Assignment 2

# Q1

1. Top 5 nodes with the highest PageRank Scores

Top:

id: 263, score: 0.0020202911815182184

id: 537, score: 0.00194334157145315

id: 965, score: 0.001925447807166263

id: 243, score: 0.001852634016241731

id: 285, score: 0.0018273721700645144

Bottom 5 nodes with the lowest PageRank Scores

Bottom:

id: 558, score: 0.0003286018525215297

id: 93, score: 0.0003513568937516577

id: 62, score: 0.00035314810510596274

id: 424, score: 0.00035481538649301454

id: 408, score: 0.00038779848719291705

1. 5 node ids with the highest hubbiness score

Highest Hubbiness:

id: 840, score: 1.0

id: 155, score: 0.9499618624906543

id: 234, score: 0.8986645288972263

id: 389, score: 0.8634171101843789

id: 472, score: 0.8632841092495218

5 node ids with the lowest hubbiness score

Lowest Hubbiness:

id: 23, score: 0.04206685489093652

id: 835, score: 0.057790593544330145

id: 141, score: 0.06453117646225177

id: 539, score: 0.0660265937341849

id: 889, score: 0.07678413939216452

5 node ids with the highest authority score

Highest Authority:

id: 893, score: 1.0

id: 16, score: 0.9635572849634398

id: 799, score: 0.9510158161074015

id: 146, score: 0.9246703586198443

id: 473, score: 0.899866197360405

5 node ids with the lowest authority score

Lowest Authority:

id: 19, score: 0.05608316377607618

id: 135, score: 0.06653910487622794

id: 462, score: 0.07544228624641901

id: 24, score: 0.08171239406816942

id: 910, score: 0.08571673456144875

# Q2

For each string in the set S, we apply two hash functions to calculate two positions in the Bloom filter and set those positions to 1. The first hash function, h1, calculates a position based on the sum of the alphabetical positions of each character in the string modulo 7. The second hash function, h2, calculates a position based on the length of the string modulo 7.

Update the Bloom filter for the set of strings S = “hi”, “big”, “data”, “spark”:

First, initialize the Bloom filter with 7 bits with all zeros.

**Initial Bloom filter: 0000000**

Now, apply the hash functions to each string and set the corresponding bits to 1:

For “hi”:

h1(“hi”) = (7 + 8) mod 7 = 1

h2(“hi”) = (2 \* 3) mod 7 = 6

Set the bits at positions 1 and 6 to 1.

Bloom filter: 0100001

For “big”:

h1(“big”) = (1 + 8 + 6) mod 7 = 1

h2(“big”) = (3 \* 3) mod 7 = 2

Set the bits at positions 1 and 2 to 1.

Bloom filter: 0110001

For “data”:

h1(“data”) = (3 + 0 + 19 + 0) mod 7 = 1

h2(“data”) = (4 \* 3) mod 7 = 5

Set the bits at positions 1 and 5 to 1.

Bloom filter: 0110011

For “spark”:

h1(“spark”) = (18 + 15 + 0 + 17 + 10) mod 7 = 1

h2(“spark”) = (5 \* 3) mod 7 = 1

Set the bit at position 1 to 1.

**Final Bloom filter: 0110011**