Que1:

(a) Represent the network in terms of its 'adjacency matrix' as well as 'edge list'.

In [55]:
```
# libereraies added
import numpy as np
import operator
from collections import Counter
import matplotlib.pyplot as plt
```

In [56]:
```
import pandas as pd
data = pd.read_csv('fb-pages-food.edges', sep=',') #data loadede here int the
 form of data frame: undirected dataset
```

In [57]:
```
data.columns=["u","v"]  #two columns name u and v both were nodes
data
```

Out[57]:

	u	v
0	0	58
1	0	132
2	0	603
3	0	398
4	0	555
...	...	...
2096	597	611
2097	601	603
2098	601	616
2099	603	616
2100	311	613

2101 rows × 2 columns

total number of nodes: 620

count total number of nodes in graph

In [58]:
```
n=set(data['u']).union(set(data['v']))
print("total number of nodes: ",len(n))
```

```
total number of nodes:  620
```

Adjancecy matix of size n by n with 0 initlaization.

In [59]:
```
# https://stackoverflow.com/questions/2397141/how-to-initialize-a-two-dimensional-array-in-python
adjancency_matrix= [[0 for i in range(len(n))] for j in range(len(n))]    #adjancecey matrix
```

fill up the matrix if there is an edge between node u and node v then put 1 in matrix block between matrix[u][v] else assign 0 in them

In [60]:
```
for i in range(0,len(data)):   #loop till o to total edges in the graph and fill the matrix
  if(data['u'][i]!=data['v'][i]):   # check for 2 nodes should't be same else self loop will occure no require for that
     adjancency_matrix[data['u'][i]][data['v'][i]]=1     #assign 1 is there us edge
     adjancency_matrix[data['v'][i]][data['u'][i]]=1     # else 0
```

Adjancecy matrix of graph

In [61]:
```
import pprint
pp = pprint.PrettyPrinter(indent=10)
pp.pprint(np.array(adjancency_matrix))
```

```
array([[0, 0, 0, ..., 0, 0, 0],
       [0, 0, 0, ..., 0, 0, 0],
       [0, 0, 0, ..., 0, 0, 0],
       ...,
       [0, 0, 0, ..., 0, 0, 0],
       [0, 0, 0, ..., 0, 0, 0],
       [0, 0, 0, ..., 0, 0, 0]])
```

Create an edge list for all the nodes of the graph:

In [62]:
```
edge_list={}   # dictionary for edge list
for i in range(len(adjancency_matrix)): # initialise empty list for all the nodes
  edge_list[i]=[]
```

