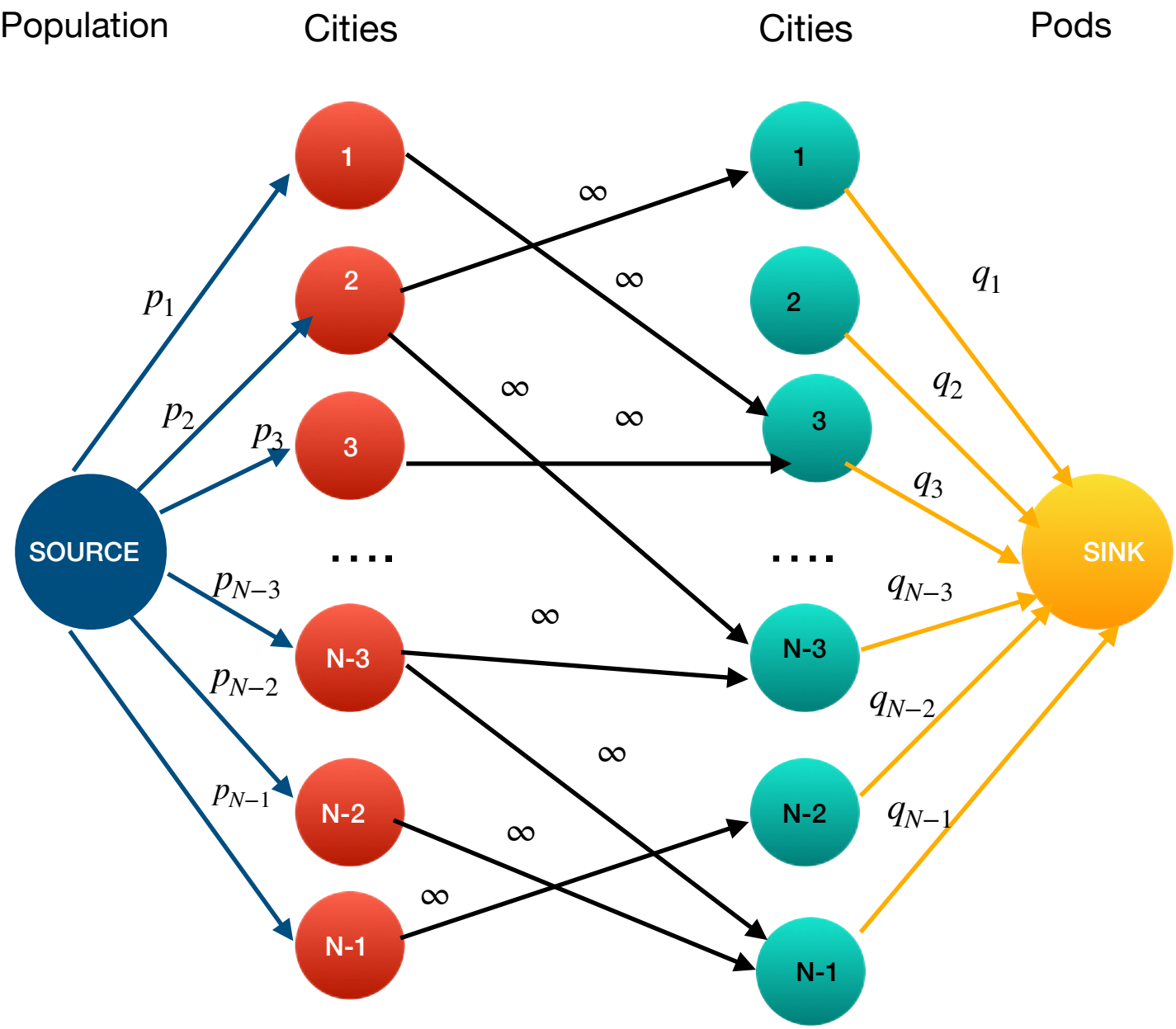
Mubarra Khan Z5161249 Tuesday, 4 August 2020

Q1

---

1. Figure out a flow graph which shows the cities inhabitants can travel to, and acquire pods.

- Do this by creating a bi-partite graph which connects cities to one another with vertices corresponding to ALL cities both on the left and on the right hand side.
    - They will however have different interpretations as the left vertices will represent the population *p* corresponding to each city and the right vertices will represent the set of pods in the corresponding cities
- Create an edge between two cities *i, j* if the minimum path between the cities' is less than X . This should include an edge between a city to itself as the population of one city can access the pods in their own city.
    - To find a minimum path you will need to do a breadth first search and then add an edge if such a path is less that X.
    - The maximum number of edges you can have between the cities on the left and right is N^2
- Next create edges between the super source and the cities on the left.
    - Each edge representing the population of the city it is connected to.
        - This prevents a city from having more than its population flow out of it.
- The edges between the cities from left to right will be infinity which will allow the possibility of any given city on the left to saturate the pods of the city on the right.
    - This is because there is no limit to the number of people that can travel between a city.
- Now create edges between the cities on the right to the super sink .
    - Each edge will represent the number of pods of the city on the right.
        - This will prevent the city on the right from letting more pods leave it than it has.
- With this bipartite graph we can run the Ford-Fulkerson maximum flow algorithm and look at occupied edges for the number of pods to determine the maximum number of invaders Earth will have to deal with.
- A visual of the bipartite graph is shown below.


Population
Cities
Cities
Pods
SOURCE
1
2
3
N-3
N-2
N-1
1
2
3
N-3
N-2
N-1
SINK
p_1
p_2
p_3
p_{N-3}
p_{N-2}
p_{N-1}
q_1
q_2
q_3
q_{N-3}
q_{N-2}
q_{N-1}
∞