x after k time steps, multiplied by the matrix that represents going from the state x to the state j in one time step. The inductive step is proven, thus $\forall k \in \mathbb{N}, (A^k)_{ij}$ is the number of ways to walk from vertex i to vertex j in k steps. $\square$

Exercise 6

Let A be an adjacency matrix of a (possibly directed) graph $G(\mathbb{V}, \mathbb{E})$. Give an expression for the one-step transition probability P_{xy} of a random walk on G in terms of A_{xy} and $\mu(x) = \sum_{y \in \mathbb{V}} A_{xy}$ Do not assume the number of vertices in $\mathbb{V}$ is finite (so A is not a matrix in the traditional sense), but do assume $\mu(x) < \infty$. What is the state space for this Markov chain? Let Δ_k denote a diagonal matrix with the ith entry equal to the sum of the ith row of A^k. Explain why when $\Delta_k^{-1} A^k$ is well defined, the result is a $|\mathbb{V}| \times |\mathbb{V}|$ matrix of the k-step transition probabilities. This matrix $\Delta_k^{-1} A^k$ may fail to exist if Δ_k has a zero on the diagonal. What minor modification to A (and G) can we make to fix this and still model the random walk accurately? The matrix $\Delta_k^{-1} A^k$ is also not well defined if $\mathbb{V}$ is not finite. Do we still have a Markov chain that models a random walk on G? Explain.

The one-step transition probability P_{xy} of a random walk on G can be given by $P_{xy} = \frac{A_{xy}}{\sum_y A_{xy}}$

And since we are given $\mu(x) = \sum_y A_{xy}$ we get that

$$P_{xy} = \frac{A_{xy}}{\mu(x)}$$

The state space is given by: $\mathbb{S} = \mathbb{Z}$

Given Δ_k is a diagonal matrix with ith entry equal to the sum of number of ways to walk from that vertex to another in k steps and the inverse of diagonal matrix is given by replacing the main diagonal elements of the matrix with their reciprocals, $\Delta_k^{-1} A^k$ implies the k-step transition probabilities.

$\Delta_k^{-1} A^k$ does not exist if there is a zero entry in Δ_k. Intuitively, this would represent a state with no edges leading out or in, effectively staying absorbed in that state forever. This means we can modify the chain to make any state become self-absorbing and lead back to itself with probability 1 if the state has no outgoing edges. That means let $P(x, x) = 1$ for a state that has no edges out and we will never leave once enter the state x.

We still have a Markov chain that models a random walk on G when $\mathbb{V}$ is not finite. This is because we are told to assume that regardless of $\mathbb{V}$, $\mu(x) < \infty$. When $\mu(x) < \infty$, this implies that the summation $\sum_{y\in\mathbb{V}} A_{xy}$ is also finite. Because of this, we can infer that

$$P_{xy} = \frac{A_{xy}}{\mu(x)}$$

is similarly well defined as before and sums to 1. The state space for the Markov chain is $\mathbb{Z}$. Intuitively, a Markov chain that has an non finite $\mathbb{V}$ could be represented as the classic random walk on a number line. While the state space in that example is all integers and thus infinite, the Markov chain itself is well defined even though it cannot be represented in a normal matrix format.

Exercise 7

Show that a random walk on "essentially" any undirected graph $G(\mathbb{V}, \mathbb{E})$ is a reversible Markov chain. By the word "essentially" we mean $\mu(\mathbb{V}) = \sum_{x\in\mathbb{V}} \mu(x) < \infty$ (with μ as in Exercise 6). Can a random walk on a directed graph be reversible? Give an example if "yes" and explain if "no".

Given that $\mu(\mathbb{V}) = \sum_{x\in\mathbb{V}} \mu(x) < \infty$, this implies that a stationary measure exists, and that a stationary distribution would be defined as: $\pi(x) = \frac{\mu(x)}{\sum_{z\in\mathbb{S}} \mu(z)}$. If a Markov chain satisfies local balance, it is reversible, so we will verify that local balance holds.

$$p(x,y) = \frac{A_{xy}}{\sum_i A_{xi}} = \frac{A_{xy}}{\mu(x)}$$

since $\sum_{i=0}^{|\mathbb{V}|} A_{xi} = \mu(x)$.

$$p(y,x) = \frac{A_{yx}}{\sum_i A_{yi}} = \frac{A_{yx}}{\mu(y)}$$

since $\sum_{i=0}^{|\mathbb{V}|} A_{yi} = \mu(y)$.