In [63]:
```
total_edges=0    #total edges are 0 intitia;;y
degree_of_nodes={}   # dictionary for degree of each node where key is node an
d value is degree of particular node.
for i in range(len(adjancency_matrix)):  # loop for rows
  for j in range(len(adjancency_matrix[i])): #loop for columns
    if adjancency_matrix[i][j]==1:      #check is there is an edge in graph
      edge_list[i].append(j)                # then append it on the list of pa
rticular node
  degree_of_nodes[i]=len(edge_list[i])       #here total number of nodes prese
nt in the list of nodes of edge list i.e.total neighbors is the degree of a pa
rticular node.
  total_edges+=len(edge_list[i])             #here counting of total edges
  print(i,"------",edge_list[i])
```

```
0 ------ [58, 132, 398, 555, 603]
1 ------ [265, 611]
2 ------ [182, 265, 345]
3 ------ [40, 65, 185, 228, 299, 352, 377, 450, 484, 608]
4 ------ [254, 265, 287, 336]
5 ------ [503]
6 ------ [163, 429, 478, 486, 518]
7 ------ [35, 265, 293, 305, 339, 444, 583]
8 ------ [73, 98, 278, 369, 434]
9 ------ [15, 23, 56, 67, 89, 90, 107, 131, 189, 206, 229, 248, 249, 254, 26
5, 288, 289, 317, 340, 351, 434, 446, 465, 498, 550, 570, 593, 611]
10 ------ [258, 393]
11 ------ [174, 212, 259, 264, 380, 485]
12 ------ [310, 548, 618]
13 ------ [352]
14 ------ [264, 326]
15 ------ [9, 43, 56, 67, 70, 265, 340, 432, 454, 505]
16 ------ [75, 117, 119, 237, 274, 334, 374, 400, 466, 516, 545, 613]
17 ------ [65, 185, 299, 450, 608]
18 ------ [265]
19 ------ [572]
20 ------ [160, 238, 242, 324, 355, 481]
21 ------ [329, 334, 441, 516]
22 ------ [104, 187, 270, 420]
23 ------ [9, 54, 58, 67, 128, 157, 182, 227, 257, 277, 288, 340, 343, 434]
24 ------ [230, 306, 501, 502, 547, 591]
25 ------ [363]
26 ------ [230]
27 ------ [424]
28 ------ [147, 230, 527]
29 ------ [49]
30 ------ [126]
31 ------ [41, 48, 63, 142, 169, 179, 193, 266, 269, 285, 315, 327, 357, 424,
449, 488, 491, 507, 508, 515, 518, 524, 543, 546, 595]
32 ------ [89, 159, 183, 265, 300, 364, 581, 611]
33 ------ [209, 490]
34 ------ [89, 208, 254, 265, 373, 419]
35 ------ [7, 56, 58, 87, 89, 90, 131, 195, 198, 243, 265, 289, 338, 383, 38
9, 439, 536, 597, 601, 611, 615, 618]
36 ------ [490]
37 ------ [389]
38 ------ [518]
39 ------ [43, 250, 340, 509, 576, 603]
40 ------ [3, 240, 333, 344, 352, 484, 538]
41 ------ [31, 62, 109, 121, 193, 265, 282, 321, 397, 434, 465, 594, 611]
42 ------ [467]
43 ------ [15, 39, 67, 83, 102, 229, 253, 265, 340, 360, 505, 577]
44 ------ [265]
45 ------ [63, 191, 357, 461, 518, 524]
46 ------ [67, 86, 110, 317, 417, 550, 578]
47 ------ [475, 500, 508]
48 ------ [31, 164, 357, 518, 524]
49 ------ [29, 71, 188, 196, 293, 512]
50 ------ [67, 70, 131, 171, 182, 183, 289, 330, 340, 389, 483, 544, 562, 57
8]
51 ------ [141]
52 ------ [113, 494]
```

```
53 ------ [122]
54 ------ [23, 89, 265, 275, 536]
55 ------ [78, 154, 158, 210, 214, 252, 437, 452]
56 ------ [9, 15, 35, 60, 67, 68, 70, 87, 89, 90, 102, 116, 128, 130, 131, 15
1, 182, 195, 198, 217, 229, 235, 238, 244, 265, 289, 338, 340, 345, 350, 351,
359, 375, 383, 389, 439, 465, 479, 505, 577, 584, 597, 601, 603, 611, 616, 61
7]
57 ------ [148, 264]
58 ------ [0, 23, 35, 87, 89, 90, 265, 302, 350, 434, 537, 601, 603, 611, 61
6]
59 ------ [116, 265]
60 ------ [56, 198, 265, 350, 389, 439]
61 ------ [244]
62 ------ [41, 169, 193, 265, 266, 269, 282, 285, 392, 449, 611]
63 ------ [31, 45, 139, 164, 357, 448, 504, 518, 572, 602]
64 ------ [498]
65 ------ [3, 17, 185, 240, 299, 333, 450, 608]
66 ------ [141]
67 ------ [9, 15, 23, 43, 46, 50, 56, 68, 70, 89, 90, 102, 107, 131, 146, 18
1, 183, 189, 198, 229, 245, 248, 249, 253, 254, 265, 288, 289, 291, 317, 323,
325, 338, 340, 351, 352, 372, 389, 410, 416, 419, 434, 440, 445, 465, 469, 49
8, 505, 531, 550, 570, 577, 578, 593, 603, 611]
68 ------ [56, 67, 70, 198, 242, 265, 288, 340, 505, 603]
69 ------ [253, 385, 498]
70 ------ [15, 50, 56, 67, 68, 72, 89, 90, 102, 107, 116, 170, 181, 189, 205,
229, 235, 244, 248, 249, 254, 265, 269, 275, 288, 317, 323, 340, 342, 383, 40
3, 434, 439, 465, 473, 498, 505, 526, 536, 550, 570, 577, 593, 599, 601, 611,
616]
71 ------ [49, 265, 495]
72 ------ [70, 240, 269]
73 ------ [8, 369]
74 ------ [98]
75 ------ [16, 117, 119, 237, 274, 334, 400, 516, 613]
76 ------ [483]
77 ------ [379]
78 ------ [55, 154, 158, 210, 214, 252, 331, 437, 452]
79 ------ [85, 357, 414, 491, 507, 518, 546, 595]
80 ------ [138]
81 ------ [290, 563]
82 ------ [171, 261, 389, 409, 451, 584]
83 ------ [43, 89, 254, 265, 288, 317, 465]
84 ------ [329, 374, 545]
85 ------ [79, 292, 335, 507, 518]
86 ------ [46, 578]
87 ------ [35, 56, 58, 89, 90, 131, 151, 175, 182, 198, 217, 227, 229, 257, 2
65, 289, 338, 340, 350, 375, 389, 439, 479, 526, 576, 580, 597, 601, 611, 61
6]
88 ------ [265, 434, 460, 474, 498, 550]
89 ------ [9, 32, 34, 35, 54, 56, 58, 67, 70, 83, 87, 90, 116, 128, 136, 151,
155, 182, 217, 221, 245, 248, 254, 275, 288, 300, 317, 324, 340, 343, 350, 35
1, 419, 432, 446, 465, 505, 558, 569, 570, 584, 597, 601]
90 ------ [9, 35, 56, 58, 67, 70, 87, 89, 116, 130, 131, 151, 157, 182, 195,
198, 206, 208, 217, 227, 229, 238, 242, 265, 275, 289, 317, 324, 340, 342, 35
0, 351, 355, 372, 373, 389, 413, 434, 439, 473, 479, 505, 550, 557, 580, 584,
597, 601, 603, 611, 616]
91 ------ [253, 563]
92 ------ [186]
```

```
93 ------ [94, 103, 146, 567, 614]
94 ------ [93, 103]
95 ------ [492, 523]
96 ------ [211, 379, 537]
97 ------ [265]
98 ------ [8, 74, 99, 216, 295, 471, 605]
99 ------ [98, 108, 142, 431]
100 ------ [518]
101 ------ [124, 152, 216, 311, 329, 374, 545]
102 ------ [43, 56, 67, 70, 198, 217, 227, 554]
103 ------ [93, 94, 567]
104 ------ [22, 187, 253, 290, 469, 563]
105 ------ [521, 548]
106 ------ [305, 444, 583]
107 ------ [9, 67, 70, 265, 291, 317, 340, 383, 416, 498, 505, 550]
108 ------ [99, 147, 230, 527, 547]
109 ------ [41, 169, 181, 320]
110 ------ [46, 578]
111 ------ [229, 265, 389, 451]
112 ------ [265, 498]
113 ------ [52, 222, 276, 319, 358, 398, 494, 522, 555, 559]
114 ------ [134, 138, 143, 184, 200, 201, 389, 397]
115 ------ [145, 485]
116 ------ [56, 59, 70, 89, 90, 151, 182, 198, 243, 248, 265, 275, 342, 366,
372, 434, 446, 536, 558, 593, 611, 616]
117 ------ [16, 75, 119, 137, 237, 274, 334, 374, 400, 466, 516, 545, 613]
118 ------ [357, 424, 507, 518]
119 ------ [16, 75, 117, 237, 274, 334, 374, 400, 466, 516, 545, 613]
120 ------ [265, 375]
121 ------ [41, 227, 265, 433, 553]
122 ------ [53, 131, 289]
123 ------ [179, 292, 347, 491, 518, 524]
124 ------ [101, 374, 545, 613]
125 ------ [308, 423, 565]
126 ------ [30, 147, 501, 502]
127 ------ [347, 357, 518, 524]
128 ------ [23, 56, 89, 131, 135, 208, 254, 265, 289, 373, 404, 419, 446, 46
5, 570, 576]
129 ------ [154, 442]
130 ------ [56, 90, 136, 151, 217, 288, 340, 355, 432, 597]
131 ------ [9, 35, 50, 56, 67, 87, 90, 122, 128, 151, 157, 182, 217, 229, 23
8, 248, 265, 288, 325, 340, 343, 351, 505, 510, 597, 601]
132 ------ [0, 198]
133 ------ [397, 407]
134 ------ [114, 143, 184, 232, 251, 456, 563]
135 ------ [128, 227, 281, 317, 321, 350, 359, 360, 386, 446, 458, 544]
