Question 1

Estimating all-pairs distances exactly

Algorithm 1: Monte Carlo Algorithm for exact distance

```
Input: G, partial distance matrix M
Output: complete distance matrix M with error probability < 1 − 1/n^2
D ← n x n matrix to store distance
for i ← 1 to n do
    for j ← 1 to n do
        if M[i][j] = # then
            D[i][j] = infinity
        else
            D[i][j] = M[i][j]
    end
end
for k ← 1 to K do
    w ← Random vertex in [1 : n]
    dist = bfs(G, w)
    for i ← 1 to n do
        for j ← 1 to n do
            D[i][j] = min(D[i][j], dist[i] + dist[j])
        end
    end
end
return M
```

Complexity analysis

In above algorithm $bfs(G, s)$ returns an array storing shortest distance of s to each vertex in G. Time complexity for bfs is $O(n + m)$ (m = number of vertices in graph). In worst case there can be atmost $\binom{n}{2}$ edges, i.e worst case complexity of bfs is $O(n^2)$. Time taken to update D in one iteration is also $O(n^2)$. Since outer loop runs for K times, total complexity is $O(n^2K)$. In below analysis it is show that $K = O(\log n)$, hence complexity of above algorithm is $\boldsymbol{O(n^2)\log n}$.

Error Analysis

Error will occur if there are ≥ 1 entries in D which are not equal shortest path distance.

$$P(error) = P(M[1][1] \text{ wrong } \cup M[1][2] \text{ wrong } \ldots \cup M[n][n] \text{ wrong })$$

Applying Boole's inequality we get:

$$P(error) \leq n^2 * P(M[i][j] \text{ wrong }) \text{ (distance between } i, j > n/100 \text{)}$$

Note that there is $\leq$ sign in above line since the points for which distance $\leq n/100$ matrix M stores correct value (since the value is taken directly from partial distance matrix).

$M[i][j]$ computed wrong

We know that distance between i, j is greater than $n/100$, i.e number of nodes between shortest path from i to j is $\geq n/100$. So now consider first iteration of first loop.
Note that w is choosen randomly, hence the probability that w belongs to shortest path from i to j is $\geq \dfrac{n/100}{n} = \dfrac{1}{100}$.
If w belong to shortest path then $D[i][j]$ will be updated correctly and will contain shortest path distance (since we are doing bfs from w, we get shortest path value from $w \to i, w \to j$, $M[i][j]$ is simple the sum

of these 2 values).
Probability that $D[i][j]$ is wrong $(P(\delta)) \leq \frac{99}{100}$
Probability that $D[i][j]$ is wrong even after K iteration $\leq P(\delta)^K = \left(\frac{99}{100}\right)^K$
Using this value in above equation we get

$$\begin{aligned} P(error) &\leq n^2 P(\delta) \\ &\leq n^2 \left(\frac{99}{100}\right)^K \end{aligned}$$

We want that error probability is $\leq 1/n^2$, or

$$\begin{aligned} P(error) &\leq \frac{1}{n^2} \\ n^2 \left(\frac{99}{100}\right)^K &\leq \frac{1}{n^2} \\ \left(\frac{99}{100}\right)^K &\leq \frac{1}{n^4} \\ K \log\left(\frac{100}{99}\right) &\geq 4 \log n \\ K &\geq \boldsymbol{\frac{4}{\log(100/99)} \log n} \end{aligned}$$

Taking $\boldsymbol{K = 400 \log n}$ we get that all entries of the distance matrix are correct with probability exceeding $1 - 1/n^2$.

Question 2

Rumour Spreading

Question 3

Approximate Ham-Sandwich Cut