$$\pi(x) = \frac{\mu(x)}{\sum_{z\in\mathbb{V}} \mu(z)} = \frac{\mu(x)}{\mu(\mathbb{V})}$$

$$\pi(y) = \frac{\mu(y)}{\sum_{z\in\mathbb{V}} \mu(z)} = \frac{\mu(y)}{\mu(\mathbb{V})}$$

Because we are dealing with an undirected graph, the adjacency matrix is symmetric, so $A_{xy} = A_{yx}\ \forall x, y$. Thus,

$$\pi(x)p(x,y) = \frac{\mu(x)}{\mu(\mathbb{V})} \cdot \frac{A_{xy}}{\mu(x)} = \frac{\mu(y)}{\mu(\mathbb{V})} \cdot \frac{A_{yx}}{\mu(y)} = \pi(y)p(y,x)$$

$$\implies \frac{A_{xy}}{\mu(\mathbb{V})} = \frac{A_{yx}}{\mu(\mathbb{V})}$$

$$\implies 1 = 1\ \checkmark$$

Since local balance holds, a random walk on "essentially" any undirected graph $G(\mathbb{V}, \mathbb{E})$ is a reversible Markov chain.

When we are dealing with a directed graph, the adjacency matrix can be asymmetric, so $\exists x, y$ s.t $A_{xy} \neq A_{yx}$. Thus,

$$\pi(x)p(x,y) = \frac{\mu(x)}{\mu(\mathbb{V})} \cdot \frac{A_{xy}}{\mu(x)} = \frac{\mu(y)}{\mu(\mathbb{V})} \cdot \frac{A_{yx}}{\mu(y)} = \pi(y)p(y,x)$$

$$\implies \frac{A_{xy}}{\mu(\mathbb{V})} = \frac{A_{yx}}{\mu(\mathbb{V})}$$

$$\implies A_{xy} = A_{yx} \ \forall x, y$$

Since $\exists x, y$ s.t $A_{xy} \neq A_{yx}$, local balance does not hold, so a random walk on a directed graph cannot be reversible. From the working definition provided by the professor in the discord, a graph is undirected if and only if for all $A_{xy} = A_{yx} \ \forall x, y$. Thus, we know that for the directed graph, local balance will be broken at least for one point, which makes the whole thing unreversible.

Exercise 8

Let $G(\mathbb{V}, \mathbb{E})$ be an undirected graph in which every pair of vertices has en edge between them (i.e. $(x, y) \in \mathbb{E} \ \forall x, y \in \mathbb{V}$. Here we allow what are called self-loop edges, i.e. edges of the form (x, x) which are represented as $A_{xx} = 1$ in the adjacency matrix of G. Let X be a random walk on G. Let $\mathscr{D}_k$ be the number of distinct vertices visited by time $k \in \mathbb{N}$. Note that $\mathscr{D}_0 = 1$. Let $n = |\mathbb{V}|$ and $T_m = \min\{k \geq 0 : \mathscr{D}_k = m\}$.

A)

Explain why this setup is identical to the coupon collector problem.

This is identical to the coupon collector problem from before. At each step, we have an equal probability of reaching any new state, inclusive of the state we are already in. From there, the variable $\mathscr{D}_k$ measures the number of unique states we have visited. The coupon collector problem had us reaching for colored marbles in a jar, and then we put them back every time we pick one out, such that there is always an equal, uniform probability of taking any colored marble out of the box, and we measured the number of unique colors (states) visited.

B)

Show that $\mathscr{D} = \{\mathscr{D}_k\}_{k \in \mathbb{N}}$ is a reversible Markov chain. Explain why this does not contradict the "movie played forward/backward" analogy.

$\mathscr{D}_k$ can be said to be reversible, because the stationary distribution of $\mathscr{D}_k$ is when we have already visited all the states, so the stationary of $\mathscr{D}_k$ will be equal to $|\mathbb{V}|$, which is N possible states. Essentially, the stationary of the $\mathscr{D}_k$ is equal to $\pi(n) = 1$, and $\pi(x) = 0$ for all other x's. Starting the chain in the stationary distribution, we wouldn't be able to see a difference playing forward or backward, because it will be equal to N. This is why it does not contradict the movie playing forward/backward analogy. While you would be able to tell the difference in the begining states, starting in the stationary we would not be able to tell a difference.

C)

Show that $E(T_{m+1} - T_m) = \frac{n}{n-m}$.

We are given: $\quad \mathrm{T}_m = \min\{k \geq 0 : \mathscr{D}_k = m\}$

Where $T_{m+1} - T_m$ represents the number of steps in between going from the state m to the state m + 1.