136 ------ [89, 130, 155, 208, 229, 265, 389, 422, 439, 536, 558, 568, 576, 6
11]
137 ------ [117, 152, 311, 329, 441]
138 ------ [80, 114, 418]
139 ------ [63, 518]
140 ------ [265, 581]
141 ------ [51, 66, 280, 461, 574]
142 ------ [31, 99, 164, 491, 518, 524, 596]
143 ------ [114, 134, 151, 172, 182, 184, 265, 397, 407, 408, 456, 580, 581]
144 ------ [150, 311, 374, 441, 545]
145 ------ [115, 241, 264, 380, 485, 525]
```

```
146 ------ [67, 93, 193, 232, 253, 354, 498, 505, 567]
147 ------ [28, 108, 126, 230, 279, 438, 501, 502, 527, 547, 591]
148 ------ [57, 253, 265, 498, 563]
149 ------ [276, 319, 494, 522, 555]
150 ------ [144, 311, 329, 374, 545, 613]
151 ------ [56, 87, 89, 90, 116, 130, 131, 143, 195, 229, 265, 289, 345, 375,
389, 557, 580, 597, 601, 603, 611]
152 ------ [101, 137, 311, 329, 374, 382, 441, 545]
153 ------ [412]
154 ------ [55, 78, 129, 158, 210, 214, 223, 252, 331, 437, 452, 533]
155 ------ [89, 136, 254, 265, 293, 419, 570, 576]
156 ------ [483]
157 ------ [23, 90, 131, 257, 288, 289, 510, 585, 603, 604]
158 ------ [55, 78, 154, 210, 214, 223, 252, 437, 452]
159 ------ [32, 265, 360, 375, 397, 432, 458]
160 ------ [20, 383]
161 ------ [265]
162 ------ [323, 498, 610]
163 ------ [6, 429, 478, 486, 535, 561]
164 ------ [48, 63, 142, 176, 179, 256, 307, 315, 327, 337, 357, 424, 488, 49
1, 507, 508, 515, 518, 524, 543, 546, 595]
165 ------ [179, 357, 518, 524, 596]
166 ------ [182, 183, 265, 418, 611]
167 ------ [601]
168 ------ [299]
169 ------ [31, 62, 109, 181, 265, 282, 397, 611]
170 ------ [70, 340, 383]
171 ------ [50, 82, 202, 406, 459]
172 ------ [143, 265, 581, 599, 611]
173 ------ [369]
174 ------ [11, 212, 259, 264, 318, 380, 485]
175 ------ [87, 195, 265, 576, 611]
176 ------ [164, 216, 220, 301, 362, 427, 478, 497]
177 ------ [350, 389, 510]
178 ------ [495, 539]
179 ------ [31, 123, 164, 165, 292, 347, 357, 491, 518, 524, 596, 605]
180 ------ [209, 331, 496, 520]
181 ------ [67, 70, 109, 169, 235, 248, 258, 320, 340, 352, 363, 476, 517, 52
6, 552]
182 ------ [2, 23, 50, 56, 87, 89, 90, 116, 131, 143, 166, 195, 198, 229, 24
6, 265, 289, 300, 317, 345, 350, 375, 389, 439, 451, 456, 463, 473, 557, 603,
611]
183 ------ [32, 50, 67, 166, 202, 265, 459]
184 ------ [114, 134, 143, 251, 397, 456]
185 ------ [3, 17, 65, 240, 333, 356, 377, 432, 538]
186 ------ [92, 202, 414, 459]
187 ------ [22, 104, 217, 270, 420, 514]
188 ------ [49]
189 ------ [9, 67, 70, 288, 340]
190 ------ [393]
191 ------ [45, 518, 524]
192 ------ [371, 559]
193 ------ [31, 41, 62, 146, 273, 282, 285, 433, 449]
194 ------ [226, 398, 619]
195 ------ [35, 56, 90, 151, 175, 182, 217, 227, 257, 340, 350, 389, 526, 59
7]
196 ------ [49]
```

```
197 ------ [399]
198 ------ [35, 56, 60, 67, 68, 87, 90, 102, 116, 132, 182, 217, 229, 238, 24
5, 265, 317, 340, 350, 355, 372, 432, 505, 558, 577, 580, 597, 601, 603]
199 ------ [316, 518, 566]
200 ------ [114]
201 ------ [114]
202 ------ [171, 183, 186, 288, 340, 343, 426, 555]
203 ------ [345, 435]
204 ------ [215, 540]
205 ------ [70]
206 ------ [9, 90, 473]
207 ------ [372]
208 ------ [34, 90, 128, 136, 288, 340, 343, 446, 455, 601]
209 ------ [33, 180]
210 ------ [55, 78, 154, 158, 223, 252, 437, 452]
211 ------ [96]
212 ------ [11, 174, 259, 264, 380, 485]
213 ------ [502]
214 ------ [55, 78, 154, 158, 223, 533, 534]
215 ------ [204, 278, 405, 540]
216 ------ [98, 101, 176, 402, 618]
217 ------ [56, 87, 89, 90, 102, 130, 131, 187, 195, 198, 229, 245, 265, 289,
317, 350, 404, 413, 439, 473, 510, 576, 577, 580, 585, 589, 597, 601, 603, 61
1, 617]
218 ------ [521]
219 ------ [548]
220 ------ [176, 427, 429, 441, 478, 518, 535]
221 ------ [89, 265]
222 ------ [113, 365, 494, 522]
223 ------ [154, 158, 210, 214, 252, 452, 542]
224 ------ [235, 244, 248, 258, 262, 352, 363, 376, 388, 443, 470, 517, 526,
552, 569]
225 ------ [253, 290, 469, 493, 498, 563, 584]
226 ------ [194, 371, 431]
227 ------ [23, 87, 90, 102, 121, 135, 195, 265, 352, 370, 436, 442, 453, 55
7, 577]
228 ------ [3, 392]
229 ------ [9, 43, 56, 67, 70, 87, 90, 111, 131, 136, 151, 182, 198, 217, 26
5, 289, 317, 340, 361, 389, 396, 436, 440, 445, 446, 505, 509, 549, 554, 576,
597]
230 ------ [24, 26, 28, 108, 147, 279, 306, 438, 501, 502, 527, 547, 591]
231 ------ [425]
232 ------ [134, 146, 253, 498, 563]
233 ------ [265]
234 ------ [618]
235 ------ [56, 70, 181, 224, 244, 258, 352, 354, 363, 368, 472, 517, 526, 55
2, 578]
236 ------ [282]
237 ------ [16, 75, 117, 119, 274, 334, 374, 400, 516, 545, 613]
238 ------ [20, 56, 90, 131, 198, 265, 289, 325, 449, 550, 603, 611, 616]
239 ------ [616]
240 ------ [40, 65, 72, 185, 299, 377, 450, 484, 563, 608]
241 ------ [145, 264, 485]
242 ------ [20, 68, 90, 248, 265, 323, 434, 498, 611]
243 ------ [35, 116, 265, 521]
244 ------ [56, 61, 70, 224, 235, 352, 368, 472]
245 ------ [67, 89, 198, 217, 265, 291, 322, 340, 416, 473, 509, 580, 593, 61
```

```
1, 616]
246 ------ [182, 265]
247 ------ [601]
248 ------ [9, 67, 70, 89, 116, 131, 181, 224, 242, 265, 269, 288, 289, 291,
340, 343, 352, 381, 416, 498, 499, 505, 550, 599, 611]
249 ------ [9, 67, 70, 288, 340]
250 ------ [39, 440, 445]
251 ------ [134, 184, 456, 560]
252 ------ [55, 78, 154, 158, 210, 223, 437, 452]
253 ------ [43, 67, 69, 91, 104, 146, 148, 225, 232, 323, 354, 469, 498, 505,
529, 563]
254 ------ [4, 9, 34, 67, 70, 83, 89, 128, 155, 265, 288, 340, 419, 465, 505,
558, 570]
255 ------ [518]
256 ------ [164, 424, 518]
257 ------ [23, 87, 157, 195, 265, 616]
258 ------ [10, 181, 224, 235, 363, 403, 476]
259 ------ [11, 174, 212, 264, 485]
260 ------ [294, 386, 432, 458]
261 ------ [82, 265, 397, 436, 451]
262 ------ [224, 368, 470]
263 ------ [311, 613]
264 ------ [11, 14, 57, 145, 174, 212, 241, 259, 318, 380, 485, 525]
265 ------ [1, 2, 4, 7, 9, 15, 18, 32, 34, 35, 41, 43, 44, 54, 56, 58, 59, 6
0, 62, 67, 68, 70, 71, 83, 87, 88, 90, 97, 107, 111, 112, 116, 120, 121, 128,
131, 136, 140, 143, 148, 151, 155, 159, 161, 166, 169, 172, 175, 182, 183, 19
8, 217, 221, 227, 229, 233, 238, 242, 243, 245, 246, 248, 254, 257, 261, 271,
275, 277, 288, 300, 302, 309, 312, 317, 320, 321, 322, 324, 338, 340, 341, 34
3, 345, 346, 349, 350, 351, 352, 360, 361, 364, 372, 375, 383, 384, 386, 397,
404, 408, 419, 426, 432, 433, 436, 439, 446, 449, 454, 456, 458, 464, 465, 47
6, 479, 505, 510, 531, 549, 558, 569, 570, 576, 578, 584, 588, 594, 597, 599,
601, 611, 615, 616]
266 ------ [31, 62]
267 ------ [548]
268 ------ [456, 463]
269 ------ [31, 62, 70, 72, 248, 410, 505, 599]
270 ------ [22, 187, 420]
271 ------ [265, 482, 603]
272 ------ [299]
273 ------ [193]
274 ------ [16, 75, 117, 119, 237, 334, 374, 400, 466, 516, 545, 613]
275 ------ [54, 70, 89, 90, 116, 265, 434, 584, 593, 611]
276 ------ [113, 149, 296, 398, 494, 555, 598, 619]
277 ------ [23, 265]
278 ------ [8, 215]
279 ------ [147, 230, 501]
280 ------ [141]
281 ------ [135, 404]
282 ------ [41, 62, 169, 193, 236, 285, 341, 364, 392, 449]
283 ------ [350]
284 ------ [374, 545]
285 ------ [31, 62, 193, 282]
286 ------ [368, 499]
287 ------ [4, 303]