Thus, We can show that for any given value of k,

$$T_{m+1} - T_m = 1 \text{ w.p } \frac{n-m}{n}$$

$$T_{m+1} - T_m = 2 \text{ w.p } \frac{n-m}{n} * (\frac{m}{n})$$

$$T_{m+1} - T_m = 3 \text{ w.p } \frac{n-m}{n} * (\frac{m}{n})^2$$

A pattern emerges, for any $k \in \mathbb{N}$:

$$T_{m+1} - T_m = k \text{ w.p } \frac{n-m}{n} * (\frac{m}{n})^{k-1}$$

$$T_{m+1} - T_m \implies \text{Geometric}(p = \frac{n-m}{n}, q = \frac{m}{n})$$

where $q = 1 - p$. Thus,

$$E(T_{m+1} - T_m) = E[Geo(\frac{n-m}{n})]$$

In words, the RHS is the expected value of a geometrically distributed random variable with parameter $p = \frac{n-m}{n}$.

$$\implies E(T_{m+1} - T_m) = \frac{1}{p} = \frac{1}{\frac{n-m}{n}} = \frac{n}{n-m}$$

Thus Shown

D)

Show that $E(T_n) = \sum_{k=1}^{n-1} \frac{n}{k}$.

Given $\mathscr{D}_0 = 1$, we can write out the transition matrix as following:

$$\begin{array}{c c}
 & \begin{array}{ccccccc} 1 & 2 & 3 & 4 & 5 & ... & \infty \end{array} \\
\begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \\ ... \\ \infty \end{array} &
\begin{pmatrix}
\frac{1}{n} & \frac{n-1}{n} & 0 & 0 & 0 & 0 & 0 \\
0 & \frac{2}{n} & \frac{n-2}{n} & 0 & 0 & 0 & 0 \\
0 & 0 & \frac{3}{n} & \frac{n-3}{n} & 0 & 0 & 0 \\
0 & 0 & 0 & \frac{4}{n} & \frac{n-4}{n} & 0 & 0 \\
0 & 0 & 0 & 0 & ... & ... & 0 \\
0 & 0 & 0 & 0 & 0 & \frac{x}{n} & \frac{n-x}{n}
\end{pmatrix}
\end{array}$$

So,

$$E[T_{1,N}] = 1 + E[T_{1,N}] \cdot P_{(1,1)} + E[T_{2,N}] \cdot P_{(1,2)}$$

$$E[T_{1,N}] - E[T_{1,N}] \cdot P_{(1,1)}) = 1 + E[T_{2,N}] \cdot P_{(1,2)}$$

$$E[T_{1,N}](1 - P_{(1,1)} = 1 + E[T_{2,N}] \cdot P_{(1,2)}$$

$$E[T_{1,N}] = \frac{1 + E[T_{2,N}] \cdot P_{(1,2)}}{1 - P_{(1,1)}}$$

Since

$$1 - P_{(1,1)} = P_{(1,2)}$$

$$\implies E[T_{1,N}] = \frac{1 + E[T_{2,N}] \cdot P_{(1,2)}}{P_{(1,2)}}$$

$$E[T_{1,N}] = \frac{1}{P_{(1,2)}} + E[T_{2,N}]$$

We can now have a expanded form:

$$\frac{1}{P_{(1,2)}} + \frac{1}{P_{(2,3)}} + \frac{1}{P_{(3,4)}} + [...] + \frac{1}{P_{(n-1,n)}} + E[T_{n,n}]$$

where $E[T_{n,n}]$ is 0, as we already are at state n.
Thus,

$$\frac{n}{n-1} + \frac{n}{n-2} + \frac{n}{n-3} + [...] + \frac{n}{n-(n-1)}$$

$$n(\frac{1}{n-1} + \frac{1}{n-2} + \frac{1}{n-3} + [...] + 1)$$

$$= n \cdot \sum_{k=1}^{n-1} \cdot \frac{1}{k}$$

$$= \sum_{k=1}^{n-1} \frac{n}{k}$$

Thus shown.

Extra Credit

Implement a random walk as a computer program in R or Python for the two example graphs on the first page of the assignment. In this exercise we will compare their expected cover times. A cover time of a graph G is a random variable C which equals the total number of steps of a random walk on G that it took to visit all the vertices. In Exercise 8 we computed the expected time cover time analytically, but this is rarely possible. Here, let's use Monte Carlo simulation to estimate the expected cover time (i.e. $E_\pi(C)$). Which of the two graphs has a smaller expected cover time?