288 ------ [9, 23, 67, 68, 70, 83, 89, 130, 131, 157, 189, 202, 208, 248, 24
9, 254, 265, 289, 317, 323, 340, 342, 350, 351, 419, 434, 455, 459, 465, 528,
536, 550, 570, 576, 593, 611, 616]
```

```
289 ------ [9, 35, 50, 56, 67, 87, 90, 122, 128, 151, 157, 182, 217, 229, 23
8, 248, 288, 317, 325, 340, 343, 351, 505, 510, 597, 601]
290 ------ [81, 104, 225, 469, 498, 563]
291 ------ [67, 107, 245, 248]
292 ------ [85, 123, 179, 307, 328, 335, 421, 518, 612]
293 ------ [7, 49, 155, 570]
294 ------ [260]
295 ------ [98]
296 ------ [276]
297 ------ [448, 518]
298 ------ [360, 386, 389]
299 ------ [3, 17, 65, 168, 240, 272, 356, 377, 538, 575, 577]
300 ------ [32, 89, 182, 265, 303, 564, 581, 611]
301 ------ [176]
302 ------ [58, 265, 536]
303 ------ [287, 300, 336]
304 ------ [369]
305 ------ [7, 106, 339, 583]
306 ------ [24, 230, 547]
307 ------ [164, 292, 335, 357, 518, 524]
308 ------ [125, 555]
309 ------ [265]
310 ------ [12]
311 ------ [101, 137, 144, 150, 152, 263, 329, 353, 374, 441, 503, 545, 613]
312 ------ [265, 389]
313 ------ [357, 488, 518, 524]
314 ------ [446]
315 ------ [31, 164, 518, 524]
316 ------ [199, 518]
317 ------ [9, 46, 67, 70, 83, 89, 90, 107, 135, 182, 198, 217, 229, 265, 28
8, 289, 324, 340, 343, 350, 352, 372, 404, 434, 446, 505, 550, 576, 584, 603,
611, 617]
318 ------ [174, 264, 447]
319 ------ [113, 149, 365, 398, 494, 522, 559, 619]
320 ------ [109, 181, 265]
321 ------ [41, 135, 265, 389, 404]
322 ------ [245, 265]
323 ------ [67, 70, 162, 242, 253, 288, 340, 434, 498, 550, 593, 610]
324 ------ [20, 89, 90, 265, 317, 372, 422, 611, 616]
325 ------ [67, 131, 238, 289, 505]
326 ------ [14]
327 ------ [31, 164, 480, 518]
328 ------ [292, 491, 508, 518, 524, 596]
329 ------ [21, 84, 101, 137, 150, 152, 311, 374, 382, 441, 545, 609]
330 ------ [50, 436]
331 ------ [78, 154, 180]
332 ------ [494]
333 ------ [40, 65, 185]
334 ------ [16, 21, 75, 117, 119, 237, 274, 374, 400, 466, 516, 545, 613]
335 ------ [85, 292, 307, 357, 414, 518]
336 ------ [4, 303]
337 ------ [164, 498]
338 ------ [35, 56, 67, 87, 265, 505, 607]
339 ------ [7, 305, 444, 583]
340 ------ [9, 15, 23, 39, 43, 50, 56, 67, 68, 70, 87, 89, 90, 107, 130, 131,
170, 181, 189, 195, 198, 202, 208, 229, 245, 248, 249, 254, 265, 288, 289, 31
7, 323, 342, 343, 351, 355, 434, 440, 445, 446, 459, 465, 473, 498, 505, 550,
```

```
568, 570, 581, 593, 601, 603, 611, 616]
341 ------ [265, 282]
342 ------ [70, 90, 116, 288, 340, 343, 351, 372, 597]
343 ------ [23, 89, 131, 202, 208, 248, 265, 289, 317, 340, 342, 372, 419, 43
4, 459, 465, 536, 550, 558, 570, 593, 611, 616]
344 ------ [40, 377]
345 ------ [2, 56, 151, 182, 203, 265, 391, 435, 451, 531, 580]
346 ------ [265, 599]
347 ------ [123, 127, 179, 488, 518, 524, 543]
348 ------ [424, 518]
349 ------ [265]
350 ------ [56, 58, 60, 87, 89, 90, 135, 177, 182, 195, 198, 217, 265, 283, 2
88, 317, 389, 404, 413, 434, 446, 463, 603, 611]
351 ------ [9, 56, 67, 89, 90, 131, 265, 288, 289, 340, 342]
352 ------ [3, 13, 40, 67, 181, 224, 227, 235, 244, 248, 265, 317, 368, 385,
434, 470, 498, 526, 616]
353 ------ [311, 374, 441, 545]
354 ------ [146, 235, 253, 498]
355 ------ [20, 90, 130, 198, 340, 550, 603, 616]
356 ------ [185, 299, 377]
357 ------ [31, 45, 48, 63, 79, 118, 127, 164, 165, 179, 307, 313, 335, 395,
430, 448, 491, 504, 508, 515, 518, 524, 532, 543, 572, 596]
358 ------ [113, 467, 494, 555]
359 ------ [56, 135, 404]
360 ------ [43, 135, 159, 265, 298, 364, 389, 404, 479, 568]
361 ------ [229, 265, 550, 576, 603, 611, 616]
362 ------ [176, 429, 478]
363 ------ [25, 181, 224, 235, 258, 368, 388, 401, 526]
364 ------ [32, 265, 282, 360, 397, 432, 458]
365 ------ [222, 319, 398, 522, 555]
366 ------ [116]
367 ------ [539]
368 ------ [235, 244, 262, 286, 352, 363, 470, 476]
369 ------ [8, 73, 173, 304]
370 ------ [227]
371 ------ [192, 226]
372 ------ [67, 90, 116, 198, 207, 265, 317, 324, 342, 343, 389, 465, 550, 60
3, 611, 616]
373 ------ [34, 90, 128]
374 ------ [16, 84, 101, 117, 119, 124, 144, 150, 152, 237, 274, 284, 311, 32
9, 334, 353, 382, 400, 441, 516, 609, 613]
375 ------ [56, 87, 120, 151, 159, 182, 265, 404, 432]
376 ------ [224]
377 ------ [3, 185, 240, 299, 344, 356, 450, 538, 563, 608]
378 ------ [604]
379 ------ [77, 96]
380 ------ [11, 145, 174, 212, 264, 485]
381 ------ [248]
382 ------ [152, 329, 374, 545]
383 ------ [35, 56, 70, 107, 160, 170, 265, 442, 611]
384 ------ [265]
385 ------ [69, 352, 460, 498, 571, 587]
386 ------ [135, 260, 265, 298, 404, 432, 458]
387 ------ [390, 425, 530]
388 ------ [224, 363, 526]
389 ------ [35, 37, 50, 56, 60, 67, 82, 87, 90, 111, 114, 136, 151, 177, 182,
195, 229, 298, 312, 321, 350, 360, 372, 432, 446, 451, 454, 458, 531, 554]
```

```
390 ------ [387]
391 ------ [345]
392 ------ [62, 228, 282]
393 ------ [10, 190, 415, 618]
394 ------ [557]
395 ------ [357, 508, 518, 524]
396 ------ [229]
397 ------ [41, 114, 133, 143, 159, 169, 184, 261, 265, 364, 581, 600]
398 ------ [0, 113, 194, 276, 319, 365, 522, 555, 598]
399 ------ [197, 597]
400 ------ [16, 75, 117, 119, 237, 274, 334, 374, 466, 516, 545, 613]
401 ------ [363]
402 ------ [216]
403 ------ [70, 258, 517, 526]
404 ------ [128, 217, 265, 281, 317, 321, 350, 359, 360, 375, 386, 446, 458,
544]
405 ------ [215, 540]
406 ------ [171]
407 ------ [133, 143]
408 ------ [143, 265, 418, 581, 599]
409 ------ [82]
410 ------ [67, 269, 505]
411 ------ [449]
412 ------ [153, 529]
413 ------ [90, 217, 350, 597]
414 ------ [79, 186, 335, 518]
415 ------ [393]
416 ------ [67, 107, 245, 248]
417 ------ [46, 578]
418 ------ [138, 166, 408, 440, 506]
419 ------ [34, 67, 89, 128, 155, 254, 265, 288, 343, 465, 570]
420 ------ [22, 187, 270]
421 ------ [292, 491, 518, 524, 546, 595]
422 ------ [136, 324]
423 ------ [125, 555]
424 ------ [27, 31, 118, 164, 256, 348, 480, 491, 518]
425 ------ [231, 387]
426 ------ [202, 265, 459, 611]
427 ------ [176, 220, 429, 478, 535]
428 ------ [524]
429 ------ [6, 163, 220, 362, 427, 478, 535]
430 ------ [357, 518]
431 ------ [99, 226]
432 ------ [15, 89, 130, 159, 185, 198, 260, 265, 364, 375, 386, 389, 454, 47
9, 611]
433 ------ [121, 193, 265]
434 ------ [8, 9, 23, 41, 58, 67, 70, 88, 90, 116, 242, 275, 288, 317, 323, 3
40, 343, 350, 352, 505, 584, 597]
435 ------ [203, 345]
436 ------ [227, 229, 261, 265, 330, 489, 531, 575, 611]
437 ------ [55, 78, 154, 158, 210, 252, 452, 521]
438 ------ [147, 230, 527]
439 ------ [35, 56, 60, 70, 87, 90, 136, 182, 217, 265]
440 ------ [67, 229, 250, 340, 418]
441 ------ [21, 137, 144, 152, 220, 311, 329, 353, 374, 545]
442 ------ [129, 227, 383]
443 ------ [224, 590]
```

```
444 ------ [7, 106, 339]
445 ------ [67, 229, 250, 340, 576]
446 ------ [9, 89, 116, 128, 135, 208, 229, 265, 314, 317, 340, 350, 389, 40
4, 505, 558, 584, 603, 611, 616]
447 ------ [318, 606]
448 ------ [63, 297, 357, 487, 518, 572, 585, 602]
449 ------ [31, 62, 193, 238, 265, 282, 411, 450, 553, 608, 611]
450 ------ [3, 17, 65, 240, 377, 449, 538]
451 ------ [82, 111, 182, 261, 345, 389, 554]
452 ------ [55, 78, 154, 158, 210, 223, 252, 437]
453 ------ [227, 597]
454 ------ [15, 265, 389, 432]