A)

Implement a random walker for both graphs in this problem that start at any given vertex x. We will start each walk from the stationary distribution of the graph. Thus you need to sample x from π. See Ross, Chapter 11 on how to draw samples from a discrete distribution.

Please note, in this simulation we have 1 = U, 2 = V, 3 = Y, 4 = Z. It functions identically but saves me time to do it this way.

```python
import numpy as np

#Stationary Distribution Generator for the initial start point
#Please Note, 1 = U, 2 = V, 3 = Y, 4 = Z. It functions Identitically but saves me time to do it like this.
def stationary():
    temp = np.random.random_sample(1)
    if temp < 0.2:
        return 1
    elif temp < 0.4:
        return 2
    elif temp < 0.8:
        return 3
    else:
        return 4
```

```python
#Directed Graph
def graph1():
    visted = []
    totalTime = 0
    current = stationary()
    visted.append(current)
    while len(visted) < 4:
        if current in visted:
            totalTime +=1
            current = probbers1(current)
        else:
            totalTime +=1
            visted.append(current)
            current = probbers1(current)
    return totalTime - 1

#Directed Graph Transition Probabilities
def probbers1(current):
    if current == 1:
        current = 2
    elif current == 2:
        current = 3
    elif current == 3:
        rand = np.random.random_sample(1)
        if rand < 0.5:
            current = 1
        else:
            current = 4
    elif current == 4:
         current = 3
    return current

#Monte Carlo Simulator for Direct Graph
def monteCarloSampler1(n):
    currentTime = 1
    estimate = []
    total = 0
    while currentTime <= n:
        tempge = graph1()
        total += tempge
        estimate.append(total/currentTime)
        currentTime += 1
    return estimate
```

```
#Undirected Graph
def graph2():
    visted = []
    totalTime = 0
    current = stationary()
    visted.append(current)
    while len(visted) < 4:
        if current in visted:
            totalTime +=1
            current = probbers2(current)
        else:
            totalTime +=1
            visted.append(current)
            current = probbers2(current)
    return totalTime - 1


#Undirected Graph Transition Probabilities
def probbers2(current):
    if current == 1:
        rand = np.random.random_sample(1)
        if rand < 0.5:
            current = 3
        else:
            current = 2
    elif current == 2:
        rand = np.random.random_sample(1)
        if rand < 0.5:
            current = 1
        else:
            current = 3
    elif current == 3:
        rand = np.random.random_sample(1)
        if rand < 0.25:
            current = 1
        elif rand < 0.5:
            current = 2
        else:
            current = 4
    elif current == 4:
         current = 3
    return current
```

B)

Add a function to generate samples $\{C^{(j)}\}_{j=1}^{m}$ of the cover time. Use the sample mean $\sum_{j=1}^{m} \frac{C^{(j)}}{m}$ with large m to estimate $E_\pi(C)$.

The two functions that estimate the expected cover time are monteCarloSampler1 and monteCarloSampler2. After m = 10,000 we calculated an expected cover time of 6.17 for the directed graph and 7.99 for the undirected graph and they are shown below, alongside a graph of the approximation. From this, we can see that the Directed graph has a shorter expected cover time, this makes sense because there is less variability, once we enter the state u, we know we will get to state v with probability 1, while the undirected version could flip backwards to state y.

```python
#Monte Carlo Simulator for Undirected Graph
def monteCarloSampler2(n):
    currentTime = 1
    estimate = []
    total = 0
    while currentTime <= n:
        tempge = graph2()
        total += tempge
        estimate.append(total/currentTime)
        currentTime += 1
    return estimate
```

```python
y = monteCarloSampler1(10000)
y2 = monteCarloSampler2(10000)
print(y[10000-1])  #Monte Carlo Estimation after 10000 trials for the Directed Graph
print(y2[10000-1]) #Monte Carlo Estimation after 10000 trials for the Undirected Graph
```

```
6.1669
7.9816
```

```python
plt.figure(figsize=(10,5))
plt.title("Monte Carlo Estimation for the Expected Cover Time")
plt.plot(np.arange(10000),y,'-',color="red",label="Directed Graph")
plt.plot(np.arange(10000),y2,'-',color="blue",label="Undirected Graph")

plt.xlabel("Number of simulations")
plt.ylabel("Estimation")
plt.legend(loc="lower right")
```

```
<matplotlib.legend.Legend at 0x202fbbb3190>
```