455 ------ [208, 288]
456 ------ [134, 143, 182, 184, 251, 265, 268]
457 ------ [518, 524]
458 ------ [135, 159, 260, 265, 364, 386, 389, 404, 479, 611]
459 ------ [171, 183, 186, 288, 340, 343, 426, 555]
460 ------ [88, 385, 498, 571, 582, 587]
461 ------ [45, 141, 574]
462 ------ [526]
463 ------ [182, 268, 350]
464 ------ [265]
465 ------ [9, 41, 56, 67, 70, 83, 89, 128, 254, 265, 288, 340, 343, 372, 41
9, 505]
466 ------ [16, 117, 119, 274, 334, 400, 516, 613]
467 ------ [42, 358, 494, 555]
468 ------ [499]
469 ------ [67, 104, 225, 253, 290, 498, 505, 563]
470 ------ [224, 262, 352, 368]
471 ------ [98, 618]
472 ------ [235, 244]
473 ------ [70, 90, 182, 206, 217, 245, 340]
474 ------ [88, 498, 529, 550]
475 ------ [47]
476 ------ [181, 258, 265, 368]
477 ------ [521]
478 ------ [6, 163, 176, 220, 362, 427, 429, 486, 497, 535]
479 ------ [56, 87, 90, 265, 360, 432, 458]
480 ------ [327, 424, 518]
481 ------ [20]
482 ------ [271]
483 ------ [50, 76, 156]
484 ------ [3, 40, 240]
485 ------ [11, 115, 145, 174, 212, 241, 259, 264, 380, 525]
486 ------ [6, 163, 478]
487 ------ [448, 518]
488 ------ [31, 164, 313, 347, 524, 596]
489 ------ [436]
490 ------ [33, 36]
491 ------ [31, 79, 123, 142, 164, 179, 328, 357, 421, 424, 508, 515, 566, 57
9, 612]
492 ------ [95]
493 ------ [225]
494 ------ [52, 113, 149, 222, 276, 319, 332, 358, 467, 522, 555, 598]
495 ------ [71, 178, 539]
496 ------ [180]
497 ------ [176, 478, 535]
```

```
498 ------ [9, 64, 67, 69, 70, 88, 107, 112, 146, 148, 162, 225, 232, 242, 24
8, 253, 290, 323, 337, 340, 352, 354, 385, 460, 469, 474, 505, 511, 526, 530,
563, 571, 587]
499 ------ [248, 286, 468]
500 ------ [47, 513]
501 ------ [24, 126, 147, 230, 279, 502, 527, 547, 605]
502 ------ [24, 126, 147, 213, 230, 501, 547]
503 ------ [5, 311]
504 ------ [63, 357, 518]
505 ------ [15, 43, 56, 67, 68, 70, 89, 90, 107, 131, 146, 198, 229, 248, 25
3, 254, 265, 269, 289, 317, 325, 338, 340, 410, 434, 446, 465, 469, 498, 550,
570, 592, 593, 599, 611]
506 ------ [418]
507 ------ [31, 79, 85, 118, 164, 546]
508 ------ [31, 47, 164, 328, 357, 395, 491, 518]
509 ------ [39, 229, 245]
510 ------ [131, 157, 177, 217, 265, 289]
511 ------ [498]
512 ------ [49]
513 ------ [500]
514 ------ [187]
515 ------ [31, 164, 357, 491, 518, 612]
516 ------ [16, 21, 75, 117, 119, 237, 274, 334, 374, 400, 466, 545, 613]
517 ------ [181, 224, 235, 403, 526]
518 ------ [6, 31, 38, 45, 48, 63, 79, 85, 100, 118, 123, 127, 139, 142, 164,
165, 179, 191, 199, 220, 255, 256, 292, 297, 307, 313, 315, 316, 327, 328, 33
5, 347, 348, 357, 395, 414, 421, 424, 430, 448, 457, 480, 487, 504, 508, 515,
524, 532, 543, 546, 566, 572, 573, 579, 585, 595, 596, 602, 612]
519 ------ [526, 584]
520 ------ [180]
521 ------ [105, 218, 243, 437, 477]
522 ------ [113, 149, 222, 319, 365, 398, 494, 555, 559, 598, 619]
523 ------ [95, 605]
524 ------ [31, 45, 48, 123, 127, 142, 164, 165, 179, 191, 307, 313, 315, 32
8, 347, 357, 395, 421, 428, 457, 488, 518, 541, 543, 551, 566, 579, 586, 596]
525 ------ [145, 264, 485]
526 ------ [70, 87, 181, 195, 224, 235, 352, 363, 388, 403, 462, 498, 517, 51
9, 536, 552, 558]
527 ------ [28, 108, 147, 230, 438, 501, 547]
528 ------ [288]
529 ------ [253, 412, 474]
530 ------ [387, 498]
531 ------ [67, 265, 345, 389, 436]
532 ------ [357, 518]
533 ------ [154, 214]
534 ------ [214]
535 ------ [163, 220, 427, 429, 478, 497]
536 ------ [35, 54, 70, 116, 136, 288, 302, 343, 526, 558]
537 ------ [58, 96]
538 ------ [40, 185, 299, 377, 450, 608]
539 ------ [178, 367, 495]
540 ------ [204, 215, 405]
541 ------ [524]
542 ------ [223]
543 ------ [31, 164, 347, 357, 518, 524]
544 ------ [50, 135, 404]
545 ------ [16, 84, 101, 117, 119, 124, 144, 150, 152, 237, 274, 284, 311, 32
```

```
9, 334, 353, 382, 400, 441, 516, 609, 613]
546 ------ [31, 79, 164, 421, 507, 518, 595]
547 ------ [24, 108, 147, 230, 306, 501, 502, 527, 591]
548 ------ [12, 105, 219, 267, 556]
549 ------ [229, 265]
550 ------ [9, 46, 67, 70, 88, 90, 107, 238, 248, 288, 317, 323, 340, 343, 35
5, 361, 372, 474, 505, 578, 584]
551 ------ [524]
552 ------ [181, 224, 235, 526]
553 ------ [121, 449]
554 ------ [102, 229, 389, 451, 577]
555 ------ [0, 113, 149, 202, 276, 308, 358, 365, 398, 423, 459, 467, 494, 52
2, 565, 598, 619]
556 ------ [548]
557 ------ [90, 151, 182, 227, 394]
558 ------ [89, 116, 136, 198, 254, 265, 343, 446, 526, 536, 593, 603, 616]
559 ------ [113, 192, 319, 522]
560 ------ [251]
561 ------ [163]
562 ------ [50]
563 ------ [81, 91, 104, 134, 148, 225, 232, 240, 253, 290, 377, 469, 498]
564 ------ [300]
565 ------ [125, 555]
566 ------ [199, 491, 518, 524]
567 ------ [93, 103, 146]
568 ------ [136, 340, 360]
569 ------ [89, 224, 265]
570 ------ [9, 67, 70, 89, 128, 155, 254, 265, 288, 293, 340, 343, 419, 505]
571 ------ [385, 460, 498, 587]
572 ------ [19, 63, 357, 448, 518]
573 ------ [518]
574 ------ [141, 461]
575 ------ [299, 436]
576 ------ [39, 87, 128, 136, 155, 175, 217, 229, 265, 288, 317, 361, 445, 58
4, 593, 601, 603]
577 ------ [43, 56, 67, 70, 198, 217, 227, 299, 554]
578 ------ [46, 50, 67, 86, 110, 235, 265, 417, 550, 588]
579 ------ [491, 518, 524]
580 ------ [87, 90, 143, 151, 198, 217, 245, 345]
581 ------ [32, 140, 143, 172, 300, 340, 397, 408]
582 ------ [460]
583 ------ [7, 106, 305, 339]
584 ------ [56, 82, 89, 90, 225, 265, 275, 317, 434, 446, 519, 550, 576, 603,
611]
585 ------ [157, 217, 448, 518]
586 ------ [524]
587 ------ [385, 460, 498, 571]
588 ------ [265, 578]
589 ------ [217, 601]
590 ------ [443]
591 ------ [24, 147, 230, 547, 605]
592 ------ [505]
593 ------ [9, 67, 70, 116, 245, 275, 288, 323, 340, 343, 505, 558, 576, 597]
594 ------ [41, 265]
595 ------ [31, 79, 164, 421, 518, 546]
596 ------ [142, 165, 179, 328, 357, 488, 518, 524]
597 ------ [35, 56, 87, 89, 90, 130, 131, 151, 195, 198, 217, 229, 265, 289,
```

```
342, 399, 413, 434, 453, 593, 601, 603, 611]
598 ------ [276, 398, 494, 522, 555]
599 ------ [70, 172, 248, 265, 269, 346, 408, 505]
600 ------ [397]
601 ------ [35, 56, 58, 70, 87, 89, 90, 131, 151, 167, 198, 208, 217, 247, 26
5, 289, 340, 576, 589, 597, 603, 616]
602 ------ [63, 448, 518]
603 ------ [0, 39, 56, 58, 67, 68, 90, 151, 157, 182, 198, 217, 238, 271, 31
7, 340, 350, 355, 361, 372, 446, 558, 576, 584, 597, 601, 616]
604 ------ [157, 378]
605 ------ [98, 179, 501, 523, 591]
606 ------ [447]
607 ------ [338]
608 ------ [3, 17, 65, 240, 377, 449, 538]
609 ------ [329, 374, 545]
610 ------ [162, 323]
611 ------ [1, 9, 32, 35, 41, 56, 58, 62, 67, 70, 87, 90, 116, 136, 151, 166,
169, 172, 175, 182, 217, 238, 242, 245, 248, 265, 275, 288, 300, 317, 324, 34
0, 343, 350, 361, 372, 383, 426, 432, 436, 446, 449, 458, 505, 584, 597, 615]
612 ------ [292, 491, 515, 518]
613 ------ [16, 75, 117, 119, 124, 150, 237, 263, 274, 311, 334, 374, 400, 46
6, 516, 545]
614 ------ [93]
615 ------ [35, 265, 611]
616 ------ [56, 58, 70, 87, 90, 116, 238, 239, 245, 257, 265, 288, 324, 340,
343, 352, 355, 361, 372, 446, 558, 601, 603]
617 ------ [56, 217, 317]
618 ------ [12, 35, 216, 234, 393, 471]
619 ------ [194, 276, 319, 522, 555]
```

(c)Comment on the 'sparseness' of the network.

As there is no self loops the maximum number possile eges present in the graph is n(n-1)/2 where n is number of nodes in the graph here n*n(n-1)/2 is 191890 which is too much coampre to current present edges that is 2090 . So graph is highly sparse because of less number of edges present. here.

In [64]:
```
print("overall there are n*(n-1)/2 edges in the graph where value of n(nodes)
 is : ",len(n))
a=int((len(n)*(len(n)-1))/2)
print("therefor n*(n-1)/2 vlaue is : ",a)
b=int(total_edges/2)
print("Actual number of edges with ",len(n)," nodes are : ",b)
print("difference between : maximum edges - actual edges : ",(a-b))
print("therefor the graph is too sparse")
```

```
overall there are n*(n-1)/2 edges in the graph where value of n(nodes) is :
620
therefor n*(n-1)/2 vlaue is :  191890
Actual number of edges with  620  nodes are :  2090
difference between : maximum edges - actual edges :  189800
therefor the graph is too sparse
```

(d) Compute its average degree < k >.

In [65]:
```
print("Degree of each nodes are : ")
pp.pprint(degree_of_nodes)
val=0
for key,value in degree_of_nodes.items():   #here count of frquency of degrees
in degree_of_nodes dictionary where key is degree and values if frquency
  val+=value
print("average degree: ",val/len(n))   #here avergae dgeree is taken out by fo
rmula :total value sum of degrees/total nodes
```

```
Degree of each nodes are :
{         0: 5,
          1: 2,
          2: 3,
          3: 10,
          4: 4,
          5: 1,
          6: 5,
          7: 7,
          8: 5,
          9: 28,
          10: 2,
          11: 6,
          12: 3,
          13: 1,
          14: 2,
          15: 10,
          16: 12,
          17: 5,
          18: 1,
          19: 1,
          20: 6,
          21: 4,
          22: 4,
          23: 14,
          24: 6,
          25: 1,
          26: 1,
          27: 1,
          28: 3,
          29: 1,
          30: 1,
          31: 25,
          32: 8,
          33: 2,
          34: 6,
          35: 22,
          36: 1,
          37: 1,
          38: 1,
          39: 6,
          40: 7,
          41: 13,
          42: 1,
          43: 12,
          44: 1,
          45: 6,
          46: 7,
          47: 3,
          48: 5,
          49: 6,
          50: 14,
          51: 1,
          52: 2,
          53: 1,
          54: 5,
          55: 8,
```

```
56: 47,
57: 2,
58: 15,
59: 2,
60: 6,
61: 1,
62: 11,
63: 10,
64: 1,
65: 8,
66: 1,
67: 56,
68: 10,
69: 3,
70: 47,
71: 3,
72: 3,
73: 2,
74: 1,
75: 9,
76: 1,
77: 1,
78: 9,
79: 8,
80: 1,
81: 2,
82: 6,
83: 7,
84: 3,
85: 5,
86: 2,
87: 30,
88: 6,
89: 43,
90: 51,
91: 2,
92: 1,
93: 5,
94: 2,
95: 2,
96: 3,
97: 1,
98: 7,
99: 4,
100: 1,
101: 7,
102: 8,
103: 3,
104: 6,
105: 2,
106: 3,
107: 12,
108: 5,
109: 4,
110: 2,
111: 4,
112: 2,
```

```
113: 10,
114: 8,
115: 2,
116: 22,
117: 13,
118: 4,
119: 12,
120: 2,
121: 5,
122: 3,
123: 6,
124: 4,
125: 3,
126: 4,
127: 4,
128: 16,
129: 2,
130: 10,
131: 26,
132: 2,
133: 2,
134: 7,
135: 12,
136: 14,
137: 5,
138: 3,
139: 2,
140: 2,
141: 5,
142: 7,
143: 13,
144: 5,
145: 6,
146: 9,
147: 11,
148: 5,
149: 5,
150: 6,
151: 21,
152: 8,
153: 1,
154: 12,
155: 8,
156: 1,
157: 10,
158: 9,
159: 7,
160: 2,
161: 1,
162: 3,
163: 6,
164: 22,
165: 5,
166: 5,
167: 1,
168: 1,
169: 8,
```

```
170: 3,
171: 5,
172: 5,
173: 1,
174: 7,
175: 5,
176: 8,
177: 3,
178: 2,
179: 12,
180: 4,
181: 15,
182: 31,
183: 7,
184: 6,
185: 9,
186: 4,
187: 6,
188: 1,
189: 5,
190: 1,
191: 3,
192: 2,
193: 9,
194: 3,
195: 14,
196: 1,
197: 1,
198: 29,
199: 3,
200: 1,
201: 1,
202: 8,
203: 2,
204: 2,
205: 1,
206: 3,
207: 1,
208: 10,
209: 2,
210: 8,
211: 1,
212: 6,
213: 1,
214: 7,
215: 4,
216: 5,
217: 31,
218: 1,
219: 1,
220: 7,
221: 2,
222: 4,
223: 7,
224: 15,
225: 7,
226: 3,
```

```
227: 15,
228: 2,
229: 31,
230: 13,
231: 1,
232: 5,
233: 1,
234: 1,
235: 15,
236: 1,
237: 11,
238: 13,
239: 1,
240: 10,
241: 3,
242: 9,
243: 4,
244: 8,
245: 15,
246: 2,
247: 1,
248: 25,
249: 5,
250: 3,
251: 4,
252: 8,
253: 16,
254: 17,
255: 1,
256: 3,
257: 6,
258: 7,
259: 5,
260: 4,
261: 5,
262: 3,
263: 2,
264: 12,
265: 132,
266: 2,
267: 1,
268: 2,
269: 8,
270: 3,
271: 3,
272: 1,
273: 1,
274: 12,
275: 10,
276: 8,
277: 2,
278: 2,
279: 3,
280: 1,
281: 2,
282: 10,
283: 1,
```

```
284: 2,
285: 4,
286: 2,
287: 2,
288: 37,
289: 26,
290: 6,
291: 4,
292: 9,
293: 4,
294: 1,
295: 1,
296: 1,
297: 2,
298: 3,
299: 11,
300: 8,
301: 1,
302: 3,
303: 3,
304: 1,
305: 4,
306: 3,
307: 6,
308: 2,
309: 1,
310: 1,
311: 13,
312: 2,
313: 4,
314: 1,
315: 4,
316: 2,
317: 32,
318: 3,
319: 8,
320: 3,
321: 5,
322: 2,
323: 12,
324: 9,
325: 5,
326: 1,
327: 4,
328: 6,
329: 12,
330: 2,
331: 3,
332: 1,
333: 3,
334: 13,
335: 6,
336: 2,
337: 2,
338: 7,
339: 4,
340: 55,
```

```
341: 2,
342: 9,
343: 23,
344: 2,
345: 11,
346: 2,
347: 7,
348: 2,
349: 1,
350: 24,
351: 11,
352: 19,
353: 4,
354: 4,
355: 8,
356: 3,
357: 26,
358: 4,
359: 3,
360: 10,
361: 7,
362: 3,
363: 9,
364: 7,
365: 5,
366: 1,
367: 1,
368: 8,
369: 4,
370: 1,
371: 2,
372: 16,
373: 3,
374: 22,
375: 9,
376: 1,
377: 10,
378: 1,
379: 2,
380: 6,
381: 1,
382: 4,
383: 9,
384: 1,
385: 6,
386: 7,
387: 3,
388: 3,
389: 30,
390: 1,
391: 1,
392: 3,
393: 4,
394: 1,
395: 4,
396: 1,
397: 12,
```

```
398: 9,
399: 2,
400: 12,
401: 1,
402: 1,
403: 4,
404: 14,
405: 2,
406: 1,
407: 2,
408: 5,
409: 1,
410: 3,
411: 1,
412: 2,
413: 4,
414: 4,
415: 1,
416: 4,
417: 2,
418: 5,
419: 11,
420: 3,
421: 6,
422: 2,
423: 2,
424: 9,
425: 2,
426: 4,
427: 5,
428: 1,
429: 7,
430: 2,
431: 2,
432: 15,
433: 3,
434: 22,
435: 2,
436: 9,
437: 8,
438: 3,
439: 10,
440: 5,
441: 10,
442: 3,
443: 2,
444: 3,
445: 5,
446: 20,
447: 2,
448: 8,
449: 11,
450: 7,
451: 7,
452: 8,
453: 2,
454: 4,
```

```
455: 2,
456: 7,
457: 2,
458: 10,
459: 8,
460: 6,
461: 3,
462: 1,
463: 3,
464: 1,
465: 16,
466: 8,
467: 4,
468: 1,
469: 8,
470: 4,
471: 2,
472: 2,
473: 7,
474: 4,
475: 1,
476: 4,
477: 1,
478: 10,
479: 7,
480: 3,
481: 1,
482: 1,
483: 3,
484: 3,
485: 10,
486: 3,
487: 2,
488: 6,
489: 1,
490: 2,
491: 15,
492: 1,
493: 1,
494: 12,
495: 3,
496: 1,
497: 3,
498: 33,
499: 3,
500: 2,
501: 9,
502: 7,
503: 2,
504: 3,
505: 35,
506: 1,
507: 6,
508: 8,
509: 3,
510: 6,
511: 1,
```

```
512: 1,
513: 1,
514: 1,
515: 6,
516: 13,
517: 5,
518: 59,
519: 2,
520: 1,
521: 5,
522: 11,
523: 2,
524: 29,
525: 3,
526: 17,
527: 7,
528: 1,
529: 3,
530: 2,
531: 5,
532: 2,
533: 2,
534: 1,
535: 6,
536: 10,
537: 2,
538: 6,
539: 3,
540: 3,
541: 1,
542: 1,
543: 6,
544: 3,
545: 22,
546: 7,
547: 9,
548: 5,
549: 2,
550: 21,
551: 1,
552: 4,
553: 2,
554: 5,
555: 17,
556: 1,
557: 5,
558: 13,
559: 4,
560: 1,
561: 1,
562: 1,
563: 13,
564: 1,
565: 2,
566: 4,
567: 3,
568: 3,
```

```
 569: 3,
 570: 14,
 571: 4,
 572: 5,
 573: 1,
 574: 2,
 575: 2,
 576: 17,
 577: 9,
 578: 10,
 579: 3,
 580: 8,
 581: 8,
 582: 1,
 583: 4,
 584: 15,
 585: 4,
 586: 1,
 587: 4,
 588: 2,
 589: 2,
 590: 1,
 591: 5,
 592: 1,
 593: 14,
 594: 2,
 595: 6,
 596: 8,
 597: 23,
 598: 5,
 599: 8,
 600: 1,
 601: 22,
 602: 3,
 603: 27,
 604: 2,
 605: 5,
 606: 1,
 607: 1,
 608: 7,
 609: 3,
 610: 2,
 611: 47,
 612: 4,
 613: 16,
 614: 1,
 615: 3,
 616: 23,
 617: 3,
 618: 6,
 619: 5}
average degree:  6.741935483870968
```

(e) Plot its 'scaled degree distribution', pk × k.

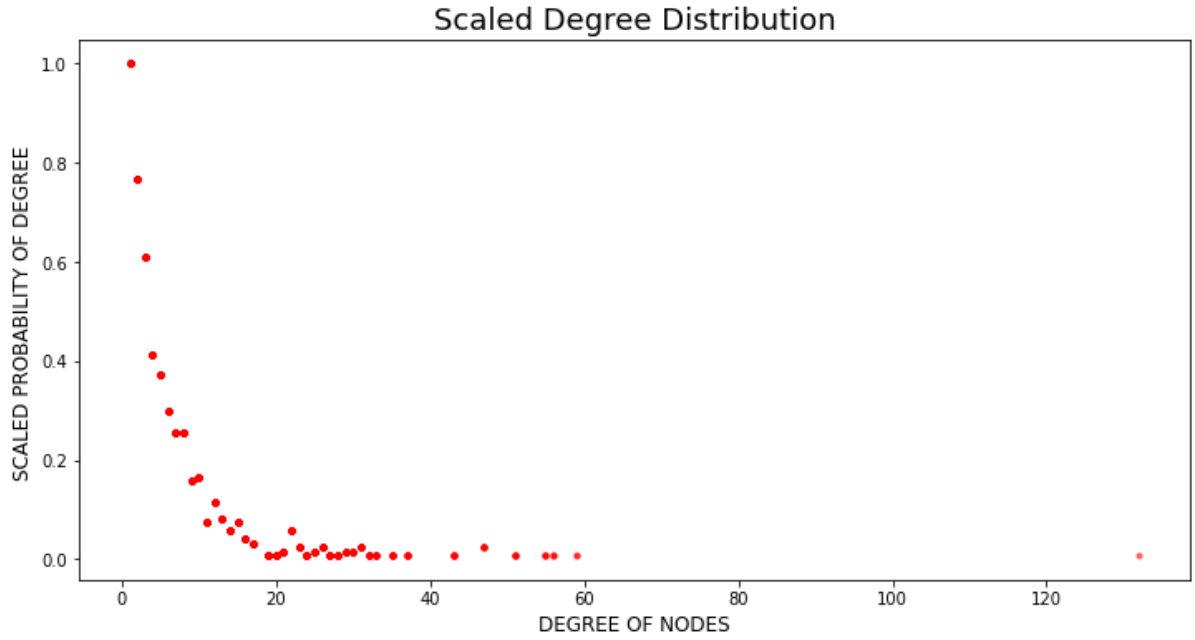
In [66]:
```
degree_frequncy={} # here frquncy count of each degree in degree_frequncy
for key,value in degree_of_nodes.items():
  if value not in degree_frequncy.keys():   #if particular frquncy is not arri
ved then put it there
    degree_frequncy[value]=0
    degree_frequncy[value]+=1
  else:
    degree_frequncy[value]+=1     ##here is particular degree is repeating the
n keep it counting
```

In [67]:
```
kmax=0
cmax=0
for key,value in degree_frequncy.items(): #for finding maximum degree kmax in
 the graph
  if cmax< value:
    kmax=key
    cmax=value
```

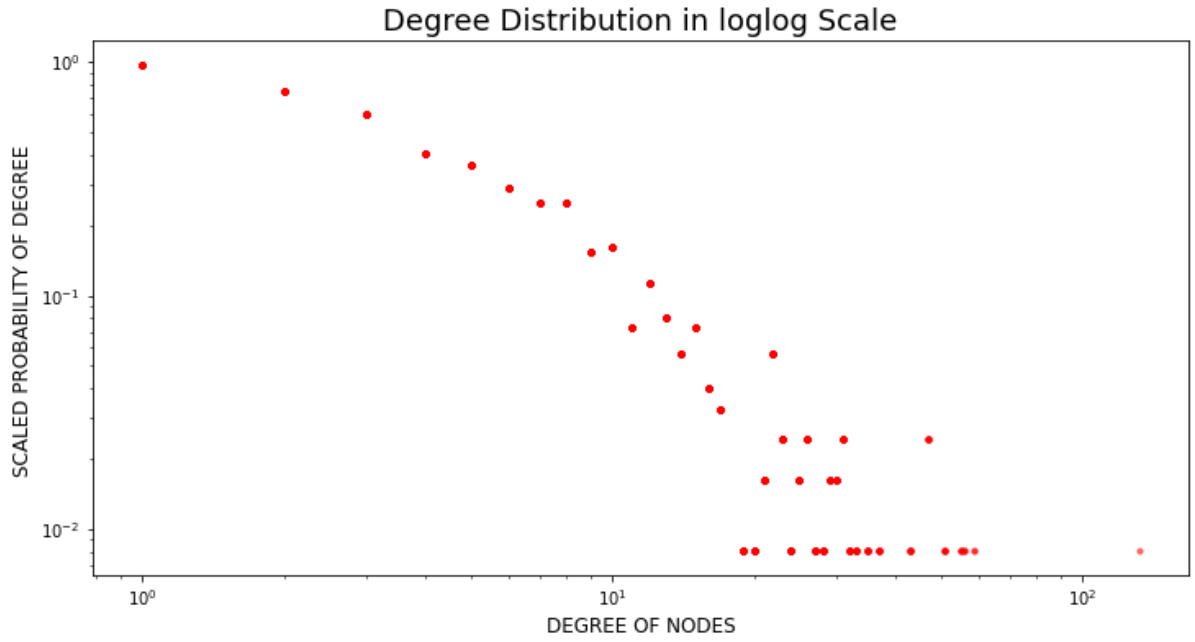
In [68]:
```
keyMax = degree_frequncy[max(degree_frequncy.items(), key = operator.itemgette
r(1))[0] ]
```

In [69]:
```
degree_frequncy = dict(sorted(degree_frequncy.items(), key=lambda x: x[0]))
```

In [70]:
```
import matplotlib.pyplot as plt
import numpy as np
fg, ax = plt.subplots(figsize =(12, 6))
x_axis=[]
y_axis=[]
kmax=max(list(degree_frequncy.keys()))
for key,value in  degree_frequncy.items():
  x_axis.append(key)
  y_axis.append((value/keyMax))
  ax.scatter(x_axis,y_axis,s=np.pi*3.2,c=("red"), alpha=0.5)

plt.xlabel("DEGREE OF NODES", fontsize=12)
plt.ylabel("SCALED PROBABILITY OF DEGREE ", fontsize=12)
plt.title("Scaled Degree Distribution",fontsize=18)
plt.show()
```



In [71]:
```
#Loglog Scaled Degree Distribution

import matplotlib.pyplot as plt
import numpy as np
fg, ax = plt.subplots(figsize =(12, 6))
x_axis=[]
y_axis=[]
for key,value in  degree_frequncy.items():
  x_axis.append(key)
  y_axis.append((value/len(n))/0.2)
  ax.scatter(x_axis,y_axis,s=np.pi*3.2,c=("red"), alpha=0.5)
  plt.yscale('log')
  plt.xscale('log')

plt.xlabel("DEGREE OF NODES", fontsize=12)
plt.ylabel("SCALED PROBABILITY OF DEGREE ", fontsize=12)
plt.title("Degree Distribution in loglog Scale",fontsize=18)
plt.show()
```



(f) Compute its Average Path Length (Implement Breadth First Search Algorithm), Diameter and Average Clustering Coefficient.

In [72]:
```
distance={}  #distance dictionary for storing of distance form src node(i) to
 destination node(j)
for i in range(0,len(adjancency_matrix)):  #initialization of distance matrix
 by 0 for all nodes
  distance[i]={}
  for j in range(0,len(adjancency_matrix[0])):
    distance[i][j]=0
```

Implement Breadth First Search Algorithm

In [73]:
```
# edge_list
sum=0
for i in range(0,len(adjancency_matrix)):  #loop for rows
  for j in range(i+1,len(adjancency_matrix)): #loop for columns
    if i!=j:  #if nodes are not same then start the algorithm
      temp={}  #temporary dictionary for assigneding distance of all the neigh
bor nodes of source node
      queue=[] #queues act as a visited list for node , here adjancent node in
serted in the queueu
      if len(edge_list[i])!=0:    #if adjancent nodes of source node are prese
nt then make every adjacent nodes distance to 1 intially
        for x in edge_list[i]:
          temp[x]=1
          queue.append(x)

      while(len(queue)!=0):    #run this loop till we got the destination and
 till loop is not not empty.
        x=queue[0]                #take out node and assign to x
        queue.remove(queue[0])     #remove 1st node from the queue
        if x==j:                    #removed node is destination then stop the
algorihtm and run for next destination
          distance[i][j]=temp[x]    #assign the distance of src i to j from te
mp x .
          sum+=temp[x]
          break                       #then break
        else:
          for l in edge_list[x]:     #else take out all the adjancent neighbor
s and add into the queue
            if l not in temp.keys():
              temp[l]=temp[x]+1                     #along with intialse the
 distance of new node as per distance of remove node from source node+1
              queue.append(l)                # then append in queue
```

Averegae path length

In [74]:
```
print("Average path length is: sum of distances/nC2 :",((2*sum)/(len(n)*(len(n
)-1))))
```

```
Average path length is: sum of distances/nC2 : 5.092605138360519
```

diameter:

In [75]:
```
diameter=0 #initially diameter is equal to 0
for i in distance.keys():    #take the maximum distance as a diameter if the g
raph
  diameter=max(diameter,max(distance[i].values()))
```

In [76]:
```
print("diameter of the graph is :",diameter)
```

```
diameter of the graph is : 17
```

Average cluster coefficient:

cluster coefficient formula= Li /nC2

where Li is number of edges between the adjancent if node i and n is the total adjanct nodes of node i.

In [77]:
```
cluster_coefficient={}  #intitilialise the cluster coeffienct of nodes
for i in range(0,len(adjancency_matrix)):
  li=0
  for j in range(0,len(edge_list[i])):  #Loop for checking if there exist and
 edge between adjanct nodes of the source node i
    for k in range(j+1, len(edge_list[i])):
      if adjancency_matrix[edge_list[i][j]][edge_list[i][k]]==1 :  # if node e
xist the incerment Li
        li+=1
  if li!=0:
    cluster_coefficient[i]=(2*li)/(len(edge_list[i])*(len(edge_list[i])-1))
#here calculate cluster coeffcient by particular formula
```

In [78]:
```
temp_sum=0
for i in list(cluster_coefficient.values()):   #total sum for cluster coeffice
nt of each nodes
  temp_sum+=i
```

In [79]:
```
print("Average cluster coefficent: ",temp_sum/len(n))  #average cluster coeffi
cent is total sum of each cluster coefficent / total numbe rof nodes
```

```
Average cluster coefficent:  0.33078404030069036
```

In [80]:
```
cluster_coefficient #cluster coeffcient of each nodes.
```

Out[80]:
```
{0: 0.2,
 1: 1.0,
 2: 1.0,
 3: 0.2222222222222222,
 4: 0.16666666666666666,
 6: 0.5,
 7: 0.23809523809523808,
 8: 0.1,
 9: 0.3439153439153439,
 11: 0.9333333333333333,
 15: 0.5777777777777777,
 16: 0.8939393939393939,
 17: 0.4,
 21: 0.3333333333333333,
 22: 0.6666666666666666,
 23: 0.16483516483516483,
 24: 0.6666666666666666,
 28: 1.0,
 31: 0.12666666666666668,
 32: 0.2857142857142857,
 34: 0.3333333333333333,
 35: 0.2987012987012987,
 39: 0.2,
 40: 0.14285714285714285,
 41: 0.15384615384615385,
 43: 0.3181818181818182,
 45: 0.4666666666666667,
 46: 0.38095238095238093,
 48: 0.9,
 50: 0.14285714285714285,
 52: 1.0,
 54: 0.2,
 55: 0.8571428571428571,
 56: 0.2506938020351526,
 58: 0.34285714285714286,
 59: 1.0,
 60: 0.6666666666666666,
 62: 0.2909090909090909,
 63: 0.37777777777777777,
 65: 0.4642857142857143,
 67: 0.18181818181818182,
 68: 0.6444444444444445,
 69: 0.6666666666666666,
 70: 0.18038852913968548,
 72: 0.3333333333333333,
 73: 1.0,
 75: 1.0,
 78: 0.6944444444444444,
 79: 0.32142857142857145,
 81: 1.0,
 82: 0.13333333333333333,
 83: 0.6190476190476191,
 84: 0.6666666666666666,
 85: 0.5,
 86: 1.0,
 87: 0.34022988505747126,
 88: 0.2,
```

```
89: 0.1849390919158361,
90: 0.25019607843137254,
91: 1.0,
93: 0.3,
94: 1.0,
101: 0.5238095238095238,
102: 0.2857142857142857,
103: 0.6666666666666666,
104: 0.4,
106: 0.3333333333333333,
107: 0.5151515151515151,
108: 0.6,
109: 0.3333333333333333,
110: 1.0,
111: 0.5,
113: 0.4,
114: 0.17857142857142858,
115: 1.0,
116: 0.29004329004329005,
117: 0.7564102564102564,
118: 0.3333333333333333,
119: 0.8939393939393939,
120: 1.0,
121: 0.3,
123: 0.6,
124: 0.6666666666666666,
126: 0.5,
127: 0.8333333333333334,
128: 0.21666666666666667,
130: 0.3111111111111111,
131: 0.32,
134: 0.38095238095238093,
135: 0.07575757575757576,
136: 0.13186813186813187,
137: 0.6,
139: 1.0,
141: 0.1,
142: 0.42857142857142855,
143: 0.23076923076923078,
144: 0.8,
145: 0.5333333333333333,
146: 0.3055555555555556,
147: 0.4,
148: 0.3,
149: 0.7,
150: 0.7333333333333333,
151: 0.3523809523809524,
152: 0.6785714285714286,
154: 0.4696969696969697,
155: 0.5,
157: 0.2,
158: 0.8055555555555556,
159: 0.3333333333333333,
162: 0.6666666666666666,
163: 0.4666666666666667,
164: 0.17316017316017315,
165: 1.0,
```

```
166: 0.4,
169: 0.21428571428571427,
170: 0.6666666666666666,
172: 0.4,
174: 0.7142857142857143,
175: 0.5,
176: 0.17857142857142858,
177: 0.3333333333333333,
178: 1.0,
179: 0.4090909090909091,
181: 0.22857142857142856,
182: 0.2021505376344086,
183: 0.19047619047619047,
184: 0.6,
185: 0.2222222222222222,
187: 0.26666666666666666,
189: 0.9,
191: 1.0,
193: 0.2777777777777778,
195: 0.2967032967032967,
198: 0.3054187192118227,
199: 0.6666666666666666,
202: 0.07142857142857142,
203: 1.0,
204: 1.0,
206: 0.6666666666666666,
208: 0.17777777777777778,
210: 0.8928571428571429,
212: 0.9333333333333333,
214: 0.42857142857142855,
215: 0.3333333333333333,
217: 0.24731182795698925,
220: 0.38095238095238093,
222: 0.6666666666666666,
223: 0.5714285714285714,
224: 0.17142857142857143,
225: 0.42857142857142855,
227: 0.0761904761904762,
229: 0.2731182795698925,
230: 0.3333333333333333,
232: 0.6,
235: 0.26666666666666666,
237: 0.9454545454545454,
238: 0.3333333333333333,
240: 0.2222222222222222,
241: 1.0,
242: 0.2777777777777778,
243: 0.3333333333333333,
244: 0.32142857142857145,
245: 0.24761904761904763,
246: 1.0,
248: 0.27666666666666667,
249: 0.9,
251: 0.5,
252: 0.8928571428571429,
253: 0.24166666666666667,
254: 0.47794117647058826,
```

```
256: 1.0,
257: 0.26666666666666666,
258: 0.2857142857142857,
259: 1.0,
260: 0.3333333333333333,
261: 0.3,
262: 0.6666666666666666,
263: 1.0,
264: 0.3181818181818182,
265: 0.05331945408281286,
269: 0.2857142857142857,
270: 1.0,
274: 0.8939393939393939,
275: 0.5555555555555556,
276: 0.39285714285714285,
279: 1.0,
282: 0.2,
285: 0.6666666666666666,
288: 0.21021021021021022,
289: 0.28615384615384615,
290: 0.6,
291: 0.5,
292: 0.3055555555555556,
293: 0.16666666666666666,
297: 1.0,
298: 0.3333333333333333,
299: 0.12727272727272726,
300: 0.2857142857142857,
302: 0.3333333333333333,
305: 0.6666666666666666,
306: 1.0,
307: 0.6666666666666666,
311: 0.38461538461538464,
313: 0.6666666666666666,
315: 0.8333333333333334,
316: 1.0,
317: 0.3387096774193548,
318: 0.3333333333333333,
319: 0.42857142857142855,
320: 0.3333333333333333,
321: 0.2,
322: 1.0,
323: 0.4090909090909091,
324: 0.4166666666666667,
325: 0.7,
327: 0.5,
328: 0.4,
329: 0.4090909090909091,
331: 0.3333333333333333,
333: 0.3333333333333333,
334: 0.7692307692307693,
335: 0.5333333333333333,
338: 0.5238095238095238,
339: 0.6666666666666666,
340: 0.19932659932659932,
342: 0.4166666666666667,
343: 0.2134387351778656,
```

```
345: 0.2,
346: 1.0,
347: 0.5238095238095238,
348: 1.0,
350: 0.31521739130434784,
351: 0.7454545454545455,
352: 0.16374269005847952,
353: 0.8333333333333334,
354: 0.5,
355: 0.4642857142857143,
356: 0.6666666666666666,
357: 0.2,
358: 0.8333333333333334,
360: 0.13333333333333333,
361: 0.3333333333333333,
362: 0.6666666666666666,
363: 0.3055555555555556,
364: 0.23809523809523808,
365: 0.6,
368: 0.21428571428571427,
369: 0.16666666666666666,
372: 0.39166666666666666,
374: 0.2597402597402597,
375: 0.3888888888888889,
377: 0.3333333333333333,
380: 0.8,
382: 0.8333333333333334,
383: 0.3333333333333333,
385: 0.5333333333333333,
386: 0.3333333333333333,
388: 1.0,
389: 0.11264367816091954,
392: 0.3333333333333333,
395: 0.8333333333333334,
397: 0.15151515151515152,
398: 0.3888888888888889,
400: 0.8939393939393939,
403: 0.3333333333333333,
404: 0.18681318681318682,
405: 1.0,
408: 0.3,
410: 0.6666666666666666,
413: 0.8333333333333334,
414: 0.3333333333333333,
416: 0.5,
417: 1.0,
419: 0.6363636363636364,
420: 1.0,
421: 0.3333333333333333,
424: 0.25,
426: 0.16666666666666666,
427: 0.8,
429: 0.5238095238095238,
430: 1.0,
432: 0.12380952380952381,
433: 0.3333333333333333,
434: 0.23376623376623376,
```

```
435: 1.0,
436: 0.1388888888888889,
437: 0.75,
438: 1.0,
439: 0.5333333333333333,
440: 0.3,
441: 0.4222222222222222,
444: 0.3333333333333333,
445: 0.4,
446: 0.3157894736842105,
448: 0.39285714285714285,
449: 0.16363636363636364,
450: 0.2857142857142857,
451: 0.2857142857142857,
452: 0.8928571428571429,
454: 0.6666666666666666,
455: 1.0,
456: 0.38095238095238093,
457: 1.0,
458: 0.2,
459: 0.07142857142857142,
460: 0.4666666666666667,
461: 0.3333333333333333,
463: 0.3333333333333333,
465: 0.5083333333333333,
466: 1.0,
467: 0.5,
469: 0.6071428571428571,
470: 0.6666666666666666,
472: 1.0,
473: 0.38095238095238093,
474: 0.3333333333333333,
476: 0.16666666666666666,
478: 0.4,
479: 0.42857142857142855,
480: 0.6666666666666666,
484: 0.6666666666666666,
485: 0.4666666666666667,
486: 1.0,
487: 1.0,
488: 0.3333333333333333,
491: 0.18095238095238095,
494: 0.3181818181818182,
495: 0.3333333333333333,
497: 0.6666666666666666,
498: 0.11931818181818182,
501: 0.4444444444444444,
502: 0.5238095238095238,
504: 1.0,
505: 0.2605042016806723,
507: 0.26666666666666666,
508: 0.42857142857142855,
510: 0.4,
515: 0.6666666666666666,
516: 0.7692307692307693,
517: 0.6,
518: 0.07130333138515488,
```

```
522: 0.43636363636363634,
524: 0.15270935960591134,
525: 1.0,
526: 0.17647058823529413,
527: 0.6190476190476191,
531: 0.4,
532: 1.0,
533: 1.0,
535: 0.6,
536: 0.15555555555555556,
538: 0.26666666666666666,
539: 0.3333333333333333,
540: 0.6666666666666666,
543: 0.7333333333333333,
545: 0.2597402597402597,
546: 0.5714285714285714,
547: 0.5555555555555556,
549: 1.0,
550: 0.2714285714285714,
552: 0.8333333333333334,
554: 0.2,
555: 0.17647058823529413,
557: 0.3,
558: 0.3333333333333333,
559: 0.5,
563: 0.2564102564102564,
566: 0.3333333333333333,
567: 0.6666666666666666,
570: 0.5494505494505495,
571: 1.0,
572: 0.6,
574: 1.0,
576: 0.2426470588235294,
577: 0.2222222222222222,
578: 0.2,
579: 0.3333333333333333,
580: 0.42857142857142855,
581: 0.14285714285714285,
583: 0.6666666666666666,
584: 0.3047619047619048,
585: 0.16666666666666666,
587: 1.0,
588: 1.0,
589: 1.0,
591: 0.5,
593: 0.27472527472527475,
594: 1.0,
595: 0.6,
596: 0.5357142857142857,
597: 0.35177865612648224,
598: 0.8,
599: 0.42857142857142855,
601: 0.41125541125541126,
602: 1.0,
603: 0.25071225071225073,
608: 0.2857142857142857,
609: 0.6666666666666666,
```

```
610: 1.0,
611: 0.15356151711378355,
612: 0.5,
613: 0.5583333333333333,
615: 1.0,
616: 0.2727272727272727,
617: 0.6666666666666666,
619: 0.3}
```

In [80]: