11508 Muzamil Khan Assignment 3

email-Eu-core Dataset

Non-overlapping/Disjoint Algorithm

1. Reading the graph from the edge list file

In [19]:
```
import networkx as nx

# Read the graph from an edge list file
graph = nx.read_edgelist('email-Eu-core.txt', create_using=nx.Graph(), nodetype=int)
# Number of nodes
print("Number of nodes:", graph.number_of_nodes())
# Number of edges
print("Number of edges:", graph.number_of_edges())
# Average degree
print("Average degree:", sum(dict(graph.degree()).values()) / graph.number_of_nodes())
```

```
Number of nodes: 1005
Number of edges: 16706
Average degree: 33.245771144278606
```

2. Applying the Girvan-Newman Algorithm

In [20]:
```
from networkx.algorithms.community import girvan_newman

# Apply Girvan-Newman algorithm
communities = girvan_newman(graph)
top_level_communities = next(communities)
sorted_communities = sorted(map(sorted, top_level_communities))
print("Girvan-Newman communities:", sorted_communities)
```

```
Girvan-Newman communities: [[0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 2
3, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 5
1, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 7
9, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96, 97, 98, 99, 100, 101, 102, 103, 104, 105,
106, 107, 108, 109, 110, 111, 112, 113, 114, 115, 116, 117, 118, 119, 120, 121, 122, 123, 124, 125, 126, 127, 12
8, 129, 130, 131, 132, 133, 134, 135, 136, 137, 138, 139, 140, 141, 142, 143, 144, 145, 146, 147, 148, 149, 150,
151, 152, 153, 154, 155, 156, 157, 158, 159, 160, 161, 162, 163, 164, 165, 166, 167, 168, 169, 170, 171, 172, 17
3, 174, 175, 176, 177, 178, 179, 180, 181, 182, 183, 184, 185, 186, 187, 188, 189, 190, 191, 192, 193, 194, 195,
196, 197, 198, 199, 200, 201, 202, 203, 204, 205, 206, 207, 208, 209, 210, 211, 212, 213, 214, 215, 216, 217, 21
8, 219, 220, 221, 222, 223, 224, 225, 226, 227, 228, 229, 230, 231, 232, 233, 234, 235, 236, 237, 238, 239, 240,
241, 242, 243, 244, 245, 246, 247, 248, 249, 250, 251, 252, 253, 254, 255, 256, 257, 258, 259, 260, 261, 262, 26
3, 264, 265, 266, 267, 268, 269, 270, 271, 272, 273, 274, 275, 276, 277, 278, 279, 280, 281, 282, 283, 284, 285,
286, 287, 288, 289, 290, 291, 292, 293, 294, 295, 296, 297, 298, 299, 300, 301, 302, 303, 304, 305, 306, 307, 30
8, 309, 310, 311, 312, 313, 314, 315, 316, 317, 318, 319, 320, 321, 322, 323, 324, 325, 326, 327, 328, 329, 330,
331, 332, 333, 334, 335, 336, 337, 338, 339, 340, 341, 342, 343, 344, 345, 346, 347, 348, 349, 350, 351, 352, 35
3, 354, 355, 356, 357, 358, 359, 360, 361, 362, 363, 364, 365, 366, 367, 368, 369, 370, 371, 372, 373, 374, 375,
376, 377, 378, 379, 380, 381, 382, 383, 384, 385, 386, 387, 388, 389, 390, 391, 392, 393, 394, 395, 396, 397, 39
8, 399, 400, 401, 402, 403, 404, 405, 406, 407, 408, 409, 410, 411, 412, 413, 415, 416, 417, 418, 419, 420, 421,
422, 423, 424, 425, 426, 427, 428, 429, 430, 431, 432, 433, 434, 435, 436, 437, 438, 439, 440, 441, 442, 443, 44
4, 445, 446, 447, 448, 450, 451, 452, 453, 454, 455, 456, 457, 458, 459, 460, 461, 462, 463, 464, 465, 466, 467,
468, 469, 470, 471, 472, 473, 474, 475, 476, 477, 478, 479, 480, 481, 482, 483, 484, 485, 486, 487, 488, 489, 49
0, 491, 492, 493, 494, 495, 496, 497, 498, 499, 500, 501, 502, 503, 504, 505, 506, 507, 508, 509, 510, 511, 512,
513, 514, 515, 516, 517, 518, 519, 520, 521, 522, 523, 524, 525, 526, 527, 528, 529, 530, 531, 532, 533, 534, 53
5, 536, 537, 538, 539, 540, 541, 542, 543, 544, 545, 546, 547, 548, 549, 550, 551, 552, 553, 554, 555, 556, 557,
558, 559, 560, 561, 562, 563, 564, 565, 566, 567, 568, 569, 570, 571, 572, 573, 574, 575, 576, 577, 578, 579, 58
1, 582, 583, 584, 585, 586, 587, 588, 589, 590, 591, 592, 593, 594, 595, 596, 597, 598, 599, 600, 601, 602, 604,
606, 607, 608, 609, 610, 611, 612, 613, 614, 615, 616, 617, 618, 619, 620, 621, 622, 623, 624, 625, 626, 627, 62
8, 629, 630, 631, 632, 634, 635, 636, 637, 638, 639, 640, 641, 642, 643, 644, 645, 646, 647, 649, 650, 651, 652,
654, 655, 656, 657, 659, 661, 662, 663, 664, 665, 666, 667, 668, 669, 671, 672, 673, 674, 676, 677, 678, 679, 68
0, 681, 682, 683, 685, 686, 687, 688, 689, 690, 692, 693, 694, 695, 696, 697, 698, 699, 700, 701, 702, 704, 705,
706, 707, 708, 709, 710, 712, 713, 714, 715, 716, 717, 718, 719, 720, 721, 722, 723, 724, 725, 726, 727, 728, 72
9, 730, 733, 734, 735, 736, 737, 738, 739, 740, 741, 742, 743, 745, 747, 748, 749, 750, 751, 752, 753, 754, 755,
756, 757, 758, 759, 760, 761, 762, 763, 764, 765, 766, 767, 768, 769, 770, 771, 773, 774, 775, 776, 777, 778, 77
9, 780, 781, 782, 783, 784, 785, 786, 787, 788, 789, 790, 791, 792, 793, 794, 795, 796, 797, 799, 800, 801, 802,
803, 804, 805, 806, 807, 809, 810, 811, 812, 813, 814, 815, 816, 817, 818, 819, 820, 821, 822, 823, 824, 825, 82
6, 827, 828, 829, 830, 831, 832, 833, 834, 835, 836, 837, 838, 839, 840, 841, 842, 843, 844, 845, 847, 848, 849,
850, 851, 852, 853, 854, 855, 856, 857, 858, 859, 860, 861, 862, 863, 864, 865, 866, 867, 868, 869, 870, 871, 87
2, 873, 874, 875, 876, 877, 878, 879, 880, 881, 882, 883, 884, 885, 886, 887, 888, 889, 890, 891, 892, 893, 894,
895, 896, 897, 898, 899, 900, 901, 902, 903, 904, 905, 906, 907, 908, 909, 910, 911, 912, 913, 914, 915, 917, 91
8, 919, 920, 921, 922, 923, 924, 925, 926, 927, 928, 929, 930, 931, 932, 933, 934, 935, 936, 937, 938, 939, 940,
941, 942, 943, 944, 945, 946, 947, 948, 949, 950, 951, 952, 953, 954, 955, 956, 957, 958, 959, 960, 961, 962, 96
3, 964, 965, 966, 967, 968, 969, 970, 971, 972, 973, 974, 975, 976, 977, 978, 979, 980, 981, 982, 983, 984, 985,
986, 987, 988, 989, 990, 991, 992, 993, 994, 995, 996, 997, 998, 999, 1000, 1001, 1002, 1003, 1004], [414, 449,
603, 605, 846, 916], [580], [633], [648], [653], [658], [660], [670], [675], [684], [691], [703], [711], [731],
[732], [744], [746], [772], [798], [808]]
```

3. Applying the Louvain Algorithm

In [21]:
```python
import networkx as nx
import community as community_louvain
import igraph as ig
import leidenalg as la
from networkx.algorithms.community import girvan_newman
```

In [22]:
```python
import community as community_louvain

# Apply Louvain algorithm
partition_louvain = community_louvain.best_partition(graph)
print("Louvain communities:", partition_louvain)
```

```
Louvain communities: {0: 14, 1: 14, 2: 1, 3: 1, 4: 1, 5: 1, 6: 1, 7: 2, 8: 2, 9: 2, 10: 11, 11: 2, 12: 2, 13: 4,
14: 7, 15: 12, 16: 11, 17: 14, 18: 14, 19: 2, 20: 11, 21: 11, 22: 11, 23: 4, 24: 4, 25: 4, 26: 4, 27: 4, 28: 4,
29: 4, 30: 4, 31: 4, 32: 4, 33: 4, 34: 4, 35: 4, 36: 4, 37: 4, 38: 4, 39: 4, 40: 4, 41: 7, 42: 11, 43: 2, 44: 2,
45: 12, 46: 12, 47: 4, 48: 4, 49: 11, 50: 11, 51: 7, 52: 14, 53: 7, 54: 1, 55: 1, 56: 1, 57: 1, 58: 1, 59: 1, 60
: 14, 61: 14, 62: 11, 63: 1, 64: 7, 65: 7, 66: 11, 67: 11, 68: 11, 69: 11, 70: 11, 71: 11, 72: 11, 73: 14, 74: 1
4, 75: 4, 76: 4, 77: 11, 78: 11, 79: 11, 80: 11, 81: 11, 82: 11, 83: 11, 84: 11, 85: 14, 86: 11, 87: 11, 88: 1,
89: 1, 90: 11, 91: 11, 92: 11, 93: 4, 94: 7, 95: 7, 96: 4, 97: 12, 98: 12, 99: 12, 100: 12, 101: 12, 102: 1, 103
: 14, 104: 14, 105: 11, 106: 11, 107: 11, 108: 11, 109: 11, 110: 11, 111: 11, 112: 11, 113: 4, 114: 4, 115: 4, 1
16: 4, 117: 11, 118: 11, 119: 4, 120: 14, 121: 11, 122: 7, 123: 4, 124: 12, 125: 12, 126: 1, 127: 11, 128: 7, 12
9: 7, 130: 7, 131: 1, 132: 1, 133: 4, 134: 11, 135: 4, 136: 4, 137: 1, 138: 1, 139: 12, 140: 12, 141: 2, 142: 11
, 143: 11, 144: 11, 145: 11, 146: 14, 147: 11, 148: 7, 149: 7, 150: 14, 151: 4, 152: 11, 153: 11, 154: 11, 155:
11, 156: 14, 157: 14, 158: 1, 159: 1, 160: 14, 161: 2, 162: 11, 163: 11, 164: 12, 165: 4, 166: 14, 167: 7, 168:
7, 169: 4, 170: 4, 171: 4, 172: 7, 173: 11, 174: 1, 175: 1, 176: 7, 177: 14, 178: 7, 179: 14, 180: 7, 181: 14, 1
82: 14, 183: 7, 184: 11, 185: 12, 186: 11, 187: 11, 188: 11, 189: 11, 190: 11, 191: 7, 192: 1, 193: 1, 194: 1, 1
95: 1, 196: 7, 197: 7, 198: 7, 199: 7, 200: 7, 201: 7, 202: 7, 203: 7, 204: 7, 205: 7, 206: 7, 207: 7, 208: 1, 2
09: 1, 210: 1, 211: 1, 212: 11, 213: 2, 214: 7, 215: 14, 216: 12, 217: 11, 218: 14, 219: 14, 220: 14, 221: 14, 2
22: 14, 223: 14, 224: 14, 225: 14, 226: 14, 227: 14, 228: 14, 229: 4, 230: 11, 231: 7, 232: 7, 233: 1, 234: 1, 2
35: 1, 236: 1, 237: 1, 238: 1, 239: 1, 240: 1, 241: 1, 242: 1, 243: 1, 244: 1, 245: 4, 246: 2, 247: 2, 248: 14,
249: 11, 250: 14, 251: 4, 252: 1, 253: 11, 254: 11, 255: 11, 256: 11, 257: 7, 258: 11, 259: 11, 260: 11, 261: 4,
262: 7, 263: 4, 264: 2, 265: 2, 266: 2, 267: 2, 268: 14, 269: 12, 270: 7, 271: 1, 272: 12, 273: 12, 274: 12, 275
: 7, 276: 7, 277: 7, 278: 7, 279: 11, 280: 7, 281: 1, 282: 11, 283: 11, 284: 7, 285: 1, 286: 1, 287: 11, 288: 11
, 289: 7, 290: 7, 291: 7, 292: 7, 293: 2, 294: 7, 295: 14, 296: 14, 297: 14, 298: 11, 299: 11, 300: 11, 301: 14,
302: 1, 303: 1, 304: 1, 305: 1, 306: 11, 307: 14, 308: 14, 309: 14, 310: 14, 311: 14, 312: 14, 313: 14, 314: 14,
315: 14, 316: 14, 317: 14, 318: 4, 319: 1, 320: 14, 321: 14, 322: 12, 323: 12, 324: 2, 325: 11, 326: 11, 327: 11
, 328: 11, 329: 11, 330: 14, 331: 14, 332: 2, 333: 4, 334: 12, 335: 12, 336: 4, 337: 4, 338: 4, 339: 4, 340: 11,
341: 14, 342: 7, 343: 7, 344: 7, 345: 7, 346: 7, 347: 7, 348: 7, 349: 7, 350: 7, 351: 7, 352: 7, 353: 12, 354: 1
2, 355: 11, 356: 11, 357: 11, 358: 2, 359: 2, 360: 2, 361: 4, 362: 2, 363: 11, 364: 11, 365: 2, 366: 11, 367: 4,
368: 14, 369: 1, 370: 4, 371: 12, 372: 11, 373: 1, 374: 2, 375: 11, 376: 14, 377: 14, 378: 14, 379: 14, 380: 14,
381: 14, 382: 14, 383: 14, 384: 14, 385: 14, 386: 14, 387: 14, 388: 14, 389: 14, 390: 14, 391: 14, 392: 14, 393:
14, 394: 14, 395: 14, 396: 14, 397: 14, 398: 14, 399: 7, 400: 11, 401: 7, 402: 7, 403: 7, 404: 12, 405: 11, 406:
2, 407: 2, 408: 1, 409: 4, 410: 11, 411: 1, 412: 1, 413: 7, 414: 14, 415: 14, 416: 11, 417: 4, 418: 11, 419: 11,
420: 11, 421: 2, 422: 11, 423: 4, 424: 11, 425: 7, 426: 7, 427: 4, 428: 12, 429: 12, 430: 2, 431: 11, 432: 11, 4
33: 11, 434: 11, 435: 14, 436: 4, 437: 4, 438: 4, 439: 4, 440: 7, 441: 2, 442: 4, 443: 4, 444: 4, 445: 11, 446:
12, 447: 12, 448: 12, 449: 14, 450: 7, 451: 2, 452: 2, 453: 11, 454: 11, 455: 4, 456: 7, 457: 7, 458: 7, 459: 14
, 460: 11, 461: 12, 462: 11, 463: 11, 464: 4, 465: 11, 466: 2, 467: 11, 468: 14, 469: 11, 470: 4, 471: 14, 472:
14, 473: 11, 474: 11, 475: 11, 476: 11, 477: 11, 478: 11, 479: 14, 480: 11, 481: 1, 482: 11, 483: 7, 484: 7, 485
: 4, 486: 7, 487: 2, 488: 2, 489: 11, 490: 11, 491: 11, 492: 11, 493: 7, 494: 4, 495: 11, 496: 2, 497: 11, 498:
2, 499: 2, 500: 2, 501: 2, 502: 2, 503: 2, 504: 2, 505: 2, 506: 2, 507: 14, 508: 11, 509: 11, 510: 2, 511: 7, 51
2: 11, 513: 11, 514: 11, 515: 7, 516: 1, 517: 1, 518: 11, 519: 14, 520: 1, 521: 11, 522: 7, 523: 7, 524: 11, 525
: 2, 526: 7, 527: 4, 528: 1, 529: 2, 530: 2, 531: 11, 532: 1, 533: 11, 534: 7, 535: 11, 536: 11, 537: 14, 538: 1
1, 539: 11, 540: 11, 541: 11, 542: 7, 543: 7, 544: 7, 545: 4, 546: 11, 547: 4, 548: 11, 549: 11, 550: 11, 551: 1
1, 552: 1, 553: 11, 554: 11, 555: 2, 556: 7, 557: 7, 558: 2, 559: 11, 560: 14, 561: 11, 562: 7, 563: 7, 564: 1,
565: 2, 566: 2, 567: 7, 568: 7, 569: 2, 570: 2, 571: 1, 572: 7, 573: 2, 574: 7, 575: 7, 576: 7, 577: 11, 578: 11
, 579: 12, 580: 16, 581: 7, 582: 11, 583: 11, 584: 7, 585: 7, 586: 1, 587: 1, 588: 4, 589: 11, 590: 4, 591: 11,
592: 12, 593: 7, 594: 11, 595: 14, 596: 14, 597: 11, 598: 11, 599: 1, 600: 7, 601: 1, 602: 2, 603: 14, 604: 1, 6
05: 7, 606: 11, 607: 11, 608: 2, 609: 4, 610: 1, 611: 7, 612: 11, 613: 14, 614: 14, 615: 11, 616: 14, 617: 12, 6
18: 12, 619: 1, 620: 11, 621: 7, 622: 1, 623: 7, 624: 11, 625: 1, 626: 11, 627: 11, 628: 14, 629: 14, 630: 1, 63
1: 1, 632: 14, 633: 17, 634: 1, 635: 1, 636: 1, 637: 1, 638: 11, 639: 1, 640: 12, 641: 11, 642: 11, 643: 11, 644
: 12, 645: 14, 646: 1, 647: 11, 648: 18, 649: 2, 650: 14, 651: 11, 652: 11, 653: 19, 654: 11, 655: 4, 656: 7, 65
7: 12, 658: 20, 659: 14, 660: 21, 661: 2, 662: 12, 663: 11, 664: 7, 665: 1, 666: 2, 667: 11, 668: 14, 669: 11, 6
70: 22, 671: 11, 672: 2, 673: 11, 674: 2, 675: 23, 676: 12, 677: 7, 678: 11, 679: 11, 680: 14, 681: 7, 682: 7, 6
83: 1, 684: 24, 685: 1, 686: 4, 687: 12, 688: 7, 689: 7, 690: 14, 691: 25, 692: 7, 693: 11, 694: 7, 695: 14, 696
: 14, 697: 14, 698: 1, 699: 2, 700: 2, 701: 11, 702: 11, 703: 8, 704: 11, 705: 7, 706: 7, 707: 2, 708: 12, 709:
12, 710: 11, 711: 9, 712: 1, 713: 11, 714: 7, 715: 11, 716: 1, 717: 1, 718: 1, 719: 14, 720: 2, 721: 4, 722: 4,
723: 11, 724: 14, 725: 4, 726: 11, 727: 11, 728: 11, 729: 2, 730: 14, 731: 10, 732: 13, 733: 11, 734: 14, 735: 1
2, 736: 11, 737: 1, 738: 1, 739: 11, 740: 7, 741: 14, 742: 14, 743: 1, 744: 15, 745: 14, 746: 0, 747: 11, 748: 1
1, 749: 7, 750: 1, 751: 7, 752: 11, 753: 4, 754: 2, 755: 1, 756: 14, 757: 4, 758: 14, 759: 14, 760: 12, 761: 14,
762: 1, 763: 1, 764: 14, 765: 2, 766: 11, 767: 4, 768: 12, 769: 11, 770: 12, 771: 11, 772: 3, 773: 11, 774: 1, 7
75: 14, 776: 7, 777: 11, 778: 2, 779: 7, 780: 11, 781: 14, 782: 14, 783: 11, 784: 1, 785: 4, 786: 7, 787: 11, 78
8: 1, 789: 7, 790: 7, 791: 7, 792: 11, 793: 14, 794: 12, 795: 4, 796: 11, 797: 7, 798: 5, 799: 11, 800: 11, 801:
7, 802: 7, 803: 1, 804: 2, 805: 2, 806: 1, 807: 1, 808: 6, 809: 1, 810: 14, 811: 11, 812: 1, 813: 7, 814: 7, 815
: 1, 816: 4, 817: 7, 818: 11, 819: 12, 820: 7, 821: 14, 822: 7, 823: 2, 824: 7, 825: 11, 826: 1, 827: 2, 828: 11
, 829: 7, 830: 2, 831: 11, 832: 1, 833: 2, 834: 11, 835: 7, 836: 7, 837: 11, 838: 12, 839: 14, 840: 7, 841: 14,
842: 1, 843: 7, 844: 4, 845: 1, 846: 7, 847: 4, 848: 11, 849: 1, 850: 7, 851: 7, 852: 14, 853: 11, 854: 1, 855:
11, 856: 2, 857: 14, 858: 11, 859: 1, 860: 4, 861: 7, 862: 11, 863: 1, 864: 1, 865: 1, 866: 11, 867: 11, 868: 14
, 869: 14, 870: 7, 871: 7, 872: 14, 873: 7, 874: 7, 875: 4, 876: 1, 877: 11, 878: 4, 879: 1, 880: 1, 881: 4, 882
: 14, 883: 4, 884: 1, 885: 11, 886: 1, 887: 11, 888: 1, 889: 11, 890: 11, 891: 11, 892: 11, 893: 2, 894: 11, 895
: 4, 896: 11, 897: 4, 898: 1, 899: 1, 900: 4, 901: 1, 902: 1, 903: 7, 904: 14, 905: 14, 906: 14, 907: 7, 908: 7,
909: 7, 910: 7, 911: 11, 912: 2, 913: 2, 914: 4, 915: 4, 916: 14, 917: 7, 918: 7, 919: 4, 920: 14, 921: 1, 922:
2, 923: 11, 924: 1, 925: 4, 926: 1, 927: 1, 928: 1, 929: 12, 930: 1, 931: 1, 932: 11, 933: 12, 934: 11, 935: 12,
936: 7, 937: 12, 938: 7, 939: 7, 940: 7, 941: 11, 942: 7, 943: 14, 944: 11, 945: 11, 946: 11, 947: 14, 948: 14,
949: 1, 950: 2, 951: 2, 952: 11, 953: 4, 954: 11, 955: 7, 956: 2, 957: 2, 958: 11, 959: 14, 960: 14, 961: 14, 96
2: 11, 963: 1, 964: 4, 965: 7, 966: 11, 967: 2, 968: 11, 969: 11, 970: 7, 971: 2, 972: 11, 973: 2, 974: 7, 975:
2, 976: 4, 977: 1, 978: 4, 979: 14, 980: 4, 981: 4, 982: 1, 983: 4, 984: 11, 985: 14, 986: 4, 987: 4, 988: 1, 98
9: 11, 990: 1, 991: 1, 992: 7, 993: 1, 994: 1, 995: 1, 996: 2, 997: 4, 998: 7, 999: 11, 1000: 7, 1001: 1, 1002:
14, 1003: 11, 1004: 1}
```

In [ ]:

4. Applying the Leiden Algorithm

In [23]:

```
import igraph as ig
import leidenalg as la

# Convert NetworkX graph to igraph
ig_graph = ig.Graph.TupleList(graph.edges(), directed=False)

# Apply Leiden algorithm
partition = la.find_partition(ig_graph, la.ModularityVertexPartition)
print("Leiden communities:", partition)
```

```
Leiden communities: Clustering with 1005 elements and 27 clusters
[ 0] 495, 549, 147, 106, 21, 82, 254, 155, 189, 548, 255, 187, 127, 121, 84,
     641, 142, 434, 162, 300, 489, 612, 249, 107, 62, 546, 651, 896, 405, 256,
     81, 932, 473, 282, 424, 431, 145, 329, 117, 327, 105, 357, 279, 669, 42,
     306, 492, 134, 419, 828, 469, 375, 184, 154, 420, 462, 325, 728, 163,
     690, 230, 667, 418, 299, 112, 422, 212, 10, 340, 260, 638, 535, 69, 173,
     143, 777, 366, 400, 508, 326, 509, 20, 91, 92, 22, 90, 518, 739, 118,
     550, 624, 87, 83, 476, 478, 78, 679, 497, 771, 190, 514, 356, 108, 767,
     474, 748, 465, 16, 541, 80, 355, 153, 363, 68, 258, 642, 783, 654, 615,
     467, 253, 643, 671, 513, 288, 49, 259, 188, 72, 287, 715, 536, 480, 885,
     887, 364, 787, 647, 551, 966, 152, 460, 811, 952, 877, 554, 186, 372,
     453, 298, 490, 607, 50, 66, 678, 71, 433, 217, 109, 597, 70, 582, 531,
     77, 410, 454, 769, 591, 521, 818, 747, 627, 733, 752, 67, 328, 477, 984,
     111, 663, 713, 710, 958, 432, 524, 911, 540, 853, 693, 594, 589, 110,
     780, 144, 598, 553, 471, 855, 626, 954, 989, 538, 723, 867, 702, 704,
     834, 837, 652, 491, 792, 796, 890, 766, 559, 577, 999, 512, 799, 934,
     416, 968, 969, 800, 941, 472, 831, 1003, 889, 475, 825, 858, 962, 945,
     923, 894, 848, 773, 463, 561, 701, 606, 673, 583, 578, 946, 944, 862
[ 1] 88, 5, 238, 6, 2, 3, 4, 192, 281, 195, 305, 412, 102, 194, 55, 252, 63,
     586, 408, 138, 58, 193, 174, 285, 137, 211, 520, 635, 481, 665, 59, 208,
     516, 517, 286, 57, 132, 411, 587, 809, 826, 158, 738, 832, 803, 845, 571,
     89, 54, 599, 564, 865, 880, 698, 899, 886, 812, 859, 131, 56, 126, 271,
     990, 685, 1001, 175, 303, 236, 239, 601, 210, 532, 625, 552, 528, 644,
     807, 815, 302, 243, 209, 718, 902, 237, 716, 234, 242, 931, 235, 763,
     619, 949, 921, 994, 950, 159, 233, 240, 304, 244, 1004, 610, 849, 604,
     863, 683, 924, 373, 963, 854, 926, 639, 750, 369, 806, 743, 319, 646,
     901, 982, 241, 927, 930, 631, 637, 977, 755, 636, 928, 898, 622, 879,
     737, 774, 630, 993, 784, 864, 884, 888, 991, 762, 876, 634
[ 2] 581, 64, 65, 568, 199, 128, 280, 450, 232, 86, 183, 712, 425, 53, 820,
     41, 458, 201, 440, 198, 291, 464, 486, 840, 207, 611, 572, 133, 557, 445,
     789, 751, 206, 515, 290, 544, 426, 79, 204, 563, 275, 562, 95, 168, 270,
     129, 14, 576, 94, 197, 413, 483, 526, 93, 172, 51, 484, 167, 403, 493,
     289, 203, 196, 292, 523, 706, 482, 205, 593, 776, 918, 556, 936, 1000,
     543, 917, 200, 202, 801, 908, 401, 399, 970, 822, 664, 749, 457, 714,
     456, 257, 294, 874, 600, 910, 870, 835, 574, 791, 998, 176, 567, 727,
     585, 623, 584, 620, 974, 909, 813, 940, 955, 542, 802, 694, 992, 276,
     829, 705, 689, 677, 850, 939, 534, 843, 836, 873, 817, 522, 688, 790,
     938, 995, 965
[ 3] 734, 178, 380, 250, 148, 377, 103, 368, 283, 284, 52, 351, 160, 308, 157,
     379, 191, 349, 730, 697, 346, 681, 511, 868, 376, 397, 181, 296, 149,
     389, 621, 352, 179, 214, 130, 278, 156, 343, 342, 390, 350, 61, 394, 277,
     231, 797, 395, 295, 393, 386, 378, 180, 104, 852, 182, 321, 387, 595,
     814, 781, 719, 60, 150, 122, 381, 320, 388, 392, 384, 385, 391, 396, 947,
     345, 347, 262, 736, 468, 539, 782, 415, 810, 741, 656, 692, 402, 344,
     575, 645, 907, 906, 668, 382, 383, 398, 851, 724, 821, 479, 869, 348,
     628, 682, 824, 839, 903, 871, 841, 943, 904, 861, 414, 632, 985, 659,
     959, 960, 961, 680, 948, 761, 449, 603, 605, 916, 846
[ 4] 13, 114, 409, 96, 494, 333, 170, 169, 115, 527, 437, 76, 35, 455, 171,
     337, 29, 47, 423, 438, 417, 24, 261, 980, 318, 135, 34, 900, 339, 36, 25,
     27, 444, 165, 23, 443, 590, 113, 336, 38, 30, 33, 32, 26, 263, 39, 785,
     655, 361, 588, 40, 367, 338, 427, 28, 686, 31, 37, 251, 116, 878, 545,
     229, 953, 964, 151, 609, 988, 997, 370, 442, 136, 721, 722, 48, 883, 892,
     914, 123, 75, 245, 547, 847, 915, 119, 753, 725, 919, 816, 976, 981, 925,
     485, 470, 866, 439, 881, 842, 983, 895, 436, 860, 978, 987, 875, 986,
     897, 757, 891, 795
[ 5] 498, 266, 506, 971, 533, 360, 430, 7, 141, 374, 496, 362, 833, 466, 11,
     44, 19, 569, 161, 666, 661, 12, 566, 452, 421, 510, 504, 359, 499, 707,
     700, 332, 406, 502, 365, 358, 451, 407, 213, 525, 8, 672, 573, 570, 608,
     555, 264, 265, 823, 247, 856, 500, 246, 674, 754, 529, 324, 913, 720,
     956, 922, 765, 9, 441, 293, 565, 487, 558, 699, 957, 530, 804, 602, 267,
     996, 43, 505, 501, 805, 503, 729, 951, 967, 649, 740, 778, 844, 488, 830,
     893, 912, 827, 972, 973, 975
[ 6] 0, 1, 17, 316, 146, 268, 221, 218, 18, 459, 215, 73, 74, 248, 226, 177,
     560, 309, 297, 313, 223, 222, 166, 120, 310, 225, 85, 726, 224, 537, 616,
     979, 317, 301, 312, 311, 307, 220, 314, 315, 228, 341, 696, 872, 219,
     331, 695, 519, 764, 756, 786, 758, 330, 435, 227, 745, 905, 507, 942,
     614, 717, 613, 759, 779, 882, 857, 742, 629, 793, 650, 596, 920, 775,
     1002, 788
[ 7] 101, 323, 140, 273, 164, 272, 371, 46, 125, 269, 100, 97, 99, 139, 429,
     446, 461, 322, 579, 98, 15, 216, 335, 334, 45, 274, 448, 592, 760, 353,
     428, 447, 404, 933, 929, 124, 838, 640, 185, 662, 708, 354, 937, 687,
     768, 618, 935, 819, 617, 735, 676, 709, 657, 794, 770
[ 8] 580
```

```
[ 9] 633
[10] 648
[11] 653
[12] 658
[13] 660
[14] 670
[15] 675
[16] 684
[17] 691
[18] 703
[19] 711
[20] 731
[21] 732
[22] 744
[23] 746
[24] 772
[25] 798
[26] 808
```

Overlapping Algorithm

Clique Percolation

In [12]:

```python
import networkx as nx

# Function to read a subset of the edges from the file
def read_subset_of_edges(file_path, num_edges):
    edges = []
    with open(file_path, 'r') as f:
        for _ in range(num_edges):
            line = f.readline().strip()
            if not line:
                break
            edges.append(tuple(map(int, line.split())))
    return edges

# Path to the edge list file
file_path = 'email-Eu-core.txt'

# Number of edges to read
num_edges = 2000  # Adjust this number to control the subset size

# Read a subset of the edges
edges = read_subset_of_edges(file_path, num_edges)

# Create a graph from the subset of edges
graph = nx.Graph()
graph.add_edges_from(edges)

# Apply Clique Percolation algorithm with k-clique size, k
k = 4
communities = list(nx.algorithms.community.k_clique_communities(graph, k))

# Convert communities to list of lists for better readability
communities = [list(c) for c in communities]

# Display the number of communities found
print(f"Number of communities found: {len(communities)}")

# Display the first 100 communities
for i, community in enumerate(communities[:100], 1):  # Adjust the number of communities to display
    print(f"Community {i}: {community}")
```

```
Number of communities found: 30
Community 1: [192, 194, 2, 195]
Community 2: [2, 3, 4, 174, 63]
Community 3: [73, 74, 17, 177, 341, 309, 311, 248, 215, 220, 222, 223]
Community 4: [23, 25, 26, 27, 28, 29, 30, 33, 34, 35, 36, 37, 38, 39, 40, 438]
Community 5: [165, 169, 367, 114, 28]
Community 6: [407, 43, 44, 499]
Community 7: [448, 164, 139, 140, 429, 428, 46, 447]
Community 8: [256, 318, 263, 81, 283, 417, 423, 361, 106, 107, 365, 47, 113, 434, 114, 245, 249, 443, 444, 62]
Community 9: [49, 50, 84, 71]
Community 10: [280, 51, 133, 167]
Community 11: [128, 196, 197, 199, 167, 41, 201, 203, 207, 51, 94, 95]
Community 12: [54, 55, 56, 57, 58, 59]
Community 13: [320, 321, 385, 393, 394, 150, 295, 104, 103, 296, 378, 368, 61, 181, 376, 377, 250, 60, 381]
Community 14: [258, 478, 329, 375, 474, 62]
Community 15: [282, 226, 62, 254]
Community 16: [115, 107, 62, 254]
Community 17: [329, 66, 108, 375]
Community 18: [288, 162, 70, 71, 84, 217]
Community 19: [248, 74, 259, 260]
Community 20: [128, 41, 205, 86]
Community 21: [96, 302, 115, 89, 252]
Community 22: [328, 329, 90, 375]
Community 23: [105, 145, 212, 121, 252]
Community 24: [165, 135, 136, 169, 336, 337, 338, 115, 116, 251]
Community 25: [129, 434, 490, 365]
Community 26: [233, 234, 235, 236, 238, 240, 209, 210, 243, 244, 159]
Community 27: [169, 170, 115, 438]
Community 28: [176, 177, 178, 344, 381]
Community 29: [329, 356, 357, 375]
Community 30: [496, 529, 466, 421]
```

Label Propagation Paper Link: Finding overlapping communities in networks by label

In [13]:

```python
import networkx as nx
import random

# Function to read a subset of the edges from the file
def read_subset_of_edges(file_path, num_edges):
    edges = []
    with open(file_path, 'r') as f:
        for _ in range(num_edges):
            line = f.readline().strip()
            if not line:
                break
            edges.append(tuple(map(int, line.split())))
    return edges

# Path to the edge list file
file_path = 'email-Eu-core.txt'

# Number of edges to read
num_edges = 2000  # Adjust this number to control the subset size

# Read a subset of the edges
edges = read_subset_of_edges(file_path, num_edges)

# Create a graph from the subset of edges
graph = nx.Graph()
graph.add_edges_from(edges)

# Implement the Label Propagation Algorithm
def label_propagation(graph):
    # Initialize each node with a unique label
    labels = {node: node for node in graph.nodes()}
    nodes = list(graph.nodes())
    random.shuffle(nodes)

    while True:
        updated = False
        # For each node, update its label based on the most frequent label of its neighbors
        for node in nodes:
            if graph.degree(node) == 0:
                continue
            neighbor_labels = [labels[neighbor] for neighbor in graph.neighbors(node)]
            most_frequent_label = max(set(neighbor_labels), key=neighbor_labels.count)
            if labels[node] != most_frequent_label:
                labels[node] = most_frequent_label
                updated = True
```

```
        # If no labels were updated, the algorithm has converged
        if not updated:
            break

    # Group nodes by labels
    communities = {}
    for node, label in labels.items():
        if label not in communities:
            communities[label] = []
        communities[label].append(node)

    return list(communities.values())

# Detect communities using Label Propagation
communities = label_propagation(graph)

# Display the number of communities found
print(f"Number of communities found: {len(communities)}")

# Display the first few communities
for i, community in enumerate(communities[:10], 1):  # Adjust the number of communities to display
    print(f"Community {i}: {community}")
```

```
Number of communities found: 36
Community 1: [0, 1, 17, 73, 74, 120, 215, 218, 219, 220, 221, 222, 223, 224, 225, 227, 228, 248, 259, 260, 297,
309, 311, 312, 313, 315, 316, 317, 330, 331, 341]
Community 2: [2, 3, 4, 5, 6, 7, 10, 11, 12, 13, 14, 16, 18, 19, 20, 21, 22, 24, 28, 31, 35, 36, 41, 42, 44, 47,
48, 51, 62, 63, 64, 65, 68, 69, 75, 76, 77, 78, 79, 80, 81, 82, 85, 86, 87, 88, 89, 92, 93, 94, 95, 96, 97, 98,
99, 100, 101, 102, 105, 106, 107, 113, 114, 115, 116, 117, 118, 119, 121, 122, 123, 126, 127, 128, 129, 130, 133
, 134, 135, 136, 137, 138, 141, 142, 144, 145, 146, 147, 151, 152, 153, 154, 155, 160, 161, 165, 166, 167, 168,
169, 170, 171, 172, 173, 174, 175, 179, 180, 183, 186, 187, 188, 189, 190, 191, 196, 197, 198, 199, 200, 201, 20
2, 203, 204, 205, 206, 207, 212, 214, 226, 229, 230, 231, 232, 245, 249, 251, 252, 253, 254, 255, 256, 257, 261,
262, 263, 268, 270, 275, 276, 278, 279, 280, 281, 282, 283, 284, 286, 287, 289, 290, 291, 292, 293, 294, 300, 30
1, 302, 303, 304, 306, 307, 308, 310, 314, 318, 319, 323, 324, 325, 326, 327, 332, 333, 336, 337, 338, 339, 340,
342, 343, 345, 346, 347, 348, 349, 350, 351, 352, 355, 361, 362, 363, 364, 365, 366, 367, 369, 370, 371, 374, 39
9, 400, 401, 402, 403, 404, 405, 408, 412, 413, 416, 417, 419, 422, 423, 424, 427, 430, 434, 435, 436, 437, 438,
439, 440, 441, 442, 443, 444, 450, 453, 454, 455, 457, 458, 459, 460, 464, 465, 467, 470, 471, 472, 473, 480, 48
2, 483, 484, 485, 486, 490, 491, 493, 495, 497, 500, 504, 506, 510, 511, 512, 513, 515, 518, 520, 521, 527, 531,
532, 533, 534, 535, 536, 537, 538, 539, 540, 542, 543, 544, 546, 547, 548, 551, 553, 554, 555, 556, 557, 559, 56
0, 563, 565, 567, 568, 571, 572, 574, 576]
Community 3: [8, 9, 359, 360, 421, 451, 452, 466, 496, 514, 529, 558, 566, 569, 570, 573]
Community 4: [15, 45, 46, 139, 140, 164, 216, 269, 271, 272, 273, 274, 322, 334, 335, 428, 429, 445, 446, 447, 4
48, 461, 579]
Community 5: [23, 25, 26, 27, 29, 30, 32, 33, 34, 37, 38, 39, 40, 420, 545]
Community 6: [43, 358, 406, 407, 499, 501, 502, 503, 505, 525]
Community 7: [49, 50, 70, 71, 83, 84, 162, 217, 288, 372, 373, 431, 432, 433, 492, 494]
Community 8: [52, 53]
Community 9: [54, 55, 56, 57, 58, 59, 208, 211, 285, 305, 481, 552]
Community 10: [60, 61, 103, 104, 150, 156, 157, 181, 182, 213, 250, 295, 296, 320, 321, 368, 376, 377, 378, 379,
380, 382, 383, 384, 385, 386, 387, 388, 389, 390, 391, 392, 393, 394, 395, 396, 397, 398, 526, 541]
```

Propagation

In [14]:

```
import networkx as nx
import random

# Function to read a subset of the edges from the file
def read_subset_of_edges(file_path, num_edges):
    edges = []
    with open(file_path, 'r') as f:
        for _ in range(num_edges):
            line = f.readline().strip()
            if not line:
                break
            edges.append(tuple(map(int, line.split())))
    return edges

# Path to the edge list file
file_path = 'email-Eu-core.txt'

# Number of edges to read
num_edges = 2000  # Adjust this number to control the subset size

# Read a subset of the edges
edges = read_subset_of_edges(file_path, num_edges)

# Create a graph from the subset of edges
graph = nx.Graph()
graph.add_edges_from(edges)

# Label Propagation Algorithm implementation
def label_propagation(graph):
```

```
    # Initialize each node with a unique label
    labels = {node: node for node in graph.nodes()}
    nodes = list(graph.nodes())
    random.shuffle(nodes)

    while True:
        updated = False
        # For each node, update its label based on the most frequent label of its neighbors
        for node in nodes:
            if graph.degree(node) == 0:
                continue
            neighbor_labels = [labels[neighbor] for neighbor in graph.neighbors(node)]
            most_frequent_label = max(set(neighbor_labels), key=neighbor_labels.count)
            if labels[node] != most_frequent_label:
                labels[node] = most_frequent_label
                updated = True

        # If no labels were updated, the algorithm has converged
        if not updated:
            break

    # Group nodes by labels
    communities = {}
    for node, label in labels.items():
        if label not in communities:
            communities[label] = []
        communities[label].append(node)

    return list(communities.values())

# Detect communities using Label Propagation
communities = label_propagation(graph)

# Display the number of communities found
print(f"Number of communities found: {len(communities)}")

# Display the first few communities
for i, community in enumerate(communities[:10], 1):  # Adjust the number of communities to display
    print(f"Community {i}: {community}")
```

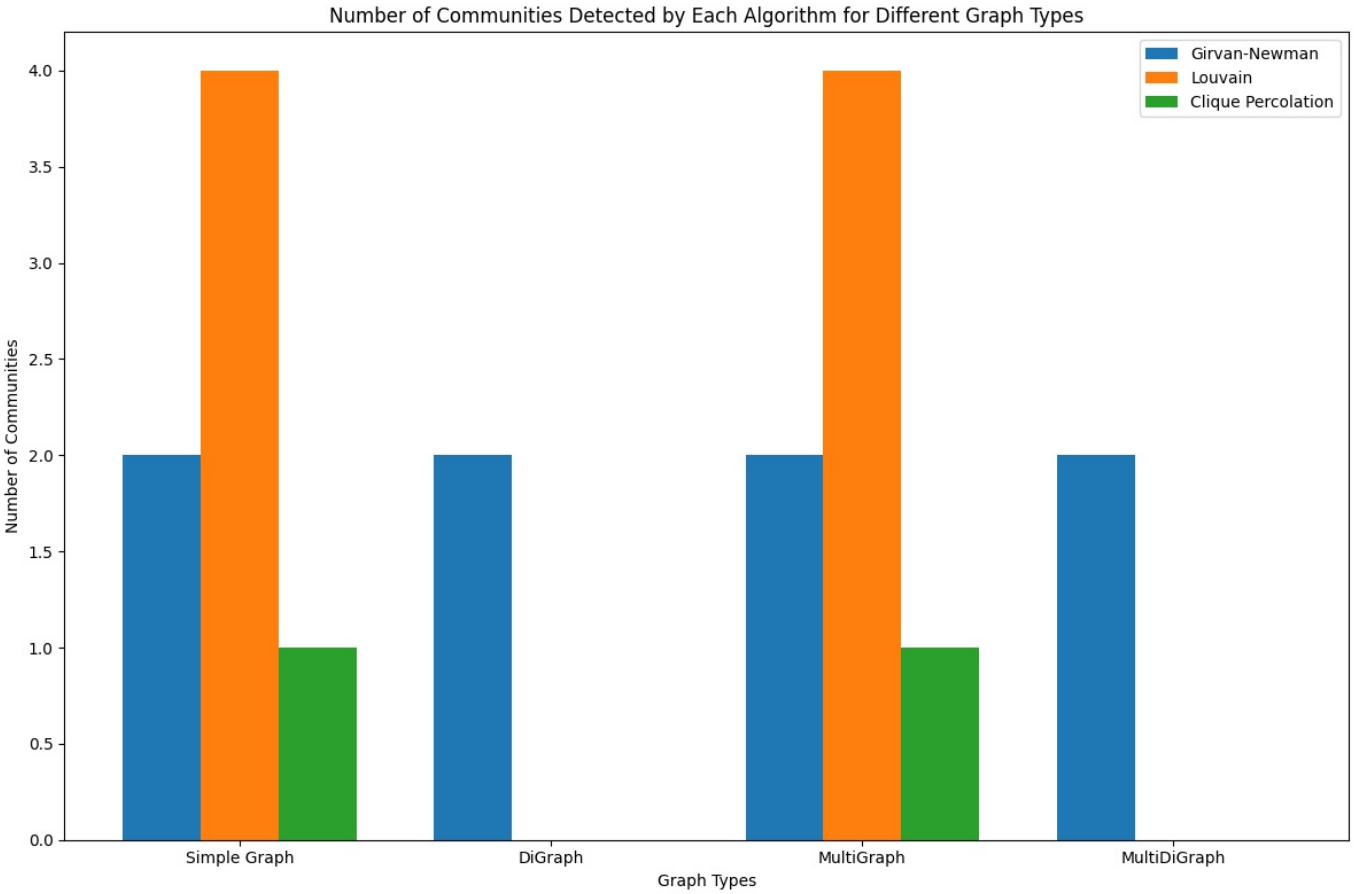
```
Number of communities found: 52
Community 1: [0, 1, 17, 18, 74, 120, 215, 218, 219, 220, 221, 222, 223, 224, 225, 227, 228, 309, 311, 312, 313,
315, 316, 317, 341]
Community 2: [2, 3, 4, 63, 137, 138, 174, 175, 192, 194, 195, 208, 211, 281, 285, 286, 305, 371, 412, 552, 571]
Community 3: [5, 6, 7, 12, 14, 19, 42, 43, 44, 47, 48, 62, 64, 65, 75, 76, 77, 78, 79, 81, 82, 85, 87, 105, 106,
107, 113, 114, 117, 118, 119, 121, 122, 127, 129, 130, 141, 142, 144, 145, 146, 147, 151, 152, 153, 154, 155, 15
6, 157, 160, 161, 173, 179, 180, 183, 186, 187, 188, 189, 190, 191, 198, 212, 214, 226, 229, 231, 245, 249, 253,
254, 255, 256, 257, 261, 262, 263, 268, 270, 278, 279, 280, 282, 283, 284, 289, 291, 292, 293, 294, 300, 306, 30
8, 310, 314, 318, 319, 327, 332, 333, 342, 343, 345, 346, 347, 348, 349, 350, 351, 352, 355, 358, 361, 362, 363,
364, 365, 366, 367, 374, 379, 396, 405, 406, 407, 416, 417, 422, 423, 427, 430, 434, 435, 441, 442, 443, 444, 45
4, 455, 457, 459, 460, 464, 465, 467, 471, 472, 473, 480, 483, 484, 485, 486, 490, 493, 495, 497, 499, 500, 501,
502, 503, 504, 505, 506, 511, 512, 513, 518, 525, 527, 533, 539, 546, 547, 551, 559, 560, 565, 572, 574, 576]
Community 4: [8, 9, 421, 466, 496, 529]
Community 5: [10, 11, 13, 126, 301, 324, 535, 555]
Community 6: [15, 16, 45, 46, 139, 140, 164, 216, 269, 271, 272, 273, 274, 322, 334, 335, 424, 428, 429, 445, 44
6, 447, 448, 461, 579]
Community 7: [20, 21, 22, 92, 97, 98, 99, 100, 101, 102, 230, 323, 325, 326, 520, 538]
Community 8: [23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 171, 370, 420, 545]
Community 9: [41, 51, 94, 95, 128, 167, 196, 197, 199, 200, 201, 202, 203, 204, 205, 206, 207]
Community 10: [49, 50, 70, 71, 83, 84, 162, 217, 288, 372, 373, 431, 432, 433, 492, 494]
```

Bar Chart

In [5]:

```
import matplotlib.pyplot as plt
import networkx as nx
import community  # Louvain algorithm for community detection
from networkx.algorithms.community import girvan_newman, k_clique_communities
from itertools import islice

# Read the dataset and create a graph
graph = nx.read_edgelist("email-Eu-core.txt")

# Limit the graph to a subgraph with fewer nodes for faster processing
graph = graph.subgraph(list(graph.nodes)[:10])

# Define the graph types
graph_types = ['Simple Graph', 'DiGraph', 'MultiGraph', 'MultiDiGraph']

# Initialize lists to store the number of communities detected by each algorithm for different graph types
girvan_newman_communities = []
louvain_communities = []
clique_communities = []
```

```python
# Function to apply Girvan-Newman algorithm
def apply_girvan_newman(graph):
    comp = girvan_newman(graph)
    limited_levels = list(islice(comp, 1))  # Limit to the first iteration
    return len(limited_levels[0])

# Function to apply Louvain algorithm
def apply_louvain(graph):
    partition = community.best_partition(graph)
    return len(set(partition.values()))

# Function to apply Clique Percolation algorithm
def apply_clique_percolation(graph, k=3):
    cliques = list(k_clique_communities(graph, k))
    return len(cliques)

# Apply Girvan-Newman algorithm to detect communities
for graph_type in graph_types:
    if graph_type == 'Simple Graph':
        g_newman = nx.Graph(graph)
    elif graph_type == 'DiGraph':
        g_newman = nx.DiGraph(graph)
    elif graph_type == 'MultiGraph':
        g_newman = nx.MultiGraph(graph)
    elif graph_type == 'MultiDiGraph':
        g_newman = nx.MultiDiGraph(graph)

    communities_count = apply_girvan_newman(g_newman)
    girvan_newman_communities.append(communities_count)

# Apply Louvain algorithm to detect communities (only for undirected graph types)
for graph_type in graph_types:
    if graph_type == 'Simple Graph':
        g_louvain = nx.Graph(graph)
        communities_count = apply_louvain(g_louvain)
        louvain_communities.append(communities_count)
    elif graph_type == 'MultiGraph':
        g_louvain = nx.MultiGraph(graph)
        communities_count = apply_louvain(g_louvain)
        louvain_communities.append(communities_count)
    else:
        louvain_communities.append(None)  # Louvain algorithm is not applicable

# Apply Clique Percolation algorithm to detect communities (only for undirected graph types)
for graph_type in graph_types:
    if graph_type == 'Simple Graph':
        g_clique = nx.Graph(graph)
        communities_count = apply_clique_percolation(g_clique)
        clique_communities.append(communities_count)
    elif graph_type == 'MultiGraph':
        g_clique = nx.MultiGraph(graph)
        communities_count = apply_clique_percolation(g_clique)
        clique_communities.append(communities_count)
    else:
        clique_communities.append(None)  # Clique Percolation is not applicable

# Create a bar chart
fig, ax = plt.subplots(figsize=(12, 8))

bar_width = 0.25
index = range(len(graph_types))

# Plotting bars for each algorithm
bar1 = ax.bar(index, girvan_newman_communities, bar_width, label='Girvan-Newman')
bar2 = ax.bar([i + bar_width for i in index],
              [count if count is not None else 0 for count in louvain_communities],
              bar_width, label='Louvain')
bar3 = ax.bar([i + 2 * bar_width for i in index],
              [count if count is not None else 0 for count in clique_communities],
              bar_width, label='Clique Percolation')

# Adding labels and title
ax.set_xlabel('Graph Types')
ax.set_ylabel('Number of Communities')
ax.set_title('Number of Communities Detected by Each Algorithm for Different Graph Types')
ax.set_xticks([i + bar_width for i in index])
ax.set_xticklabels(graph_types)
ax.legend()

# Display the bar chart
plt.tight_layout()
plt.show()
```



facebook_combined Dataset

Non-overlapping/Disjoint Algorithm

1. Reading the graph from the edge list file

In [1]:
```
import networkx as nx

# Read the graph from an edge list file
graph = nx.read_edgelist('facebook_combined.txt', create_using=nx.Graph(), nodetype=int)
# Number of nodes
print("Number of nodes:", graph.number_of_nodes())
# Number of edges
print("Number of edges:", graph.number_of_edges())
# Average degree
print("Average degree:", sum(dict(graph.degree()).values()) / graph.number_of_nodes())
```

```
Number of nodes: 4039
Number of edges: 88234
Average degree: 43.69101262688784
```

2. Applying the Girvan-Newman Algorithm

In [8]:
```
# from networkx.algorithms.community import girvan_newman
# import networkx as nx

# # Apply Girvan-Newman algorithm
# communities = girvan_newman(graph)
# top_level_communities = next(communities)

# # Sort communities by size (number of nodes in each community)
# sorted_communities = sorted(map(list, top_level_communities), key=len, reverse=True)

# # Print the sizes of the top 50 communities
# print("Top 50 Girvan-Newman communities:")
# for i, community in enumerate(sorted_communities[:50]):
#     print(f"Community {i+1}: Size {len(community)}")
```

```
Top 100 Louvain communities:
Community 1: Size 50, Nodes: ['17', '6', '8', '36', '81', '19', '20', '11', '52', '44', '89', '35', '58', '86',
'70', '46', '93', '15', '63', '78', '14', '74', '95', '77', '0', '2', '45', '49', '76', '12', '64', '96', '18',
'10', '50', '60', '4', '47', '32', '33', '41', '97', '42', '71', '90', '91', '37', '28', '43', '34']
Community 2: Size 20, Nodes: ['75', '66', '21', '98', '56', '62', '13', '9', '25', '69', '67', '72', '79', '39',
'55', '40', '26', '85', '59', '3']
Community 3: Size 13, Nodes: ['30', '48', '24', '27', '57', '80', '88', '53', '94', '54', '1', '92', '73']
Community 4: Size 9, Nodes: ['38', '82', '7', '22', '29', '87', '5', '16', '65']
Community 5: Size 8, Nodes: ['61', '83', '31', '84', '99', '68', '23', '51']
```

3. Applying the Louvain Algorithm

In [9]:

```python
# import networkx as nx
# import community as community_louvain
# import igraph as ig
# import leidenalg as la
# from networkx.algorithms.community import girvan_newman

# import community as community_louvain

# # Apply Louvain algorithm
# partition_louvain = community_louvain.best_partition(graph)
# print("Louvain communities:", partition_louvain)

from networkx.algorithms.community import girvan_newman
import networkx as nx

# Read the dataset and create a graph
graph = nx.read_edgelist("facebook_combined.txt")

# Limit the graph to a subgraph with fewer nodes for faster processing
graph = graph.subgraph(list(graph.nodes)[:100])

# Apply Girvan-Newman algorithm
communities = girvan_newman(graph)
top_level_communities = next(communities)

# Sort communities by size (number of nodes in each community)
sorted_communities = sorted(map(list, top_level_communities), key=len, reverse=True)

# Print the sizes of the top 5 communities
print("Top 5 Girvan-Newman communities:")
for i, community in enumerate(sorted_communities[:5]):
    print(f"Community {i+1}: Size {len(community)}")
```

```
Top 5 Girvan-Newman communities:
Community 1: Size 99
Community 2: Size 1
```

4. Applying the Leiden Algorithm

In [10]:

```python
# import igraph as ig
# import leidenalg as la

# # Convert NetworkX graph to igraph
# ig_graph = ig.Graph.TupleList(graph.edges(), directed=False)

# # Apply Leiden algorithm
# partition = la.find_partition(ig_graph, la.ModularityVertexPartition)
# print("Leiden communities:", partition)


import igraph as ig
import leidenalg as la

# Convert NetworkX graph to igraph
ig_graph = ig.Graph.TupleList(graph.edges(), directed=False)

# Apply Leiden algorithm
partition = la.find_partition(ig_graph, la.ModularityVertexPartition)

# Count the size of each community
community_sizes = {}
for node, community_id in enumerate(partition.membership):
    community_sizes.setdefault(community_id, 0)
    community_sizes[community_id] += 1

# Sort communities by size
sorted_communities = sorted(community_sizes.items(), key=lambda x: x[1], reverse=True)
```

```
# Get the top 100 communities
top_100_communities = sorted_communities[:50]

print("Top 100 Leiden communities:")
for i, (community_id, size) in enumerate(top_100_communities):
    nodes_in_community = [node for node, comm_id in enumerate(partition.membership) if comm_id == community_id]
    print(f"Community {i+1}: Size {size}, Nodes: {nodes_in_community}")
```

```
Top 100 Leiden communities:
Community 1: Size 50, Nodes: [1, 6, 7, 8, 9, 10, 11, 12, 13, 19, 20, 34, 35, 36, 37, 41, 47, 54, 55, 56, 58, 59,
60, 61, 62, 63, 64, 68, 73, 74, 75, 76, 77, 78, 79, 80, 82, 83, 85, 86, 87, 88, 89, 90, 91, 92, 93, 95, 97, 98]
Community 2: Size 20, Nodes: [0, 2, 3, 4, 5, 16, 17, 18, 21, 22, 24, 25, 26, 28, 51, 53, 57, 69, 70, 71]
Community 3: Size 13, Nodes: [27, 31, 32, 33, 42, 43, 44, 45, 46, 65, 66, 67, 72]
Community 4: Size 9, Nodes: [14, 15, 48, 49, 50, 52, 81, 84, 96]
Community 5: Size 8, Nodes: [23, 29, 30, 38, 39, 40, 94, 99]
```

Overlapping Algorithm

Clique Percolation

In [11]:

```
import networkx as nx

# Function to read a subset of the edges from the file
def read_subset_of_edges(file_path, num_edges):
    edges = []
    with open(file_path, 'r') as f:
        for _ in range(num_edges):
            line = f.readline().strip()
            if not line:
                break
            edges.append(tuple(map(int, line.split())))
    return edges

# Path to the edge list file
file_path = 'facebook_combined.txt'

# Number of edges to read
num_edges = 2000  # Adjust this number to control the subset size

# Read a subset of the edges
edges = read_subset_of_edges(file_path, num_edges)

# Create a graph from the subset of edges
graph = nx.Graph()
graph.add_edges_from(edges)

# Apply Clique Percolation algorithm with k-clique size, k
k = 4
communities = list(nx.algorithms.community.k_clique_communities(graph, k))

# Convert communities to list of lists for better readability
communities = [list(c) for c in communities]

# Display the number of communities found
print(f"Number of communities found: {len(communities)}")

# Display the first 100 communities
for i, community in enumerate(communities[:100], 1):  # Adjust the number of communities to display
    print(f"Community {i}: {community}")
```

```
Number of communities found: 9
Community 1: [0, 1, 3, 5, 7, 9, 10, 13, 16, 21, 22, 23, 24, 25, 26, 27, 29, 30, 31, 38, 39, 40, 45, 48, 51, 53,
54, 55, 56, 57, 59, 60, 62, 63, 65, 66, 67, 69, 72, 73, 75, 77, 79, 80, 82, 83, 84, 85, 87, 88, 92, 94, 96, 98,
101, 103, 104, 105, 106, 109, 113, 118, 119, 121, 122, 123, 126, 128, 129, 130, 132, 133, 134, 136, 141, 142, 14
8, 156, 158, 161, 165, 168, 169, 170, 172, 176, 180, 185, 186, 187, 188, 194, 196, 199, 200, 203, 204, 211, 212,
213, 221, 222, 223, 224, 231, 232, 236, 237, 238, 239, 242, 246, 248, 249, 252, 254, 257, 258, 261, 265, 266, 26
8, 271, 272, 274, 276, 277, 280, 283, 285, 290, 291, 295, 297, 298, 299, 300, 302, 303, 304, 308, 313, 314, 315,
317, 320, 322, 323, 325, 329, 330, 331, 332, 334, 336, 339, 341, 342, 344, 345, 346, 347]
Community 2: [0, 2, 326, 137, 140, 333, 14, 144, 17, 337, 19, 20, 149, 214, 343, 151, 28, 93, 32, 226, 167, 41,
44, 111, 115, 116, 310, 312]
Community 3: [0, 195, 4, 328, 78, 273, 306, 275, 181, 218]
Community 4: [0, 319, 6, 327, 147, 19, 89, 219, 95]
Community 5: [0, 193, 259, 8, 264, 201, 110, 245, 91]
Community 6: [0, 225, 99, 68, 227, 102, 263, 296, 175, 143, 177, 23]
Community 7: [0, 25, 76, 270]
Community 8: [64, 0, 100, 150, 119, 217, 189]
Community 9: [0, 58, 107, 171]
```

Label Propagation Paper Link: Finding overlapping communities

in networks by label

In [12]:
```python
import networkx as nx
import random

# Function to read a subset of the edges from the file
def read_subset_of_edges(file_path, num_edges):
    edges = []
    with open(file_path, 'r') as f:
        for _ in range(num_edges):
            line = f.readline().strip()
            if not line:
                break
            edges.append(tuple(map(int, line.split())))
    return edges

# Path to the edge list file
file_path = 'facebook_combined.txt'

# Number of edges to read
num_edges = 2000  # Adjust this number to control the subset size

# Read a subset of the edges
edges = read_subset_of_edges(file_path, num_edges)

# Create a graph from the subset of edges
graph = nx.Graph()
graph.add_edges_from(edges)

# Implement the Label Propagation Algorithm
def label_propagation(graph):
    # Initialize each node with a unique label
    labels = {node: node for node in graph.nodes()}
    nodes = list(graph.nodes())
    random.shuffle(nodes)

    while True:
        updated = False
        # For each node, update its label based on the most frequent label of its neighbors
        for node in nodes:
            if graph.degree(node) == 0:
                continue
            neighbor_labels = [labels[neighbor] for neighbor in graph.neighbors(node)]
            most_frequent_label = max(set(neighbor_labels), key=neighbor_labels.count)
            if labels[node] != most_frequent_label:
                labels[node] = most_frequent_label
                updated = True

        # If no labels were updated, the algorithm has converged
        if not updated:
            break

    # Group nodes by labels
    communities = {}
    for node, label in labels.items():
        if label not in communities:
            communities[label] = []
        communities[label].append(node)

    return list(communities.values())

# Detect communities using Label Propagation
communities = label_propagation(graph)

# Display the number of communities found
print(f"Number of communities found: {len(communities)}")

# Display the first few communities
for i, community in enumerate(communities[:10], 1):  # Adjust the number of communities to display
    print(f"Community {i}: {community}")
```

```
Number of communities found: 3
Community 1: [0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26,
27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54,
55, 56, 57, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83,
84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96, 97, 98, 99, 100, 101, 102, 103, 104, 105, 106, 108, 109, 110
, 111, 112, 113, 114, 115, 116, 117, 118, 119, 120, 121, 122, 123, 124, 125, 126, 127, 128, 129, 130, 131, 132,
133, 134, 135, 136, 137, 138, 139, 140, 141, 142, 143, 144, 145, 146, 147, 148, 149, 150, 151, 152, 153, 154, 15
5, 156, 157, 158, 159, 160, 161, 162, 163, 164, 165, 166, 167, 168, 169, 170, 171, 172, 173, 174, 175, 176, 177,
178, 179, 180, 181, 182, 183, 184, 185, 186, 187, 188, 189, 190, 191, 192, 193, 194, 195, 196, 197, 198, 199, 20
0, 201, 202, 203, 204, 205, 206, 207, 208, 209, 210, 211, 212, 213, 214, 215, 216, 217, 218, 219, 220, 221, 222,
223, 224, 225, 226, 227, 228, 229, 230, 231, 232, 233, 234, 235, 236, 237, 238, 239, 240, 241, 242, 243, 244, 24
5, 246, 247, 248, 249, 250, 251, 252, 253, 254, 255, 256, 257, 258, 259, 260, 261, 262, 263, 264, 265, 266, 267,
268, 269, 270, 271, 272, 273, 274, 275, 276, 277, 278, 279, 280, 281, 282, 283, 284, 285, 286, 287, 288, 289, 29
0, 291, 292, 293, 294, 295, 296, 297, 298, 299, 300, 301, 302, 303, 304, 305, 306, 307, 308, 309, 310, 311, 312,
313, 314, 315, 316, 317, 318, 319, 320, 321, 322, 323, 324, 325, 326, 327, 328, 329, 330, 331, 332, 333, 334, 33
5, 336, 337, 338, 339, 340, 341, 342, 343, 344, 345, 346, 347, 348, 414, 428]
Community 2: [58, 1684, 1912, 2814, 2838, 2885, 3003, 3173, 3290]
Community 3: [107, 353, 363, 366, 376, 389, 420, 475, 483, 484, 517, 526, 538, 563, 566, 580, 596, 601, 606, 629
, 637, 641, 649, 651, 896, 897, 898, 899, 900, 901, 902, 903, 904, 905, 906, 907, 908, 909, 910, 911, 912, 913,
914, 915, 916, 917, 918, 919, 920, 921, 922, 923, 924, 925, 926, 927, 928, 929, 930, 931, 932, 933, 934, 935, 93
6, 937, 938, 939, 940, 941, 942, 943, 944, 945, 946, 947, 948, 949, 950, 951, 952, 953, 954, 955, 956, 957, 958,
959, 960, 961, 962, 963, 964, 965, 966, 967, 968, 969, 970, 971, 972, 973, 974, 975, 976, 977, 978, 979, 980, 98
1, 982, 983, 984, 985, 986, 987, 988, 989, 990, 991, 992, 993, 994, 995, 996, 997, 998, 999, 1000, 1001, 1002, 1
003, 1004, 1005, 1006, 1007, 1008, 1009, 1010, 1011, 1012, 1013, 1014, 1015, 1016, 1017, 1018, 1019, 1020, 1021,
1022, 1023, 1024, 1025, 1026, 1027, 1028, 1029, 1030, 1031, 1032, 1033, 1034, 1035, 1036, 1037, 1038, 1039, 1040
, 1041, 1042, 1043, 1044, 1045, 1046, 1047, 1048, 1049, 1050, 1051, 1052, 1053, 1054, 1055, 1056, 1057, 1058, 10
59, 1060, 1061, 1062, 1063, 1064, 1065, 1066, 1067, 1068, 1069, 1070, 1071, 1072, 1073, 1074, 1075, 1076, 1077,
1078, 1079, 1080, 1081, 1082, 1083, 1084, 1085, 1086, 1087, 1088, 1089, 1090, 1091, 1092, 1093, 1094, 1095, 1096
, 1097, 1098, 1099, 1100, 1101, 1102, 1103, 1104, 1105, 1106, 1107, 1108, 1109, 1110, 1111, 1112, 1113, 1114, 11
15, 1116, 1117, 1118, 1119, 1120, 1121, 1122, 1123, 1124, 1125, 1126, 1127, 1128, 1129, 1130, 1131, 1132, 1133,
1134, 1135, 1136, 1137, 1138, 1139, 1140, 1141, 1142, 1143, 1144, 1145, 1146, 1147, 1148, 1149, 1150, 1151, 1152
, 1153, 1154, 1155, 1156, 1157, 1158, 1159, 1160, 1161, 1162, 1163, 1164, 1165, 1166, 1167, 1168, 1169, 1170, 11
71, 1172, 1173, 1174, 1175, 1176, 1177, 1178, 1179, 1180, 1181, 1182, 1183, 1184, 1185, 1186, 1187, 1188, 1189,
1190, 1191, 1192, 1193, 1194, 1195, 1196, 1197, 1198, 1199, 1200, 1201, 1202, 1203, 1204, 1205, 1206, 1207, 1208
, 1209, 1210, 1211, 1212, 1213, 1214, 1215, 1216, 1217, 1218, 1219, 1220, 1221, 1222, 1223, 1224, 1225, 1226]
```

Propagation

In [13]:

```python
import networkx as nx
import random

# Function to read a subset of the edges from the file
def read_subset_of_edges(file_path, num_edges):
    edges = []
    with open(file_path, 'r') as f:
        for _ in range(num_edges):
            line = f.readline().strip()
            if not line:
                break
            edges.append(tuple(map(int, line.split())))
    return edges

# Path to the edge list file
file_path = 'facebook_combined.txt'

# Number of edges to read
num_edges = 2000  # Adjust this number to control the subset size

# Read a subset of the edges
edges = read_subset_of_edges(file_path, num_edges)

# Create a graph from the subset of edges
graph = nx.Graph()
graph.add_edges_from(edges)

# Label Propagation Algorithm implementation
def label_propagation(graph):
    # Initialize each node with a unique label
    labels = {node: node for node in graph.nodes()}
    nodes = list(graph.nodes())
    random.shuffle(nodes)

    while True:
        updated = False
        # For each node, update its label based on the most frequent label of its neighbors
        for node in nodes:
            if graph.degree(node) == 0:
                continue
            neighbor_labels = [labels[neighbor] for neighbor in graph.neighbors(node)]
            most_frequent_label = max(set(neighbor_labels), key=neighbor_labels.count)
            if labels[node] != most_frequent_label:
```

```
                labels[node] = most_frequent_label
                updated = True

        # If no labels were updated, the algorithm has converged
        if not updated:
            break

    # Group nodes by labels
    communities = {}
    for node, label in labels.items():
        if label not in communities:
            communities[label] = []
        communities[label].append(node)

    return list(communities.values())

# Detect communities using Label Propagation
communities = label_propagation(graph)

# Display the number of communities found
print(f"Number of communities found: {len(communities)}")

# Display the first few communities
for i, community in enumerate(communities[:10], 1):  # Adjust the number of communities to display
    print(f"Community {i}: {community}")
```

```
Number of communities found: 3
Community 1: [0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26,
27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54,
55, 56, 57, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83,
84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96, 97, 98, 99, 100, 101, 102, 103, 104, 105, 106, 108, 109, 110
, 111, 112, 113, 114, 115, 116, 117, 118, 119, 120, 121, 122, 123, 124, 125, 126, 127, 128, 129, 130, 131, 132,
133, 134, 135, 136, 137, 138, 139, 140, 141, 142, 143, 144, 145, 146, 147, 148, 149, 150, 151, 152, 153, 154, 15
5, 156, 157, 158, 159, 160, 161, 162, 163, 164, 165, 166, 167, 168, 169, 170, 171, 172, 173, 174, 175, 176, 177,
178, 179, 180, 181, 182, 183, 184, 185, 186, 187, 188, 189, 190, 191, 192, 193, 194, 195, 196, 197, 198, 199, 20
0, 201, 202, 203, 204, 205, 206, 207, 208, 209, 210, 211, 212, 213, 214, 215, 216, 217, 218, 219, 220, 221, 222,
223, 224, 225, 226, 227, 228, 229, 230, 231, 232, 233, 234, 235, 236, 237, 238, 239, 240, 241, 242, 243, 244, 24
5, 246, 247, 248, 249, 250, 251, 252, 253, 254, 255, 256, 257, 258, 259, 260, 261, 262, 263, 264, 265, 266, 267,
268, 269, 270, 271, 272, 273, 274, 275, 276, 277, 278, 279, 280, 281, 282, 283, 284, 285, 286, 287, 288, 289, 29
0, 291, 292, 293, 294, 295, 296, 297, 298, 299, 300, 301, 302, 303, 304, 305, 306, 307, 308, 309, 310, 311, 312,
313, 314, 315, 316, 317, 318, 319, 320, 321, 322, 323, 324, 325, 326, 327, 328, 329, 330, 331, 332, 333, 334, 33
5, 336, 337, 338, 339, 340, 341, 342, 343, 344, 345, 346, 347, 348, 414, 428]
Community 2: [58, 1684, 1912, 2814, 2838, 2885, 3003, 3173, 3290]
Community 3: [107, 353, 363, 366, 376, 389, 420, 475, 483, 484, 517, 526, 538, 563, 566, 580, 596, 601, 606, 629
, 637, 641, 649, 651, 896, 897, 898, 899, 900, 901, 902, 903, 904, 905, 906, 907, 908, 909, 910, 911, 912, 913,
914, 915, 916, 917, 918, 919, 920, 921, 922, 923, 924, 925, 926, 927, 928, 929, 930, 931, 932, 933, 934, 935, 93
6, 937, 938, 939, 940, 941, 942, 943, 944, 945, 946, 947, 948, 949, 950, 951, 952, 953, 954, 955, 956, 957, 958,
959, 960, 961, 962, 963, 964, 965, 966, 967, 968, 969, 970, 971, 972, 973, 974, 975, 976, 977, 978, 979, 980, 98
1, 982, 983, 984, 985, 986, 987, 988, 989, 990, 991, 992, 993, 994, 995, 996, 997, 998, 999, 1000, 1001, 1002, 1
003, 1004, 1005, 1006, 1007, 1008, 1009, 1010, 1011, 1012, 1013, 1014, 1015, 1016, 1017, 1018, 1019, 1020, 1021,
1022, 1023, 1024, 1025, 1026, 1027, 1028, 1029, 1030, 1031, 1032, 1033, 1034, 1035, 1036, 1037, 1038, 1039, 1040
, 1041, 1042, 1043, 1044, 1045, 1046, 1047, 1048, 1049, 1050, 1051, 1052, 1053, 1054, 1055, 1056, 1057, 1058, 10
59, 1060, 1061, 1062, 1063, 1064, 1065, 1066, 1067, 1068, 1069, 1070, 1071, 1072, 1073, 1074, 1075, 1076, 1077,
1078, 1079, 1080, 1081, 1082, 1083, 1084, 1085, 1086, 1087, 1088, 1089, 1090, 1091, 1092, 1093, 1094, 1095, 1096
, 1097, 1098, 1099, 1100, 1101, 1102, 1103, 1104, 1105, 1106, 1107, 1108, 1109, 1110, 1111, 1112, 1113, 1114, 11
15, 1116, 1117, 1118, 1119, 1120, 1121, 1122, 1123, 1124, 1125, 1126, 1127, 1128, 1129, 1130, 1131, 1132, 1133,
1134, 1135, 1136, 1137, 1138, 1139, 1140, 1141, 1142, 1143, 1144, 1145, 1146, 1147, 1148, 1149, 1150, 1151, 1152
, 1153, 1154, 1155, 1156, 1157, 1158, 1159, 1160, 1161, 1162, 1163, 1164, 1165, 1166, 1167, 1168, 1169, 1170, 11
71, 1172, 1173, 1174, 1175, 1176, 1177, 1178, 1179, 1180, 1181, 1182, 1183, 1184, 1185, 1186, 1187, 1188, 1189,
1190, 1191, 1192, 1193, 1194, 1195, 1196, 1197, 1198, 1199, 1200, 1201, 1202, 1203, 1204, 1205, 1206, 1207, 1208
, 1209, 1210, 1211, 1212, 1213, 1214, 1215, 1216, 1217, 1218, 1219, 1220, 1221, 1222, 1223, 1224, 1225, 1226]
```

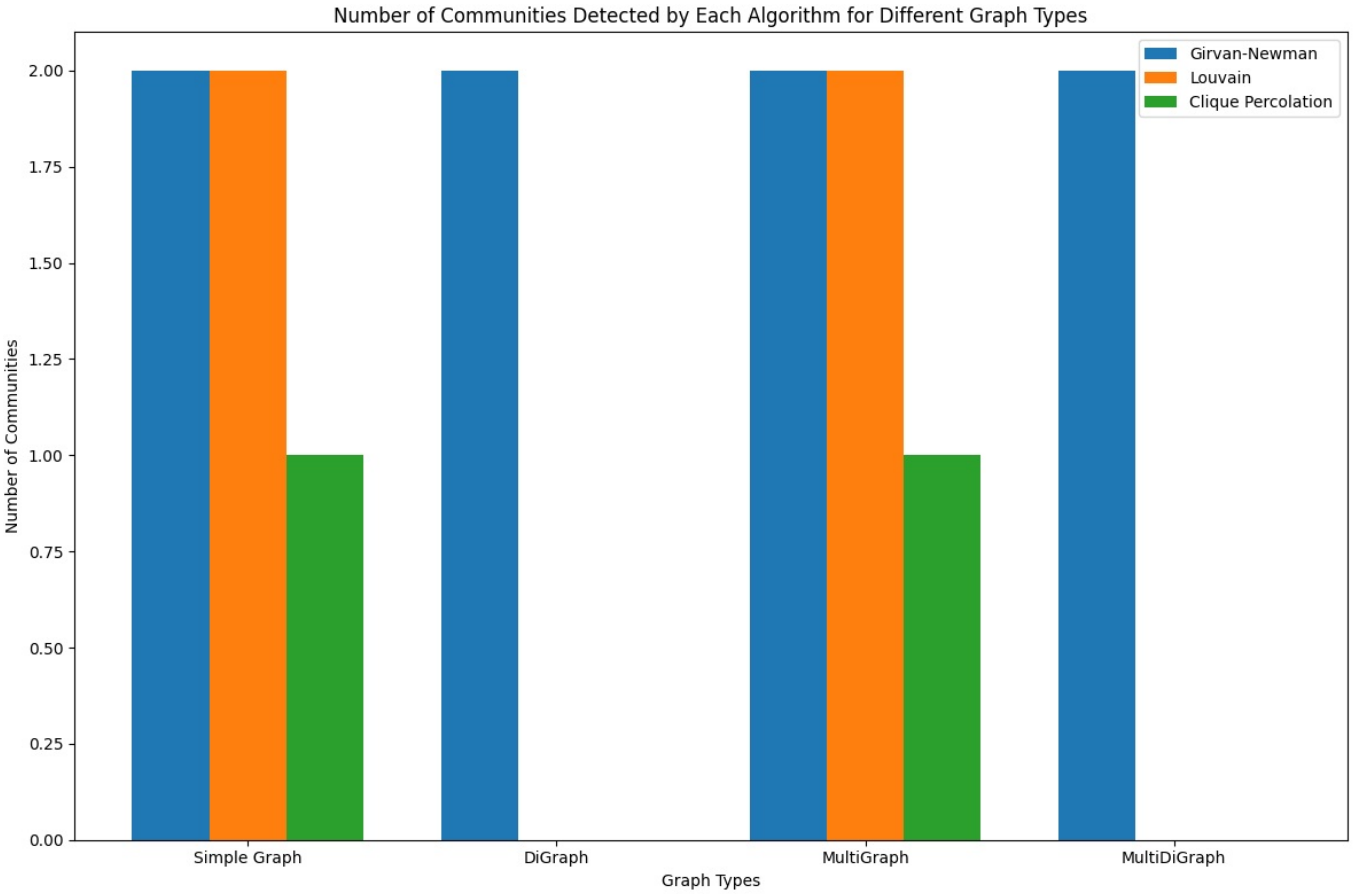
Bar Chart

In [6]:

```
import matplotlib.pyplot as plt
import networkx as nx
import community  # Louvain algorithm for community detection
from networkx.algorithms.community import girvan_newman, k_clique_communities
from itertools import islice

# Read the dataset and create a graph
graph = nx.read_edgelist("facebook_combined.txt")

# Limit the graph to a subgraph with fewer nodes for faster processing
graph = graph.subgraph(list(graph.nodes)[:10])

# Define the graph types
graph_types = ['Simple Graph', 'DiGraph', 'MultiGraph', 'MultiDiGraph']

# Initialize lists to store the number of communities detected by each algorithm for different graph types
girvan_newman_communities = []
louvain_communities = []
```

```
clique_communities = []

# Function to apply Girvan-Newman algorithm
def apply_girvan_newman(graph):
    comp = girvan_newman(graph)
    limited_levels = list(islice(comp, 1))  # Limit to the first iteration
    return len(limited_levels[0])

# Function to apply Louvain algorithm
def apply_louvain(graph):
    partition = community.best_partition(graph)
    return len(set(partition.values()))

# Function to apply Clique Percolation algorithm
def apply_clique_percolation(graph, k=3):
    cliques = list(k_clique_communities(graph, k))
    return len(cliques)

# Apply Girvan-Newman algorithm to detect communities
for graph_type in graph_types:
    if graph_type == 'Simple Graph':
        g_newman = nx.Graph(graph)
    elif graph_type == 'DiGraph':
        g_newman = nx.DiGraph(graph)
    elif graph_type == 'MultiGraph':
        g_newman = nx.MultiGraph(graph)
    elif graph_type == 'MultiDiGraph':
        g_newman = nx.MultiDiGraph(graph)

    communities_count = apply_girvan_newman(g_newman)
    girvan_newman_communities.append(communities_count)

# Apply Louvain algorithm to detect communities (only for undirected graph types)
for graph_type in graph_types:
    if graph_type == 'Simple Graph':
        g_louvain = nx.Graph(graph)
        communities_count = apply_louvain(g_louvain)
        louvain_communities.append(communities_count)
    elif graph_type == 'MultiGraph':
        g_louvain = nx.MultiGraph(graph)
        communities_count = apply_louvain(g_louvain)
        louvain_communities.append(communities_count)
    else:
        louvain_communities.append(None)  # Louvain algorithm is not applicable

# Apply Clique Percolation algorithm to detect communities (only for undirected graph types)
for graph_type in graph_types:
    if graph_type == 'Simple Graph':
        g_clique = nx.Graph(graph)
        communities_count = apply_clique_percolation(g_clique)
        clique_communities.append(communities_count)
    elif graph_type == 'MultiGraph':
        g_clique = nx.MultiGraph(graph)
        communities_count = apply_clique_percolation(g_clique)
        clique_communities.append(communities_count)
    else:
        clique_communities.append(None)  # Clique Percolation is not applicable

# Create a bar chart
fig, ax = plt.subplots(figsize=(12, 8))

bar_width = 0.25
index = range(len(graph_types))

# Plotting bars for each algorithm
bar1 = ax.bar(index, girvan_newman_communities, bar_width, label='Girvan-Newman')
bar2 = ax.bar([i + bar_width for i in index],
              [count if count is not None else 0 for count in louvain_communities],
              bar_width, label='Louvain')
bar3 = ax.bar([i + 2 * bar_width for i in index],
              [count if count is not None else 0 for count in clique_communities],
              bar_width, label='Clique Percolation')

# Adding labels and title
ax.set_xlabel('Graph Types')
ax.set_ylabel('Number of Communities')
ax.set_title('Number of Communities Detected by Each Algorithm for Different Graph Types')
ax.set_xticks([i + bar_width for i in index])
ax.set_xticklabels(graph_types)
ax.legend()

# Display the bar chart
plt.tight_layout()
```

```
plt.show()
```



p2p-Gnutella08 Dataset

Non-overlapping/Disjoint Algorithm

1. Reading the graph from the edge list file

In [3]:
```
import networkx as nx

# Read the graph from an edge list file
graph = nx.read_edgelist('p2p-Gnutella08.txt', create_using=nx.Graph(), nodetype=int)
# Number of nodes
print("Number of nodes:", graph.number_of_nodes())
# Number of edges
print("Number of edges:", graph.number_of_edges())
# Average degree
print("Average degree:", sum(dict(graph.degree()).values()) / graph.number_of_nodes())
```

```
Number of nodes: 6301
Number of edges: 20777
Average degree: 6.594826218060625
```

2. Applying the Girvan-Newman Algorithm

In [4]:
```
from networkx.algorithms.community import girvan_newman

# Apply Girvan-Newman algorithm
communities = girvan_newman(graph)
top_level_communities = next(communities)
sorted_communities = sorted(map(sorted, top_level_communities))
print("Girvan-Newman communities:", sorted_communities)
```

```
Girvan-Newman communities: [[0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 2
3, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 5
1, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 7
9, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96, 97, 98, 99, 100, 101, 102, 103, 104, 105,
106, 107, 108, 109, 110, 111, 112, 113, 114, 115, 116, 117, 118, 119, 120, 121, 122, 123, 124, 125, 126, 127, 12
8, 129, 130, 131, 132, 133, 134, 135, 136, 137, 138, 139, 140, 141, 142, 143, 144, 145, 146, 147, 148, 149, 150,
151, 152, 153, 154, 155, 156, 157, 158, 159, 160, 161, 162, 163, 164, 165, 166, 167, 168, 169, 170, 171, 172, 17
3, 174, 175, 176, 177, 178, 179, 180, 181, 182, 183, 184, 185, 186, 187, 188, 189, 190, 191, 192, 193, 194, 195,
```

196, 197, 198, 199, 200, 201, 202, 203, 204, 205, 206, 207, 208, 209, 210, 211, 212, 213, 214, 215, 216, 217, 21
8, 219, 220, 221, 222, 223, 224, 225, 226, 227, 228, 229, 230, 231, 232, 233, 234, 235, 236, 237, 238, 239, 240,
241, 242, 243, 244, 245, 246, 247, 248, 249, 250, 251, 252, 253, 254, 255, 256, 257, 258, 259, 260, 261, 262, 26
3, 264, 265, 266, 267, 268, 269, 270, 271, 272, 273, 274, 275, 276, 277, 278, 279, 280, 281, 282, 283, 284, 285,
286, 287, 288, 289, 290, 291, 292, 293, 294, 295, 296, 297, 298, 299, 300, 301, 302, 303, 304, 305, 306, 307, 30
8, 309, 310, 311, 312, 313, 314, 315, 316, 317, 318, 319, 320, 321, 322, 323, 324, 325, 326, 327, 328, 329, 330,
331, 332, 333, 334, 335, 336, 337, 338, 339, 340, 341, 342, 343, 344, 345, 346, 347, 348, 349, 350, 351, 352, 35
3, 354, 355, 356, 357, 358, 359, 360, 361, 362, 363, 364, 365, 366, 367, 368, 369, 370, 371, 372, 373, 374, 375,
376, 377, 378, 379, 380, 381, 382, 383, 384, 385, 386, 387, 388, 389, 390, 391, 392, 393, 394, 395, 396, 397, 39
8, 399, 400, 401, 402, 403, 404, 405, 406, 407, 408, 409, 410, 411, 412, 413, 414, 415, 416, 417, 418, 419, 420,
421, 422, 423, 424, 425, 426, 427, 428, 429, 430, 431, 432, 433, 434, 435, 436, 437, 438, 439, 440, 441, 442, 44
3, 444, 445, 446, 447, 448, 449, 450, 451, 452, 453, 454, 455, 456, 457, 458, 459, 460, 461, 462, 463, 464, 465,
466, 467, 468, 469, 470, 471, 472, 473, 474, 475, 476, 477, 478, 479, 480, 481, 482, 483, 484, 485, 486, 487, 48
8, 489, 490, 491, 492, 493, 494, 495, 496, 497, 498, 499, 500, 501, 502, 503, 504, 505, 506, 507, 508, 509, 510,
511, 512, 513, 514, 515, 516, 517, 518, 519, 520, 521, 522, 523, 524, 525, 526, 527, 528, 529, 530, 531, 532, 53
3, 534, 535, 536, 537, 538, 539, 540, 541, 542, 543, 544, 545, 546, 547, 548, 549, 550, 551, 552, 553, 554, 555,
556, 557, 558, 559, 560, 561, 562, 563, 564, 565, 566, 567, 568, 569, 570, 571, 572, 573, 574, 575, 576, 577, 57
8, 579, 580, 581, 582, 583, 584, 585, 586, 587, 599, 600, 601, 602, 603, 604, 605, 606, 607, 608, 609, 610, 611,
612, 613, 614, 615, 616, 617, 618, 619, 620, 621, 622, 623, 624, 625, 626, 627, 628, 629, 630, 631, 632, 633, 63
4, 635, 636, 637, 638, 639, 640, 641, 642, 643, 644, 645, 646, 647, 648, 649, 650, 651, 652, 653, 654, 655, 656,
657, 658, 659, 660, 661, 662, 663, 664, 665, 666, 667, 668, 669, 670, 671, 672, 673, 674, 675, 676, 677, 678, 67
9, 680, 681, 682, 683, 684, 685, 686, 687, 688, 689, 690, 691, 692, 693, 694, 695, 696, 697, 698, 699, 700, 701,
702, 703, 704, 705, 706, 707, 708, 709, 710, 711, 712, 713, 714, 715, 716, 717, 718, 719, 720, 721, 722, 723, 72
4, 725, 726, 727, 728, 729, 730, 731, 732, 733, 734, 735, 736, 737, 738, 739, 740, 741, 742, 743, 744, 745, 746,
747, 748, 749, 750, 751, 752, 753, 754, 755, 756, 757, 758, 759, 760, 761, 762, 763, 764, 765, 766, 767, 768, 76
9, 770, 771, 772, 773, 774, 775, 776, 777, 778, 779, 780, 781, 782, 783, 784, 785, 786, 787, 788, 789, 790, 791,
792, 793, 794, 795, 796, 797, 798, 799, 800, 801, 802, 803, 804, 805, 806, 807, 808, 809, 810, 811, 812, 813, 81
4, 815, 816, 817, 818, 819, 820, 821, 822, 823, 824, 825, 826, 827, 828, 829, 830, 831, 832, 833, 834, 835, 836,
837, 838, 839, 840, 841, 842, 843, 844, 845, 846, 847, 848, 849, 850, 851, 852, 853, 854, 855, 856, 857, 858, 85
9, 860, 861, 862, 863, 864, 865, 866, 867, 868, 869, 870, 871, 872, 873, 874, 875, 876, 877, 878, 879, 880, 881,
882, 883, 884, 885, 886, 887, 888, 889, 890, 891, 892, 893, 894, 895, 896, 897, 898, 899, 900, 901, 902, 903, 90
4, 905, 906, 907, 908, 909, 910, 911, 912, 913, 914, 915, 916, 917, 918, 919, 920, 921, 922, 923, 924, 925, 926,
927, 928, 929, 930, 931, 932, 933, 934, 935, 936, 937, 938, 939, 940, 941, 942, 943, 944, 945, 946, 947, 948, 94
9, 950, 951, 952, 953, 954, 955, 956, 957, 958, 959, 960, 961, 962, 963, 964, 965, 966, 967, 968, 969, 970, 971,
972, 973, 974, 975, 976, 977, 978, 979, 980, 981, 982, 983, 984, 985, 986, 987, 988, 989, 990, 991, 992, 993, 99
4, 995, 996, 997, 998, 999, 1000, 1001, 1002, 1003, 1004, 1005, 1006, 1007, 1008, 1009, 1010, 1011, 1012, 1013,
1014, 1015, 1016, 1017, 1018, 1019, 1020, 1021, 1022, 1023, 1024, 1025, 1026, 1027, 1028, 1029, 1030, 1031, 1032
, 1033, 1034, 1035, 1036, 1037, 1038, 1039, 1040, 1041, 1042, 1043, 1044, 1045, 1046, 1047, 1048, 1049, 1050, 10
51, 1052, 1053, 1054, 1055, 1056, 1057, 1058, 1059, 1060, 1061, 1062, 1063, 1064, 1065, 1066, 1067, 1068, 1069,
1070, 1071, 1072, 1073, 1074, 1075, 1076, 1077, 1078, 1079, 1080, 1081, 1082, 1083, 1084, 1085, 1086, 1087, 1088
, 1089, 1090, 1091, 1092, 1093, 1094, 1095, 1096, 1097, 1098, 1099, 1100, 1101, 1102, 1103, 1104, 1105, 1106, 11
07, 1108, 1109, 1110, 1111, 1112, 1113, 1114, 1115, 1116, 1117, 1118, 1119, 1120, 1121, 1122, 1123, 1124, 1125,
1126, 1127, 1128, 1129, 1130, 1131, 1132, 1133, 1134, 1135, 1136, 1137, 1138, 1139, 1140, 1141, 1142, 1143, 1144
, 1145, 1146, 1147, 1148, 1149, 1150, 1151, 1152, 1153, 1154, 1155, 1156, 1157, 1158, 1159, 1160, 1161, 1162, 11
63, 1164, 1165, 1166, 1167, 1168, 1169, 1170, 1171, 1172, 1173, 1174, 1175, 1176, 1177, 1178, 1179, 1180, 1181,
1182, 1183, 1184, 1185, 1186, 1187, 1188, 1189, 1190, 1191, 1192, 1193, 1194, 1195, 1196, 1197, 1198, 1199, 1200
, 1201, 1202, 1203, 1204, 1205, 1206, 1207, 1208, 1209, 1210, 1211, 1212, 1213, 1214, 1215, 1216, 1217, 1218, 12
19, 1220, 1221, 1222, 1223, 1224, 1225, 1226, 1227, 1228, 1229, 1230, 1231, 1232, 1233, 1234, 1235, 1236, 1237,
1238, 1239, 1240, 1241, 1242, 1243, 1244, 1245, 1246, 1247, 1248, 1249, 1250, 1251, 1252, 1253, 1254, 1255, 1256
, 1257, 1258, 1259, 1260, 1261, 1262, 1263, 1264, 1265, 1266, 1267, 1268, 1269, 1270, 1271, 1272, 1273, 1274, 12
75, 1276, 1277, 1278, 1279, 1280, 1281, 1282, 1283, 1284, 1285, 1286, 1287, 1288, 1289, 1290, 1291, 1292, 1293,
1294, 1295, 1296, 1297, 1298, 1299, 1300, 1301, 1302, 1303, 1304, 1305, 1306, 1307, 1308, 1309, 1310, 1311, 1312
, 1313, 1314, 1315, 1316, 1317, 1318, 1319, 1320, 1321, 1322, 1323, 1324, 1325, 1326, 1327, 1328, 1329, 1330, 13
31, 1332, 1333, 1334, 1335, 1336, 1337, 1338, 1339, 1340, 1341, 1342, 1343, 1344, 1345, 1346, 1347, 1348, 1349,
1350, 1351, 1352, 1353, 1354, 1355, 1356, 1357, 1358, 1359, 1360, 1361, 1362, 1363, 1364, 1365, 1366, 1367, 1368
, 1369, 1370, 1371, 1372, 1373, 1374, 1375, 1376, 1377, 1378, 1379, 1380, 1381, 1382, 1383, 1384, 1385, 1386, 13
87, 1388, 1389, 1390, 1391, 1392, 1393, 1394, 1395, 1396, 1397, 1398, 1399, 1400, 1401, 1402, 1403, 1404, 1405,
1406, 1407, 1408, 1409, 1410, 1411, 1412, 1413, 1414, 1415, 1416, 1417, 1418, 1419, 1420, 1421, 1422, 1423, 1424
, 1425, 1426, 1427, 1428, 1429, 1430, 1431, 1432, 1433, 1434, 1435, 1436, 1437, 1438, 1439, 1440, 1441, 1442, 14
43, 1444, 1445, 1446, 1447, 1448, 1449, 1450, 1451, 1452, 1453, 1454, 1455, 1456, 1457, 1458, 1459, 1460, 1461,
1462, 1463, 1464, 1465, 1466, 1467, 1468, 1469, 1470, 1471, 1472, 1473, 1474, 1475, 1476, 1477, 1478, 1479, 1480
, 1481, 1482, 1483, 1484, 1485, 1486, 1487, 1488, 1489, 1490, 1491, 1492, 1493, 1494, 1495, 1496, 1497, 1498, 14
99, 1500, 1501, 1502, 1503, 1504, 1505, 1506, 1507, 1508, 1509, 1510, 1511, 1512, 1513, 1514, 1515, 1516, 1517,
1518, 1519, 1520, 1521, 1522, 1523, 1524, 1525, 1526, 1527, 1528, 1529, 1530, 1531, 1532, 1533, 1534, 1535, 1536
, 1537, 1538, 1539, 1540, 1541, 1542, 1543, 1544, 1545, 1546, 1547, 1548, 1549, 1550, 1551, 1552, 1553, 1554, 15
55, 1556, 1557, 1558, 1559, 1560, 1561, 1562, 1563, 1564, 1565, 1566, 1567, 1568, 1569, 1570, 1571, 1572, 1573,
1574, 1575, 1576, 1577, 1578, 1579, 1580, 1581, 1582, 1583, 1584, 1585, 1586, 1587, 1588, 1589, 1590, 1591, 1592
, 1593, 1594, 1595, 1596, 1597, 1598, 1599, 1600, 1601, 1602, 1603, 1604, 1605, 1606, 1607, 1608, 1609, 1610, 16
11, 1612, 1613, 1614, 1615, 1616, 1617, 1618, 1619, 1620, 1621, 1622, 1623, 1624, 1625, 1626, 1627, 1628, 1629,
1630, 1631, 1632, 1633, 1634, 1635, 1636, 1637, 1638, 1639, 1640, 1641, 1642, 1643, 1644, 1645, 1646, 1647, 1648
, 1649, 1650, 1651, 1652, 1653, 1654, 1655, 1656, 1657, 1658, 1659, 1660, 1661, 1662, 1663, 1664, 1665, 1666, 16
67, 1668, 1669, 1670, 1671, 1672, 1673, 1674, 1675, 1676, 1677, 1678, 1679, 1680, 1681, 1682, 1685, 1686, 1687,
1688, 1689, 1690, 1691, 1692, 1693, 1694, 1695, 1696, 1697, 1698, 1699, 1700, 1701, 1702, 1703, 1704, 1705, 1706
, 1707, 1708, 1709, 1710, 1711, 1712, 1713, 1714, 1715, 1716, 1717, 1718, 1719, 1720, 1721, 1722, 1723, 1724, 17
25, 1726, 1727, 1728, 1729, 1730, 1731, 1732, 1733, 1734, 1735, 1736, 1737, 1738, 1739, 1740, 1741, 1742, 1743,
1744, 1745, 1746, 1747, 1748, 1749, 1750, 1751, 1752, 1753, 1754, 1755, 1756, 1757, 1758, 1759, 1760, 1761, 1762
, 1763, 1764, 1765, 1766, 1767, 1768, 1769, 1770, 1771, 1772, 1773, 1774, 1775, 1776, 1777, 1778, 1779, 1780, 17
81, 1782, 1783, 1784, 1785, 1786, 1787, 1788, 1789, 1790, 1791, 1792, 1793, 1794, 1795, 1796, 1797, 1798, 1799,
1800, 1801, 1802, 1803, 1804, 1805, 1806, 1807, 1808, 1809, 1810, 1811, 1812, 1813, 1814, 1815, 1816, 1817, 1818
, 1819, 1820, 1821, 1822, 1823, 1824, 1825, 1826, 1827, 1828, 1829, 1830, 1831, 1832, 1833, 1834, 1835, 1836, 18
37, 1838, 1839, 1840, 1841, 1842, 1843, 1844, 1845, 1846, 1847, 1848, 1849, 1850, 1851, 1852, 1853, 1854, 1855,
1856, 1857, 1858, 1859, 1860, 1861, 1862, 1863, 1864, 1865, 1866, 1867, 1868, 1869, 1870, 1871, 1872, 1873, 1874
, 1875, 1876, 1877, 1878, 1879, 1880, 1881, 1882, 1883, 1884, 1885, 1886, 1887, 1888, 1889, 1890, 1891, 1892, 18

93, 1894, 1895, 1896, 1897, 1898, 1899, 1900, 1901, 1902, 1903, 1904, 1905, 1906, 1907, 1908, 1909, 1910, 1911, 1912, 1913, 1914, 1915, 1916, 1917, 1918, 1919, 1920, 1921, 1922, 1923, 1924, 1925, 1926, 1927, 1928, 1929, 1930, 1931, 1932, 1933, 1934, 1935, 1936, 1937, 1938, 1939, 1940, 1941, 1942, 1943, 1944, 1945, 1946, 1947, 1948, 1949, 1950, 1951, 1952, 1953, 1954, 1955, 1956, 1957, 1958, 1959, 1960, 1961, 1962, 1963, 1964, 1965, 1966, 1967, 1968, 1969, 1970, 1971, 1972, 1973, 1974, 1975, 1976, 1977, 1978, 1979, 1980, 1981, 1982, 1983, 1984, 1985, 1986, 1987, 1988, 1989, 1990, 1991, 1992, 1993, 1994, 1995, 1996, 1997, 1998, 1999, 2000, 2001, 2002, 2003, 2004, 2005, 2006, 2007, 2008, 2009, 2010, 2011, 2012, 2013, 2014, 2015, 2016, 2017, 2018, 2019, 2020, 2021, 2022, 2023, 2024, 2025, 2026, 2027, 2028, 2029, 2030, 2031, 2032, 2033, 2034, 2035, 2036, 2037, 2038, 2039, 2040, 2041, 2042, 2043, 2044, 2045, 2046, 2047, 2048, 2049, 2050, 2051, 2052, 2053, 2054, 2055, 2056, 2057, 2058, 2059, 2060, 2061, 2062, 2063, 2064, 2065, 2066, 2067, 2068, 2069, 2070, 2071, 2072, 2073, 2074, 2075, 2076, 2077, 2078, 2079, 2080, 2081, 2082, 2083, 2084, 2085, 2086, 2087, 2088, 2089, 2090, 2091, 2092, 2093, 2094, 2095, 2096, 2097, 2098, 2099, 2100, 2101, 2102, 2103, 2104, 2105, 2106, 2107, 2108, 2109, 2110, 2111, 2112, 2113, 2114, 2115, 2116, 2117, 2118, 2119, 2120, 2121, 2122, 2123, 2124, 2125, 2126, 2127, 2128, 2129, 2130, 2131, 2132, 2133, 2134, 2135, 2136, 2137, 2138, 2139, 2140, 2141, 2142, 2143, 2144, 2145, 2146, 2147, 2148, 2149, 2150, 2151, 2152, 2153, 2154, 2155, 2156, 2157, 2158, 2159, 2160, 2161, 2162, 2163, 2164, 2165, 2166, 2167, 2168, 2169, 2170, 2171, 2172, 2173, 2174, 2175, 2176, 2177, 2178, 2179, 2180, 2181, 2182, 2183, 2184, 2185, 2186, 2187, 2188, 2189, 2190, 2191, 2192, 2193, 2194, 2195, 2196, 2197, 2198, 2199, 2200, 2201, 2202, 2203, 2204, 2205, 2206, 2207, 2208, 2209, 2210, 2211, 2212, 2213, 2214, 2215, 2216, 2217, 2218, 2219, 2220, 2221, 2222, 2223, 2224, 2225, 2226, 2227, 2228, 2229, 2230, 2231, 2232, 2233, 2234, 2235, 2236, 2237, 2238, 2239, 2240, 2241, 2242, 2243, 2244, 2245, 2246, 2247, 2248, 2249, 2250, 2251, 2252, 2253, 2254, 2255, 2256, 2257, 2258, 2259, 2260, 2261, 2262, 2263, 2264, 2265, 2266, 2267, 2268, 2269, 2270, 2271, 2272, 2273, 2274, 2275, 2276, 2277, 2278, 2279, 2280, 2281, 2282, 2283, 2284, 2285, 2286, 2287, 2288, 2289, 2290, 2291, 2292, 2293, 2294, 2295, 2296, 2297, 2298, 2299, 2300, 2301, 2302, 2303, 2304, 2305, 2306, 2307, 2308, 2309, 2310, 2311, 2312, 2313, 2314, 2315, 2316, 2317, 2318, 2319, 2320, 2321, 2322, 2323, 2324, 2325, 2326, 2327, 2328, 2329, 2330, 2331, 2332, 2333, 2334, 2335, 2336, 2337, 2338, 2339, 2340, 2341, 2342, 2343, 2344, 2345, 2346, 2347, 2348, 2349, 2350, 2351, 2352, 2353, 2354, 2355, 2356, 2357, 2358, 2359, 2360, 2361, 2362, 2363, 2364, 2365, 2366, 2367, 2368, 2369, 2370, 2371, 2372, 2373, 2374, 2375, 2376, 2377, 2378, 2379, 2380, 2381, 2382, 2383, 2384, 2385, 2386, 2387, 2388, 2389, 2390, 2391, 2392, 2393, 2394, 2395, 2396, 2397, 2398, 2399, 2400, 2401, 2402, 2403, 2404, 2405, 2406, 2407, 2408, 2409, 2410, 2411, 2412, 2413, 2414, 2415, 2416, 2417, 2418, 2419, 2420, 2421, 2422, 2423, 2424, 2425, 2426, 2427, 2428, 2429, 2430, 2431, 2432, 2433, 2434, 2435, 2436, 2437, 2438, 2439, 2440, 2441, 2442, 2443, 2444, 2445, 2446, 2447, 2448, 2449, 2450, 2451, 2452, 2453, 2454, 2455, 2456, 2457, 2458, 2459, 2460, 2461, 2462, 2463, 2464, 2465, 2466, 2467, 2468, 2469, 2470, 2471, 2472, 2473, 2474, 2475, 2476, 2477, 2478, 2479, 2480, 2481, 2482, 2483, 2484, 2485, 2486, 2487, 2488, 2489, 2490, 2491, 2492, 2493, 2494, 2495, 2496, 2497, 2498, 2499, 2500, 2501, 2502, 2503, 2504, 2505, 2506, 2507, 2508, 2509, 2510, 2511, 2512, 2513, 2514, 2515, 2516, 2517, 2518, 2519, 2520, 2521, 2522, 2523, 2524, 2525, 2526, 2527, 2528, 2529, 2530, 2531, 2532, 2533, 2534, 2535, 2536, 2537, 2538, 2539, 2540, 2541, 2542, 2543, 2544, 2545, 2546, 2547, 2548, 2549, 2550, 2551, 2552, 2553, 2554, 2555, 2556, 2557, 2558, 2559, 2560, 2561, 2562, 2563, 2564, 2565, 2566, 2567, 2568, 2569, 2570, 2571, 2572, 2573, 2574, 2575, 2576, 2577, 2578, 2579, 2580, 2581, 2582, 2583, 2584, 2585, 2586, 2587, 2588, 2589, 2590, 2591, 2592, 2593, 2594, 2595, 2596, 2597, 2598, 2599, 2600, 2601, 2602, 2603, 2604, 2605, 2606, 2607, 2608, 2609, 2610, 2611, 2612, 2613, 2614, 2615, 2616, 2617, 2618, 2619, 2620, 2621, 2622, 2623, 2624, 2625, 2626, 2627, 2628, 2629, 2630, 2631, 2632, 2633, 2634, 2635, 2636, 2637, 2638, 2639, 2640, 2641, 2642, 2643, 2644, 2645, 2646, 2647, 2648, 2649, 2650, 2651, 2652, 2653, 2654, 2655, 2656, 2657, 2658, 2659, 2660, 2661, 2662, 2663, 2664, 2665, 2666, 2667, 2668, 2669, 2670, 2671, 2672, 2673, 2674, 2675, 2676, 2677, 2678, 2679, 2680, 2681, 2682, 2683, 2684, 2685, 2686, 2687, 2688, 2689, 2690, 2691, 2692, 2693, 2694, 2695, 2696, 2697, 2698, 2699, 2700, 2701, 2702, 2703, 2704, 2705, 2706, 2707, 2708, 2709, 2710, 2711, 2712, 2713, 2714, 2715, 2716, 2717, 2718, 2719, 2720, 2721, 2722, 2723, 2724, 2725, 2726, 2727, 2728, 2729, 2730, 2731, 2732, 2733, 2734, 2735, 2736, 2737, 2738, 2739, 2740, 2741, 2742, 2743, 2744, 2745, 2746, 2747, 2748, 2749, 2750, 2751, 2752, 2753, 2754, 2755, 2756, 2757, 2758, 2759, 2760, 2761, 2762, 2763, 2764, 2765, 2766, 2767, 2768, 2769, 2770, 2771, 2772, 2773, 2774, 2775, 2776, 2777, 2778, 2779, 2780, 2781, 2782, 2783, 2784, 2785, 2786, 2787, 2788, 2789, 2790, 2791, 2792, 2793, 2794, 2795, 2796, 2797, 2798, 2799, 2800, 2801, 2802, 2803, 2804, 2805, 2806, 2807, 2808, 2809, 2810, 2811, 2812, 2813, 2814, 2815, 2816, 2817, 2818, 2819, 2820, 2821, 2822, 2823, 2824, 2825, 2826, 2827, 2828, 2829, 2830, 2831, 2832, 2833, 2834, 2835, 2836, 2837, 2838, 2839, 2840, 2841, 2842, 2843, 2844, 2845, 2846, 2847, 2848, 2849, 2850, 2851, 2852, 2853, 2854, 2855, 2856, 2857, 2858, 2859, 2860, 2861, 2862, 2863, 2864, 2865, 2866, 2867, 2868, 2869, 2870, 2871, 2872, 2873, 2874, 2875, 2876, 2877, 2878, 2879, 2880, 2881, 2882, 2883, 2884, 2885, 2886, 2887, 2888, 2889, 2890, 2891, 2892, 2893, 2894, 2895, 2896, 2897, 2898, 2899, 2900, 2901, 2902, 2903, 2904, 2905, 2906, 2907, 2908, 2909, 2910, 2911, 2912, 2913, 2914, 2915, 2916, 2917, 2918, 2919, 2920, 2921, 2922, 2923, 2924, 2925, 2926, 2927, 2928, 2929, 2930, 2931, 2932, 2933, 2934, 2935, 2936, 2937, 2938, 2939, 2940, 2941, 2942, 2943, 2944, 2945, 2946, 2947, 2948, 2949, 2950, 2951, 2952, 2953, 2954, 2955, 2956, 2957, 2958, 2959, 2960, 2961, 2962, 2963, 2964, 2965, 2966, 2967, 2968, 2969, 2970, 2971, 2972, 2973, 2974, 2975, 2976, 2977, 2978, 2979, 2980, 2981, 2982, 2983, 2984, 2985, 2986, 2987, 2988, 2989, 2990, 2991, 2992, 2993, 2994, 2995, 2996, 2997, 2998, 2999, 3000, 3001, 3002, 3003, 3004, 3005, 3006, 3007, 3008, 3009, 3010, 3011, 3012, 3013, 3014, 3015, 3016, 3017, 3018, 3019, 3020, 3021, 3022, 3023, 3024, 3025, 3026, 3027, 3028, 3029, 3030, 3031, 3032, 3033, 3034, 3035, 3036, 3037, 3038, 3039, 3040, 3041, 3042, 3043, 3044, 3045, 3046, 3047, 3048, 3049, 3050, 3051, 3052, 3053, 3054, 3055, 3056, 3057, 3058, 3059, 3060, 3061, 3062, 3063, 3064, 3065, 3066, 3067, 3068, 3069, 3070, 3071, 3072, 3073, 3074, 3075, 3076, 3077, 3078, 3079, 3080, 3081, 3082, 3083, 3084, 3085, 3086, 3087, 3088, 3089, 3090, 3091, 3092, 3093, 3094, 3095, 3096, 3097, 3098, 3099, 3100, 3101, 3102, 3103, 3104, 3105, 3106, 3107, 3108, 3109, 3110, 3111, 3112, 3113, 3114, 3115, 3116, 3117, 3118, 3119, 3120, 3121, 3122, 3123, 3124, 3125, 3126, 3127, 3128, 3129, 3130, 3131, 3132, 3133, 3134, 3135, 3136, 3137, 3138, 3139, 3140, 3141, 3142, 3143, 3144, 3145, 3146, 3147, 3148, 3149, 3150, 3151, 3152, 3153, 3154, 3155, 3156, 3157, 3158, 3159, 3160, 3161, 3162, 3163, 3164, 3165, 3166, 3167, 3168, 3169, 3170, 3171, 3172, 3173, 3174, 3175, 3176, 3177, 3178, 3179, 3180, 3181, 3182, 3183, 3184, 3185, 3186, 3187, 3188, 3189, 3190, 3191, 3192, 3193, 3194, 3195, 3196, 3197, 3198, 3199, 3200, 3201, 3202, 3203, 3204, 3205, 3206, 3207, 3208, 3209, 3210, 3211, 3212, 3213, 3214, 3215, 3216, 3217, 3218, 3219, 3220, 3221, 3222, 3223, 3224, 3225, 3226, 3227, 3228, 3229, 3230, 3231, 3232, 3233, 3234, 3235, 3236, 3237, 3238, 3239, 3240, 3241, 3242, 3243, 3244, 3245, 3246, 3247, 3248, 3249, 3250, 3251, 3252, 3253, 3254, 3255, 3256, 3257, 3258, 3259, 3260, 3261, 3262, 3263, 3264, 3265, 3266, 3267, 3268, 3269, 3270, 3271, 3272, 3273, 3274, 3275, 3276, 3277, 3278, 3279, 3280, 3281, 3282, 3283, 3284, 3285, 3286, 3287, 3288, 3289, 3290, 3291, 3292, 3293, 3294, 3295, 3296, 3297, 3298, 3299, 3300, 3301, 3302, 3303, 3304, 3305, 3306, 3307, 3308, 3309, 3310, 3311, 3312, 3313, 3314, 3315, 3316, 3317, 3318, 3319, 3320, 3321, 3322, 3323, 3324, 3325, 3326, 3327, 3328, 3329, 3330, 3331, 3332, 3333, 3334, 3335, 3336, 3337, 3338, 3339, 3340, 3341, 3342, 3343, 3344, 3345, 3346, 3347, 3348, 3349, 3350, 3351, 3352, 3353, 3354, 3355, 3356, 3357, 3358, 3359, 3360, 3361, 3362, 3363, 3364, 3365, 3366, 3367, 3368, 3369, 3370, 3371, 3372, 3373, 3374, 3375, 3376, 3377, 3378, 3379, 3380, 3381, 3382, 3383, 3384, 3385, 3386, 3387, 3388, 3389, 3390, 3391, 3392, 3393, 3394, 3395, 3396, 3397, 3398, 3399, 3400, 3401, 3402, 3403, 3404, 3405, 3406, 3407, 3408, 3409, 3410, 3411, 3412, 3413, 3414, 3415, 3416, 3417, 3418, 3419, 3420, 3421, 3422, 3423, 3424, 3425, 3426, 3427, 3428, 3429, 3430, 3431, 3432, 3433, 3434, 3435, 3436, 3437, 3438, 3439, 3440, 3441, 3442

, 3443, 3444, 3445, 3446, 3447, 3448, 3449, 3450, 3451, 3452, 3453, 3454, 3455, 3456, 3457, 3458, 3459, 3460, 34
61, 3462, 3463, 3464, 3465, 3466, 3467, 3468, 3469, 3470, 3471, 3472, 3473, 3474, 3475, 3476, 3477, 3478, 3479,
3480, 3481, 3482, 3483, 3484, 3485, 3486, 3487, 3488, 3489, 3490, 3491, 3492, 3493, 3494, 3495, 3496, 3497, 3498
, 3499, 3500, 3501, 3502, 3503, 3504, 3505, 3506, 3507, 3508, 3509, 3510, 3511, 3512, 3513, 3514, 3515, 3516, 35
17, 3518, 3519, 3520, 3521, 3522, 3523, 3524, 3525, 3526, 3527, 3528, 3529, 3530, 3531, 3532, 3533, 3534, 3535,
3536, 3537, 3538, 3539, 3540, 3541, 3542, 3543, 3544, 3545, 3546, 3547, 3548, 3549, 3550, 3551, 3552, 3553, 3554
, 3555, 3556, 3557, 3558, 3559, 3560, 3561, 3562, 3563, 3564, 3565, 3566, 3567, 3568, 3569, 3570, 3571, 3572, 35
73, 3574, 3575, 3576, 3577, 3578, 3579, 3580, 3581, 3582, 3583, 3584, 3585, 3586, 3587, 3588, 3589, 3590, 3591,
3592, 3593, 3594, 3595, 3596, 3597, 3598, 3599, 3600, 3601, 3602, 3603, 3604, 3605, 3606, 3607, 3608, 3609, 3610
, 3611, 3612, 3613, 3614, 3615, 3616, 3617, 3618, 3619, 3620, 3621, 3622, 3623, 3624, 3625, 3626, 3627, 3628, 36
29, 3630, 3631, 3632, 3633, 3634, 3635, 3636, 3637, 3638, 3639, 3640, 3641, 3642, 3643, 3644, 3645, 3646, 3647,
3648, 3649, 3650, 3651, 3652, 3653, 3654, 3655, 3656, 3657, 3658, 3659, 3660, 3661, 3662, 3663, 3664, 3665, 3666
, 3667, 3668, 3669, 3670, 3671, 3672, 3673, 3674, 3675, 3676, 3677, 3678, 3679, 3680, 3681, 3682, 3683, 3684, 36
85, 3686, 3687, 3688, 3689, 3690, 3691, 3692, 3693, 3694, 3695, 3696, 3697, 3698, 3699, 3700, 3701, 3702, 3703,
3704, 3705, 3706, 3707, 3708, 3709, 3710, 3711, 3712, 3713, 3714, 3715, 3716, 3717, 3718, 3719, 3720, 3721, 3722
, 3723, 3724, 3725, 3726, 3727, 3728, 3729, 3730, 3731, 3732, 3733, 3734, 3735, 3736, 3737, 3738, 3739, 3740, 37
41, 3742, 3743, 3744, 3745, 3746, 3747, 3748, 3749, 3750, 3751, 3752, 3753, 3754, 3755, 3756, 3757, 3758, 3759,
3760, 3761, 3762, 3763, 3764, 3765, 3766, 3767, 3768, 3769, 3770, 3771, 3772, 3773, 3774, 3775, 3776, 3777, 3778
, 3779, 3780, 3781, 3782, 3783, 3784, 3785, 3786, 3787, 3788, 3789, 3790, 3791, 3792, 3793, 3794, 3795, 3796, 37
97, 3798, 3799, 3800, 3801, 3802, 3803, 3804, 3805, 3806, 3807, 3808, 3809, 3810, 3811, 3812, 3813, 3814, 3815,
3816, 3817, 3818, 3819, 3820, 3821, 3822, 3823, 3824, 3825, 3826, 3827, 3828, 3829, 3830, 3831, 3832, 3833, 3834
, 3835, 3836, 3837, 3838, 3839, 3840, 3841, 3842, 3843, 3844, 3845, 3846, 3847, 3848, 3849, 3850, 3851, 3852, 38
53, 3854, 3855, 3856, 3857, 3858, 3859, 3860, 3861, 3862, 3863, 3864, 3865, 3866, 3867, 3868, 3869, 3870, 3871,
3872, 3873, 3874, 3875, 3876, 3877, 3878, 3879, 3880, 3881, 3882, 3883, 3884, 3885, 3886, 3887, 3888, 3889, 3890
, 3891, 3892, 3893, 3894, 3895, 3896, 3897, 3898, 3899, 3900, 3901, 3902, 3903, 3904, 3905, 3906, 3907, 3908, 39
09, 3910, 3911, 3912, 3913, 3914, 3915, 3916, 3917, 3918, 3919, 3920, 3921, 3922, 3923, 3924, 3925, 3926, 3927,
3928, 3929, 3930, 3931, 3932, 3933, 3934, 3935, 3936, 3937, 3938, 3939, 3940, 3941, 3942, 3943, 3944, 3945, 3946
, 3947, 3948, 3949, 3950, 3951, 3952, 3953, 3954, 3955, 3956, 3957, 3958, 3959, 3960, 3961, 3962, 3963, 3964, 39
65, 3966, 3967, 3968, 3969, 3970, 3971, 3972, 3973, 3974, 3975, 3976, 3977, 3978, 3979, 3980, 3981, 3982, 3983,
3984, 3985, 3986, 3987, 3988, 3989, 3990, 3991, 3992, 3993, 3994, 3995, 3996, 3997, 3998, 3999, 4000, 4001, 4002
, 4003, 4004, 4005, 4006, 4007, 4008, 4009, 4010, 4011, 4012, 4013, 4014, 4015, 4016, 4017, 4018, 4019, 4020, 40
21, 4022, 4023, 4024, 4025, 4026, 4027, 4028, 4029, 4030, 4031, 4032, 4033, 4034, 4035, 4036, 4037, 4038, 4039,
4040, 4041, 4042, 4043, 4044, 4045, 4046, 4047, 4048, 4049, 4050, 4051, 4052, 4053, 4054, 4055, 4056, 4057, 4058
, 4059, 4060, 4061, 4062, 4063, 4064, 4065, 4066, 4067, 4068, 4069, 4070, 4071, 4072, 4073, 4074, 4075, 4076, 40
77, 4078, 4079, 4080, 4081, 4082, 4083, 4084, 4085, 4086, 4087, 4088, 4089, 4090, 4091, 4092, 4093, 4094, 4095,
4096, 4097, 4098, 4099, 4100, 4101, 4102, 4103, 4104, 4105, 4106, 4107, 4108, 4109, 4110, 4111, 4112, 4113, 4114
, 4115, 4116, 4117, 4118, 4119, 4120, 4121, 4122, 4123, 4124, 4125, 4126, 4127, 4128, 4129, 4130, 4131, 4132, 41
33, 4134, 4135, 4136, 4137, 4138, 4139, 4140, 4141, 4142, 4143, 4144, 4145, 4146, 4147, 4148, 4149, 4150, 4151,
4152, 4153, 4154, 4155, 4156, 4157, 4158, 4159, 4160, 4161, 4162, 4163, 4164, 4165, 4166, 4167, 4168, 4169, 4170
, 4171, 4172, 4173, 4174, 4175, 4176, 4177, 4178, 4179, 4180, 4181, 4182, 4183, 4184, 4185, 4186, 4187, 4188, 41
89, 4190, 4191, 4192, 4193, 4194, 4195, 4196, 4197, 4198, 4199, 4200, 4201, 4202, 4203, 4204, 4205, 4206, 4207,
4208, 4209, 4210, 4211, 4212, 4213, 4214, 4215, 4216, 4217, 4218, 4219, 4220, 4221, 4222, 4223, 4224, 4225, 4226
, 4227, 4228, 4229, 4230, 4231, 4232, 4233, 4234, 4235, 4236, 4237, 4238, 4239, 4240, 4241, 4242, 4243, 4244, 42
45, 4246, 4247, 4248, 4249, 4250, 4251, 4252, 4253, 4254, 4255, 4256, 4257, 4258, 4259, 4260, 4261, 4262, 4263,
4264, 4265, 4266, 4267, 4268, 4269, 4270, 4271, 4272, 4273, 4274, 4275, 4276, 4277, 4278, 4279, 4280, 4281, 4282
, 4283, 4284, 4285, 4286, 4287, 4288, 4289, 4290, 4291, 4292, 4293, 4294, 4295, 4296, 4297, 4298, 4299, 4300, 43
01, 4302, 4303, 4304, 4305, 4306, 4307, 4308, 4309, 4310, 4311, 4312, 4313, 4314, 4315, 4316, 4317, 4318, 4319,
4320, 4321, 4322, 4323, 4324, 4325, 4326, 4327, 4328, 4329, 4330, 4331, 4332, 4333, 4334, 4335, 4336, 4337, 4338
, 4339, 4340, 4341, 4342, 4343, 4344, 4345, 4346, 4347, 4348, 4349, 4350, 4351, 4352, 4353, 4354, 4355, 4356, 43
57, 4358, 4359, 4360, 4361, 4362, 4363, 4364, 4365, 4366, 4367, 4368, 4369, 4370, 4371, 4372, 4373, 4374, 4375,
4376, 4377, 4378, 4379, 4380, 4381, 4382, 4383, 4384, 4385, 4386, 4387, 4388, 4389, 4390, 4391, 4392, 4393, 4394
, 4395, 4396, 4397, 4398, 4399, 4400, 4401, 4402, 4403, 4404, 4405, 4406, 4407, 4408, 4409, 4410, 4411, 4412, 44
13, 4414, 4415, 4416, 4417, 4418, 4419, 4420, 4421, 4422, 4423, 4424, 4425, 4426, 4427, 4428, 4429, 4430, 4431,
4432, 4433, 4434, 4435, 4436, 4437, 4438, 4439, 4440, 4441, 4442, 4443, 4444, 4445, 4446, 4447, 4448, 4449, 4450
, 4451, 4452, 4453, 4454, 4455, 4456, 4457, 4458, 4459, 4460, 4461, 4462, 4463, 4464, 4465, 4466, 4467, 4468, 44
69, 4470, 4471, 4472, 4473, 4474, 4475, 4476, 4477, 4478, 4479, 4480, 4481, 4482, 4483, 4484, 4485, 4486, 4487,
4488, 4489, 4490, 4491, 4492, 4493, 4494, 4495, 4496, 4497, 4498, 4499, 4500, 4501, 4502, 4503, 4504, 4505, 4506
, 4507, 4508, 4509, 4510, 4511, 4512, 4513, 4514, 4515, 4516, 4517, 4518, 4519, 4520, 4521, 4522, 4523, 4524, 45
25, 4526, 4527, 4528, 4529, 4530, 4531, 4532, 4533, 4534, 4535, 4536, 4537, 4538, 4539, 4540, 4541, 4542, 4543,
4544, 4545, 4546, 4547, 4548, 4549, 4550, 4551, 4552, 4553, 4554, 4555, 4556, 4557, 4558, 4559, 4560, 4561, 4562
, 4563, 4564, 4565, 4566, 4567, 4568, 4569, 4570, 4571, 4572, 4573, 4574, 4575, 4576, 4577, 4578, 4579, 4580, 45
81, 4582, 4583, 4584, 4585, 4586, 4587, 4588, 4589, 4590, 4591, 4592, 4593, 4594, 4595, 4596, 4597, 4598, 4599,
4600, 4601, 4602, 4603, 4604, 4605, 4606, 4607, 4608, 4609, 4610, 4611, 4612, 4613, 4614, 4615, 4616, 4617, 4618
, 4619, 4620, 4621, 4622, 4623, 4624, 4625, 4626, 4627, 4628, 4629, 4630, 4631, 4632, 4633, 4634, 4635, 4636, 46
37, 4638, 4639, 4640, 4641, 4642, 4643, 4644, 4645, 4646, 4647, 4648, 4649, 4650, 4651, 4652, 4653, 4654, 4655,
4656, 4657, 4658, 4659, 4660, 4661, 4662, 4663, 4664, 4665, 4666, 4667, 4668, 4669, 4670, 4671, 4672, 4673, 4674
, 4675, 4676, 4677, 4678, 4679, 4680, 4681, 4682, 4683, 4684, 4685, 4686, 4687, 4688, 4689, 4690, 4691, 4692, 46
93, 4694, 4695, 4696, 4697, 4698, 4699, 4700, 4701, 4702, 4703, 4704, 4705, 4706, 4707, 4708, 4709, 4710, 4711,
4712, 4713, 4714, 4715, 4716, 4717, 4718, 4719, 4720, 4721, 4722, 4723, 4724, 4725, 4726, 4727, 4728, 4729, 4730
, 4731, 4732, 4733, 4734, 4735, 4736, 4737, 4738, 4739, 4740, 4741, 4742, 4743, 4744, 4745, 4746, 4747, 4748, 47
49, 4750, 4751, 4752, 4753, 4754, 4755, 4756, 4757, 4758, 4759, 4760, 4761, 4762, 4763, 4764, 4765, 4766, 4767,
4768, 4769, 4770, 4771, 4772, 4773, 4774, 4775, 4776, 4777, 4778, 4779, 4780, 4781, 4782, 4783, 4784, 4785, 4786
, 4787, 4788, 4789, 4790, 4791, 4792, 4793, 4794, 4795, 4796, 4797, 4798, 4799, 4800, 4801, 4802, 4803, 4804, 48
05, 4806, 4807, 4808, 4809, 4810, 4811, 4812, 4813, 4814, 4815, 4816, 4817, 4818, 4819, 4820, 4821, 4822, 4823,
4824, 4825, 4826, 4827, 4828, 4829, 4830, 4831, 4832, 4833, 4834, 4835, 4836, 4837, 4838, 4839, 4840, 4841, 4842
, 4843, 4844, 4845, 4846, 4847, 4848, 4849, 4850, 4851, 4852, 4853, 4854, 4855, 4856, 4857, 4858, 4859, 4860, 48
61, 4862, 4863, 4864, 4865, 4866, 4867, 4868, 4869, 4870, 4871, 4872, 4873, 4874, 4875, 4876, 4877, 4878, 4879,
4880, 4881, 4882, 4883, 4884, 4885, 4886, 4887, 4888, 4889, 4890, 4891, 4892, 4893, 4894, 4895, 4896, 4897, 4898
, 4899, 4900, 4901, 4902, 4903, 4904, 4905, 4906, 4907, 4908, 4909, 4910, 4911, 4912, 4913, 4914, 4915, 4916, 49
17, 4918, 4919, 4920, 4921, 4922, 4923, 4924, 4925, 4926, 4927, 4928, 4929, 4930, 4931, 4932, 4933, 4934, 4935,
4936, 4937, 4938, 4939, 4940, 4941, 4942, 4943, 4944, 4945, 4946, 4947, 4948, 4949, 4950, 4951, 4952, 4953, 4954
, 4955, 4956, 4957, 4958, 4959, 4960, 4961, 4962, 4963, 4964, 4965, 4966, 4967, 4968, 4969, 4970, 4971, 4972, 49
73, 4974, 4975, 4976, 4977, 4978, 4979, 4980, 4981, 4982, 4983, 4984, 4985, 4986, 4987, 4988, 4989, 4990, 4991,

```
4992, 4993, 4994, 4995, 4996, 4997, 4998, 4999, 5000, 5001, 5002, 5003, 5004, 5005, 5006, 5007, 5008, 5009, 5010
, 5011, 5012, 5013, 5014, 5015, 5016, 5017, 5018, 5019, 5020, 5021, 5022, 5023, 5024, 5025, 5026, 5027, 5028, 50
29, 5030, 5031, 5032, 5033, 5034, 5035, 5036, 5037, 5038, 5039, 5040, 5041, 5042, 5043, 5044, 5045, 5046, 5047,
5048, 5049, 5050, 5051, 5052, 5053, 5054, 5055, 5056, 5057, 5058, 5059, 5060, 5061, 5062, 5063, 5064, 5065, 5066
, 5067, 5068, 5069, 5070, 5071, 5072, 5073, 5074, 5075, 5076, 5077, 5078, 5079, 5080, 5081, 5082, 5083, 5084, 50
85, 5086, 5087, 5088, 5089, 5090, 5091, 5092, 5093, 5094, 5095, 5096, 5097, 5098, 5099, 5100, 5101, 5102, 5103,
5104, 5105, 5106, 5107, 5108, 5109, 5110, 5111, 5112, 5113, 5114, 5115, 5116, 5117, 5118, 5119, 5120, 5121, 5122
, 5123, 5124, 5125, 5126, 5127, 5128, 5129, 5130, 5131, 5132, 5133, 5134, 5135, 5136, 5137, 5138, 5139, 5140, 51
41, 5142, 5143, 5144, 5145, 5146, 5147, 5148, 5149, 5150, 5151, 5152, 5153, 5154, 5155, 5156, 5157, 5158, 5159,
5160, 5161, 5162, 5163, 5164, 5165, 5166, 5167, 5168, 5169, 5170, 5171, 5172, 5173, 5174, 5175, 5176, 5177, 5178
, 5179, 5180, 5181, 5182, 5183, 5184, 5185, 5186, 5187, 5188, 5189, 5190, 5191, 5192, 5193, 5194, 5195, 5196, 51
97, 5198, 5199, 5200, 5201, 5202, 5203, 5204, 5205, 5206, 5207, 5208, 5209, 5210, 5211, 5212, 5213, 5214, 5215,
5216, 5217, 5218, 5219, 5220, 5221, 5222, 5223, 5224, 5225, 5226, 5227, 5228, 5229, 5230, 5231, 5232, 5233, 5234
, 5235, 5236, 5237, 5238, 5239, 5240, 5241, 5242, 5243, 5244, 5245, 5246, 5247, 5248, 5249, 5250, 5251, 5252, 52
53, 5254, 5255, 5256, 5257, 5258, 5259, 5260, 5261, 5262, 5263, 5264, 5265, 5266, 5267, 5268, 5269, 5270, 5271,
5272, 5273, 5274, 5275, 5276, 5277, 5278, 5279, 5280, 5281, 5282, 5283, 5284, 5285, 5286, 5287, 5288, 5289, 5290
, 5291, 5292, 5293, 5294, 5295, 5296, 5297, 5298, 5299, 5300, 5301, 5302, 5303, 5304, 5305, 5306, 5307, 5308, 53
09, 5310, 5311, 5312, 5313, 5314, 5315, 5316, 5317, 5318, 5319, 5320, 5321, 5322, 5323, 5324, 5325, 5326, 5327,
5328, 5329, 5330, 5331, 5332, 5333, 5334, 5335, 5336, 5337, 5338, 5339, 5340, 5341, 5342, 5343, 5344, 5345, 5346
, 5347, 5348, 5349, 5350, 5351, 5352, 5353, 5354, 5355, 5356, 5357, 5358, 5359, 5360, 5361, 5362, 5363, 5364, 53
65, 5366, 5367, 5368, 5369, 5370, 5371, 5372, 5373, 5374, 5375, 5376, 5377, 5378, 5379, 5380, 5381, 5382, 5383,
5384, 5385, 5386, 5387, 5388, 5389, 5390, 5391, 5392, 5393, 5394, 5395, 5396, 5397, 5398, 5399, 5400, 5401, 5402
, 5403, 5404, 5405, 5406, 5407, 5408, 5409, 5410, 5411, 5412, 5413, 5414, 5415, 5416, 5417, 5418, 5419, 5420, 54
21, 5422, 5423, 5424, 5425, 5426, 5427, 5428, 5429, 5430, 5431, 5432, 5433, 5434, 5435, 5436, 5437, 5438, 5439,
5440, 5441, 5442, 5443, 5444, 5445, 5446, 5447, 5448, 5449, 5450, 5451, 5452, 5453, 5454, 5455, 5456, 5457, 5458
, 5459, 5460, 5461, 5462, 5463, 5464, 5465, 5466, 5467, 5468, 5469, 5470, 5471, 5472, 5473, 5474, 5475, 5476, 54
77, 5478, 5479, 5480, 5481, 5482, 5483, 5484, 5485, 5486, 5487, 5488, 5489, 5490, 5491, 5492, 5493, 5494, 5495,
5496, 5497, 5498, 5499, 5500, 5501, 5502, 5503, 5504, 5505, 5506, 5507, 5508, 5509, 5510, 5511, 5512, 5513, 5514
, 5515, 5516, 5517, 5518, 5519, 5520, 5521, 5522, 5523, 5524, 5525, 5526, 5527, 5528, 5529, 5530, 5531, 5532, 55
33, 5534, 5535, 5536, 5537, 5538, 5539, 5540, 5541, 5542, 5543, 5544, 5545, 5546, 5547, 5548, 5549, 5550, 5551,
5552, 5553, 5554, 5555, 5556, 5557, 5558, 5559, 5560, 5561, 5562, 5563, 5564, 5565, 5566, 5567, 5568, 5569, 5570
, 5571, 5572, 5573, 5574, 5575, 5576, 5577, 5578, 5579, 5580, 5581, 5582, 5583, 5584, 5585, 5586, 5587, 5588, 55
89, 5590, 5591, 5592, 5593, 5594, 5595, 5596, 5597, 5598, 5599, 5600, 5601, 5602, 5603, 5604, 5605, 5606, 5607,
5608, 5609, 5610, 5611, 5612, 5613, 5614, 5615, 5616, 5617, 5618, 5619, 5620, 5621, 5622, 5623, 5624, 5625, 5626
, 5627, 5628, 5629, 5630, 5631, 5632, 5633, 5634, 5635, 5636, 5637, 5638, 5639, 5640, 5641, 5642, 5643, 5644, 56
45, 5646, 5647, 5648, 5649, 5650, 5651, 5652, 5653, 5654, 5655, 5656, 5657, 5658, 5659, 5660, 5661, 5662, 5663,
5664, 5665, 5666, 5667, 5668, 5669, 5670, 5671, 5672, 5673, 5674, 5675, 5676, 5677, 5678, 5679, 5680, 5681, 5682
, 5683, 5684, 5685, 5686, 5687, 5688, 5689, 5690, 5691, 5692, 5693, 5694, 5695, 5696, 5697, 5698, 5699, 5700, 57
01, 5702, 5703, 5704, 5705, 5706, 5707, 5708, 5709, 5710, 5711, 5712, 5713, 5714, 5715, 5716, 5717, 5718, 5719,
5720, 5721, 5722, 5723, 5724, 5725, 5726, 5727, 5728, 5729, 5730, 5731, 5732, 5733, 5734, 5735, 5736, 5737, 5738
, 5739, 5740, 5741, 5742, 5743, 5744, 5745, 5746, 5747, 5748, 5749, 5750, 5751, 5752, 5753, 5754, 5755, 5756, 57
57, 5758, 5759, 5760, 5761, 5762, 5763, 5764, 5765, 5766, 5767, 5768, 5769, 5770, 5771, 5772, 5773, 5774, 5775,
5776, 5777, 5778, 5779, 5780, 5781, 5782, 5783, 5784, 5785, 5786, 5787, 5788, 5789, 5790, 5791, 5792, 5793, 5794
, 5795, 5796, 5797, 5798, 5799, 5800, 5801, 5802, 5803, 5804, 5805, 5806, 5807, 5808, 5809, 5810, 5811, 5812, 58
13, 5814, 5815, 5816, 5817, 5818, 5819, 5820, 5821, 5822, 5823, 5824, 5825, 5826, 5827, 5828, 5829, 5830, 5831,
5832, 5833, 5834, 5835, 5836, 5837, 5838, 5839, 5840, 5841, 5842, 5843, 5844, 5845, 5846, 5847, 5848, 5849, 5850
, 5851, 5852, 5853, 5854, 5855, 5856, 5857, 5858, 5859, 5860, 5861, 5862, 5863, 5864, 5865, 5866, 5867, 5868, 58
69, 5870, 5871, 5872, 5873, 5874, 5875, 5876, 5877, 5878, 5879, 5880, 5881, 5882, 5883, 5884, 5885, 5886, 5887,
5888, 5889, 5890, 5891, 5892, 5893, 5894, 5895, 5896, 5897, 5898, 5899, 5900, 5901, 5902, 5903, 5904, 5905, 5906
, 5907, 5908, 5909, 5910, 5911, 5912, 5913, 5914, 5915, 5916, 5917, 5918, 5919, 5920, 5921, 5922, 5923, 5924, 59
25, 5926, 5927, 5928, 5929, 5930, 5931, 5932, 5933, 5934, 5935, 5936, 5937, 5938, 5939, 5940, 5941, 5942, 5943,
5944, 5945, 5946, 5947, 5948, 5949, 5950, 5951, 5952, 5953, 5954, 5955, 5956, 5957, 5958, 5959, 5960, 5961, 5962
, 5963, 5964, 5965, 5966, 5967, 5968, 5969, 5970, 5971, 5972, 5973, 5974, 5975, 5976, 5977, 5978, 5979, 5980, 59
81, 5982, 5983, 5984, 5985, 5986, 5987, 5988, 5989, 5990, 5991, 5992, 5993, 5994, 5995, 5996, 5997, 5998, 5999,
6000, 6001, 6002, 6003, 6004, 6005, 6006, 6007, 6008, 6009, 6010, 6011, 6012, 6013, 6014, 6015, 6016, 6017, 6018
, 6019, 6020, 6021, 6022, 6023, 6024, 6025, 6026, 6027, 6028, 6029, 6030, 6031, 6032, 6033, 6034, 6035, 6036, 60
37, 6038, 6039, 6040, 6041, 6042, 6043, 6044, 6045, 6046, 6047, 6048, 6049, 6050, 6051, 6052, 6053, 6054, 6055,
6056, 6057, 6058, 6059, 6060, 6061, 6062, 6063, 6064, 6065, 6066, 6067, 6068, 6069, 6070, 6071, 6072, 6073, 6074
, 6075, 6076, 6077, 6078, 6079, 6080, 6081, 6082, 6083, 6084, 6085, 6086, 6087, 6088, 6089, 6090, 6091, 6092, 60
93, 6094, 6095, 6096, 6097, 6098, 6099, 6100, 6101, 6102, 6103, 6104, 6105, 6106, 6107, 6108, 6109, 6110, 6111,
6112, 6113, 6114, 6115, 6116, 6117, 6118, 6119, 6120, 6121, 6122, 6123, 6124, 6125, 6126, 6127, 6128, 6129, 6130
, 6131, 6132, 6133, 6134, 6135, 6136, 6137, 6138, 6139, 6140, 6141, 6142, 6143, 6144, 6145, 6146, 6147, 6148, 61
49, 6150, 6151, 6152, 6153, 6154, 6155, 6156, 6157, 6158, 6159, 6160, 6161, 6162, 6163, 6164, 6165, 6166, 6167,
6168, 6169, 6170, 6171, 6172, 6173, 6174, 6175, 6176, 6177, 6178, 6179, 6180, 6181, 6182, 6183, 6184, 6185, 6186
, 6187, 6188, 6189, 6190, 6191, 6192, 6193, 6194, 6195, 6196, 6197, 6198, 6199, 6200, 6201, 6202, 6203, 6204, 62
05, 6206, 6207, 6208, 6209, 6210, 6211, 6212, 6213, 6214, 6215, 6216, 6217, 6218, 6219, 6220, 6221, 6222, 6223,
6224, 6225, 6226, 6227, 6228, 6229, 6230, 6231, 6232, 6233, 6234, 6235, 6236, 6237, 6238, 6239, 6240, 6241, 6242
, 6243, 6244, 6245, 6246, 6247, 6248, 6249, 6250, 6251, 6252, 6253, 6254, 6255, 6256, 6257, 6258, 6259, 6260, 62
61, 6262, 6263, 6264, 6265, 6266, 6267, 6268, 6269, 6270, 6271, 6272, 6273, 6274, 6275, 6276, 6277, 6278, 6279,
6280, 6281, 6282, 6283, 6284, 6285, 6286, 6287, 6288, 6289, 6290, 6291, 6292, 6293, 6294, 6295, 6296, 6297, 6298
, 6299, 6300], [588, 589, 590, 591, 592, 593, 594, 595, 596, 597, 598], [1683, 1684]]
```

3. Applying the Louvain Algorithm

In [5]:

```python
import networkx as nx
import community as community_louvain

# Apply Louvain algorithm
partition_louvain = community_louvain.best_partition(graph)

# Count the size of each community
```

```python
community_sizes = {}
for node, community_id in partition_louvain.items():
    community_sizes.setdefault(community_id, 0)
    community_sizes[community_id] += 1

# Sort communities by size
sorted_communities = sorted(community_sizes.items(), key=lambda x: x[1], reverse=True)

# Get the top 100 communities
top_100_communities = sorted_communities[:100]

print("Top 100 Louvain communities:")
for i, (community_id, size) in enumerate(top_100_communities):
    nodes_in_community = [node for node, comm_id in partition_louvain.items() if comm_id == community_id]
    print(f"Community {i+1}: Size {size}, Nodes: {nodes_in_community}")
```

```
Top 100 Louvain communities:
Community 1: Size 932, Nodes: [258, 1903, 852, 4953, 975, 1918, 1921, 28, 29, 1511, 1913, 846, 1516, 1517, 2630,
1701, 43, 1926, 1928, 1604, 1930, 1931, 1933, 1935, 1936, 3998, 1947, 534, 1123, 219, 346, 1957, 1969, 1971, 106
7, 107, 363, 1240, 2034, 2060, 2075, 2078, 1831, 253, 2103, 2984, 842, 5940, 2108, 2114, 169, 531, 818, 2118, 21
19, 2121, 178, 3066, 5325, 255, 256, 554, 871, 2125, 2126, 2128, 2129, 1489, 182, 2145, 208, 209, 210, 212, 215,
217, 976, 2430, 2431, 228, 232, 233, 234, 2221, 2223, 2196, 612, 2200, 2211, 2213, 254, 257, 259, 1137, 1302, 13
04, 269, 2229, 2234, 275, 280, 283, 1202, 2235, 779, 838, 1571, 1892, 2236, 1573, 315, 330, 2274, 376, 1888, 415
2, 1755, 340, 342, 347, 1805, 2301, 2302, 2303, 2306, 2307, 2292, 2300, 356, 2308, 572, 980, 1212, 2322, 2323, 2
332, 2337, 1234, 388, 1538, 2381, 977, 4254, 2470, 2403, 2408, 940, 2409, 2413, 2414, 433, 437, 1824, 2433, 1090
, 2500, 456, 2444, 2445, 450, 451, 452, 453, 454, 457, 458, 459, 461, 462, 465, 468, 477, 1492, 4207, 4282, 2486
, 478, 482, 483, 486, 488, 1812, 532, 2497, 2475, 2474, 577, 4876, 4979, 571, 573, 576, 1329, 2499, 547, 2506, 2
511, 2514, 2515, 3312, 4477, 4515, 562, 570, 721, 2521, 2522, 2523, 979, 1353, 4318, 4319, 2520, 4846, 5235, 523
8, 5239, 1610, 2525, 2526, 2528, 2535, 2543, 2554, 796, 644, 645, 2563, 2564, 1023, 660, 1084, 669, 1844, 1845,
690, 2604, 2612, 2617, 706, 4623, 4624, 1803, 2645, 2646, 2653, 2657, 2650, 2659, 2663, 2678, 2680, 1618, 2690,
2694, 1335, 2705, 789, 801, 2724, 2725, 2726, 2727, 2732, 833, 849, 853, 854, 938, 2749, 2752, 2745, 1146, 1655,
884, 1671, 902, 1272, 2780, 2782, 2783, 913, 2788, 923, 925, 926, 927, 939, 945, 2846, 4409, 4410, 1570, 2828, 2
840, 2843, 4892, 1777, 2847, 2871, 969, 972, 2862, 2863, 978, 2865, 2866, 1140, 2877, 2878, 2882, 1320, 988, 288
7, 2896, 994, 995, 2936, 2909, 1010, 2920, 2923, 2924, 2918, 2938, 1798, 2958, 2991, 2976, 2977, 2973, 1085, 108
6, 1088, 2985, 1568, 3111, 1115, 2997, 1127, 1136, 1138, 1139, 1141, 3003, 3005, 3008, 3015, 3016, 3017, 3012, 3
013, 3014, 3024, 3027, 3180, 3037, 3038, 1882, 1175, 3781, 1382, 3063, 3105, 3079, 3905, 1881, 3095, 3097, 1237,
1238, 1239, 1241, 1242, 1243, 1244, 3106, 3107, 3108, 3120, 1281, 3153, 3133, 3135, 3136, 1308, 3181, 3182, 3183
, 1530, 3185, 1359, 1360, 3206, 3207, 3209, 3214, 3217, 1817, 1437, 3235, 4825, 5241, 5428, 5487, 5786, 5787, 57
88, 1396, 3244, 3247, 3266, 3254, 3259, 3260, 1407, 3275, 3284, 4970, 3295, 1456, 3321, 1527, 1528, 3331, 3333,
4359, 1544, 3340, 1636, 3374, 1569, 1576, 3408, 1599, 1613, 1614, 1615, 1630, 1633, 1634, 1635, 1637, 1638, 1639
, 1640, 3389, 3390, 3391, 3392, 3393, 3387, 3395, 3396, 4803, 3419, 3422, 3423, 3428, 3429, 1687, 3440, 3441, 34
42, 3451, 3463, 3460, 3461, 3469, 3479, 3480, 1721, 3476, 3496, 3566, 1753, 1756, 1759, 1761, 3524, 1770, 3700,
1794, 1795, 1796, 1799, 1804, 1807, 1809, 1810, 3549, 1815, 1819, 3564, 3567, 3558, 1821, 1823, 1825, 1826, 3568
, 3585, 3586, 3588, 3589, 3594, 3601, 1862, 3799, 3800, 3802, 3803, 3806, 3676, 1877, 1878, 1879, 1880, 3629, 36
33, 3634, 3635, 3624, 3625, 3628, 1893, 3684, 3696, 3699, 3704, 3705, 3706, 3707, 3716, 3717, 3727, 3728, 3729,
3730, 3731, 3733, 3754, 3755, 3739, 3740, 3772, 4438, 3815, 3826, 3989, 3886, 3901, 3902, 3903, 5343, 3919, 3930
, 3942, 3944, 4067, 5909, 3982, 3983, 3984, 3995, 4012, 4014, 4015, 4017, 4018, 4019, 4020, 4031, 4033, 4036, 40
37, 4038, 4060, 4061, 4068, 4074, 4095, 4104, 4106, 5450, 4124, 4127, 4128, 4136, 4131, 4142, 4158, 4166, 4168,
4169, 4210, 4215, 4237, 4426, 5289, 4240, 4241, 4245, 4247, 4249, 4251, 4253, 4255, 4258, 4259, 4261, 4268, 4288
, 4294, 4300, 4312, 4324, 4325, 4335, 4339, 4368, 4342, 4343, 4365, 4370, 4414, 4416, 4424, 4434, 4443, 4446, 44
47, 4448, 4444, 4452, 5324, 4460, 4462, 4468, 4469, 4467, 4480, 4471, 4492, 4521, 4522, 4523, 4524, 4525, 4526,
4527, 4528, 4529, 4530, 4531, 4542, 4543, 4551, 4552, 4563, 4587, 4588, 4589, 4591, 4620, 4621, 4622, 4626, 4627
, 4634, 4638, 4639, 4654, 4655, 4656, 4663, 4667, 4670, 4675, 4677, 4686, 4723, 4725, 4731, 4732, 4733, 4753, 47
56, 4757, 4839, 4840, 4842, 4843, 4760, 4767, 4770, 4773, 4774, 4779, 4780, 4872, 4802, 4799, 4827, 4823, 4826,
4836, 4837, 4847, 4848, 4849, 4850, 4851, 4857, 4873, 4875, 4885, 4888, 4889, 4899, 4901, 5823, 4910, 4911, 4915
, 4921, 4923, 4924, 4925, 5089, 5852, 4937, 5155, 4977, 4978, 4982, 4983, 4992, 4996, 5003, 5004, 5005, 5006, 50
07, 5088, 5039, 5041, 5035, 5052, 5069, 5060, 5061, 5062, 5075, 5081, 5087, 5092, 5099, 5101, 5113, 5151, 5160,
5169, 5166, 5182, 5196, 5202, 5203, 5204, 5206, 5208, 5210, 5212, 5214, 5215, 5216, 5231, 5253, 5254, 5269, 5280
, 5276, 5274, 5275, 5287, 5284, 5285, 5286, 5290, 5297, 5300, 5309, 5321, 5322, 5349, 5357, 5367, 5399, 5400, 54
01, 5402, 5406, 5410, 5954, 5955, 5421, 5422, 5440, 5442, 5443, 5456, 5447, 5448, 5474, 5464, 5499, 5507, 5508,
5513, 5514, 5523, 5524, 5525, 5526, 5530, 5535, 5555, 5556, 5587, 5591, 5592, 5593, 5594, 5595, 5596, 5597, 5598
, 5606, 5608, 5609, 5610, 5618, 5619, 5623, 5627, 5641, 5658, 5659, 5675, 5695, 5696, 5697, 5702, 5704, 5703, 57
14, 5723, 5728, 5729, 5725, 5731, 5732, 5747, 5748, 5752, 5756, 5759, 5775, 5776, 5781, 5782, 5816, 5801, 5803,
5810, 5811, 5818, 5825, 5826, 5835, 5858, 5856, 5857, 5886, 5898, 5904, 5905, 5906, 5907, 5908, 5911, 5933, 5934
, 5936, 5975, 5952, 5953, 5968, 5969, 5970, 6008, 6009, 6010, 5980, 6063, 6064, 6065, 6080, 6081, 6090, 6093, 60
96, 6102, 6124, 6165, 6172, 6173, 6280, 6281, 6196, 6193, 6183, 6246, 6247, 6248, 6258, 6271, 6272, 6273, 6277,
6278, 6290]
Community 2: Size 877, Nodes: [1287, 491, 1050, 2254, 3169, 1916, 962, 2201, 38, 725, 1376, 2002, 2167, 2168, 21
69, 53, 57, 59, 1950, 1980, 68, 3956, 152, 965, 1953, 1955, 1958, 1975, 78, 1157, 1894, 2204, 93, 2010, 2023, 20
24, 2035, 2036, 2050, 510, 1089, 2062, 222, 2083, 130, 134, 136, 138, 139, 140, 2086, 2087, 1096, 1598, 2089, 20
90, 2091, 156, 1688, 2117, 4539, 1524, 192, 193, 194, 2388, 2389, 2390, 2391, 1020, 2135, 2136, 2137, 2139, 2140
, 202, 412, 2147, 2148, 2152, 213, 2432, 2164, 224, 893, 2179, 241, 2186, 2187, 2195, 1068, 1070, 918, 2209, 147
5, 1303, 2225, 2227, 2228, 552, 790, 1371, 712, 307, 2253, 587, 2257, 883, 2260, 2262, 2263, 2264, 2266, 325, 32
7, 329, 2269, 2270, 2271, 2272, 2278, 627, 681, 2273, 2275, 2276, 335, 1052, 2319, 3590, 350, 2305, 2328, 2310,
2326, 1030, 1231, 1232, 381, 382, 383, 384, 386, 387, 389, 2346, 2347, 2349, 2351, 609, 1077, 2360, 2361, 2362,
2363, 2353, 2354, 2355, 2356, 882, 1383, 2358, 2359, 944, 2385, 2375, 5502, 1541, 2395, 2398, 2399, 2411, 2412,
438, 487, 1370, 1522, 2443, 2446, 878, 466, 2447, 2449, 2451, 3205, 480, 722, 726, 1048, 2462, 2493, 489, 490, 4
92, 493, 495, 497, 498, 2466, 892, 1372, 500, 508, 509, 511, 515, 828, 2488, 2489, 521, 582, 583, 4248, 540, 179
0, 550, 2517, 3141, 821, 560, 563, 564, 565, 566, 567, 569, 2582, 2584, 584, 2527, 1782, 715, 2536, 2559, 2562,
617, 618, 619, 622, 623, 624, 625, 626, 634, 2941, 3112, 5880, 6131, 6133, 668, 638, 654, 732, 2570, 2571, 1032,
2578, 670, 672, 673, 674, 675, 677, 679, 682, 684, 685, 687, 688, 851, 2585, 2586, 2631, 2632, 2590, 1690, 2593,
2596, 1441, 1589, 2600, 2601, 1056, 708, 1047, 1107, 2647, 2648, 2649, 719, 723, 724, 2713, 1313, 1434, 3726, 40
72, 5139, 728, 729, 730, 733, 734, 735, 928, 1435, 2651, 2652, 2684, 2685, 742, 2661, 2662, 2666, 2681, 2682, 26
83, 2669, 760, 2687, 2688, 1689, 2693, 787, 788, 791, 792, 793, 794, 1224, 2735, 2736, 2737, 886, 858, 1705, 170
```

8, 867, 2746, 2766, 2769, 2758, 879, 2774, 887, 888, 890, 891, 2775, 2959, 904, 2781, 2792, 2793, 2795, 2808, 28
18, 2820, 929, 932, 934, 935, 936, 958, 1249, 4123, 4973, 4974, 1778, 1779, 1781, 1783, 1054, 2844, 2872, 2874,
2876, 2868, 1318, 998, 1536, 2898, 2899, 2931, 2911, 2921, 2925, 2939, 1316, 2945, 2947, 2949, 1034, 4264, 1051,
1053, 1060, 1063, 2987, 2988, 2990, 2992, 2995, 2972, 2974, 2975, 2981, 1374, 3110, 3019, 3022, 1133, 3158, 3006
, 3033, 1158, 3039, 3041, 1173, 3865, 5009, 3062, 1203, 3098, 1209, 1210, 1214, 1215, 1222, 1225, 1226, 1229, 31
56, 3085, 3086, 3087, 3089, 3090, 3091, 3092, 3093, 3094, 3096, 1247, 1248, 1250, 1253, 3119, 1276, 1278, 3154,
3155, 3157, 1299, 3132, 3139, 3140, 1307, 1312, 1314, 1315, 3159, 3161, 3162, 3173, 3187, 3188, 3212, 1373, 4170
, 3229, 3252, 3231, 1401, 3245, 1421, 1423, 1436, 1438, 3291, 3293, 3294, 1718, 3298, 3311, 3319, 1520, 1521, 15
23, 1525, 3328, 3347, 3343, 3375, 3352, 3355, 3358, 1577, 1578, 3364, 3365, 3366, 3367, 3368, 1619, 1626, 3379,
3380, 3381, 3405, 3411, 3572, 1685, 1686, 3466, 1719, 3489, 1740, 3498, 3499, 1743, 3502, 3509, 1757, 3531, 3532
, 1793, 3541, 1801, 3560, 1839, 3591, 3592, 3580, 3576, 3613, 3606, 3804, 1872, 3621, 3622, 3623, 1883, 3630, 36
31, 3627, 3636, 3671, 3721, 3735, 3737, 3749, 3779, 3782, 3798, 3819, 3822, 3823, 3824, 3825, 3829, 3988, 3842,
3845, 3847, 3848, 3864, 3866, 3863, 3861, 3888, 3904, 3891, 3892, 3926, 3917, 3928, 3929, 3931, 3932, 3935, 3936
, 3949, 3950, 4501, 4738, 3952, 3955, 3957, 3958, 3960, 3961, 3969, 3970, 3966, 3977, 3974, 3975, 3985, 4010, 41
30, 5056, 4034, 4035, 4056, 6168, 6232, 4053, 4054, 4055, 4088, 4045, 4046, 4047, 4048, 4049, 4057, 4058, 4073,
4078, 4079, 4087, 4085, 4086, 4089, 4090, 4091, 4094, 4129, 4134, 5600, 4182, 4184, 4173, 4222, 4223, 4224, 4204
, 4186, 4189, 4190, 4193, 4194, 4195, 4198, 4209, 4232, 5417, 4244, 4250, 4868, 5806, 5807, 5808, 4262, 4291, 43
08, 4317, 4344, 4341, 4383, 4391, 4413, 4418, 4420, 4421, 4436, 4437, 4440, 4442, 4450, 5179, 4459, 4464, 4599,
4484, 4485, 5001, 5403, 4505, 4507, 4509, 4512, 4513, 4514, 4517, 4519, 4520, 4533, 4534, 4541, 5931, 6071, 4564
, 4565, 4568, 4583, 4629, 6038, 4665, 4666, 4669, 4694, 4696, 4697, 4698, 4699, 4706, 4717, 4718, 4736, 4737, 47
50, 4838, 4765, 4766, 4778, 4805, 4821, 4822, 4845, 4856, 4867, 4869, 4890, 4896, 4902, 4904, 4913, 4931, 4940,
4941, 4943, 4960, 5264, 5337, 4963, 4964, 4965, 4966, 4967, 4968, 4969, 5002, 5051, 5016, 5017, 5021, 5023, 5025
, 5046, 5047, 5068, 5057, 5072, 5070, 5071, 5074, 5085, 5110, 5119, 5120, 5123, 5132, 5150, 5053, 5190, 5191, 51
95, 5205, 5226, 5228, 5271, 5279, 5302, 5303, 5336, 5572, 5884, 5344, 5348, 5350, 5354, 5355, 5356, 5358, 5382,
5383, 5391, 5393, 5412, 5413, 5415, 5416, 5430, 5431, 5432, 5433, 5463, 5469, 5470, 5485, 5486, 5506, 5512, 5521
, 5881, 5531, 5544, 5553, 5569, 5557, 5558, 5579, 5604, 5643, 5670, 5692, 5693, 5694, 5700, 5750, 5755, 5821, 58
22, 5812, 5813, 5817, 5831, 5836, 5838, 5902, 5903, 5918, 5924, 5930, 5932, 5963, 5982, 5992, 5997, 5998, 5999,
6023, 6012, 6014, 6016, 6017, 6019, 6020, 6021, 6040, 6041, 6049, 6069, 6097, 6110, 6126, 6128, 6129, 6130, 6134
, 6135, 6136, 6163, 6140, 6141, 6142, 6167, 6250, 6251, 6195, 6229, 6233, 6253, 6254, 6263]
Community 3: Size 569, Nodes: [0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 1897, 144, 1418, 1902, 121, 127, 128, 179, 247,
249, 264, 353, 424, 426, 145, 176, 177, 753, 754, 762, 2064, 3002, 665, 1394, 1786, 1904, 1905, 1907, 124, 147,
246, 248, 250, 251, 252, 15, 17, 20, 123, 129, 143, 367, 427, 718, 3893, 368, 717, 856, 1908, 1909, 30, 31, 126,
174, 1199, 36, 1929, 1934, 101, 1992, 1948, 83, 148, 351, 946, 2000, 2001, 369, 422, 559, 2018, 820, 2020, 2037,
2038, 2043, 113, 2032, 1561, 2066, 2046, 2047, 1317, 2052, 2053, 2056, 120, 122, 125, 1428, 146, 390, 423, 2063,
132, 2071, 2072, 2073, 696, 698, 755, 947, 2077, 175, 1246, 2096, 629, 2084, 141, 142, 149, 150, 2098, 265, 667,
5338, 5942, 1245, 238, 155, 167, 172, 266, 173, 180, 3849, 697, 2122, 2130, 2132, 183, 195, 558, 736, 2146, 1412
, 2178, 314, 1564, 2176, 2217, 2182, 2183, 2185, 2193, 1284, 1073, 1074, 915, 917, 919, 920, 983, 4181, 586, 220
6, 2208, 2210, 263, 2224, 1259, 311, 2243, 298, 352, 666, 378, 1704, 2252, 909, 2261, 339, 2286, 343, 345, 2290,
751, 930, 855, 358, 2312, 2313, 366, 371, 377, 2338, 1235, 2350, 391, 394, 395, 396, 397, 2374, 401, 409, 5014,
1477, 421, 2401, 2402, 2404, 2406, 2407, 2415, 425, 695, 434, 449, 1868, 1485, 1487, 481, 485, 2453, 507, 1352,
2465, 924, 504, 505, 2491, 2490, 574, 2501, 2503, 553, 557, 5570, 5571, 5236, 605, 613, 740, 2546, 2548, 2549, 6
28, 630, 1430, 1431, 1432, 6132, 646, 1411, 2580, 680, 1841, 2606, 694, 699, 2598, 1653, 2621, 1600, 2627, 2629,
716, 1885, 720, 3459, 5137, 1258, 743, 745, 1255, 2679, 2670, 1417, 752, 2712, 2699, 2695, 1866, 2728, 2733, 112
9, 830, 845, 2750, 2751, 857, 1706, 2767, 2770, 2779, 2796, 2807, 2790, 2797, 2819, 2822, 937, 2829, 1784, 2873,
2875, 2860, 2861, 2879, 2881, 1319, 2885, 2886, 986, 990, 992, 993, 2900, 2901, 2922, 1029, 1176, 1046, 1059, 10
82, 2986, 1094, 1113, 1135, 1144, 3009, 3026, 3032, 3040, 3042, 3043, 1161, 3058, 3059, 3077, 3323, 3088, 3116,
3118, 1290, 3138, 3142, 1504, 1310, 3167, 3168, 3192, 3176, 3190, 1355, 3213, 3236, 3552, 5785, 1388, 1393, 3233
, 1409, 3255, 3258, 3276, 1414, 1416, 3269, 3290, 1433, 3304, 1458, 1473, 1494, 3327, 3348, 1537, 5895, 1550, 15
59, 3356, 1586, 1596, 1601, 1602, 1603, 1647, 3403, 1665, 1667, 3415, 3417, 3448, 1676, 3452, 3465, 3484, 1722,
1723, 1732, 3495, 1739, 3529, 3503, 3505, 3515, 3526, 3537, 1806, 1813, 3544, 1816, 3551, 3553, 3577, 3579, 1884
, 3673, 3693, 3734, 3757, 3764, 3817, 3820, 3827, 3843, 3850, 3856, 3883, 3890, 3910, 3914, 3915, 5724, 3937, 39
46, 3948, 3945, 6181, 4275, 3979, 3976, 4002, 4003, 4016, 4032, 4051, 4076, 4103, 4111, 4112, 4113, 4114, 4160,
4164, 4161, 4162, 4174, 4179, 4187, 4196, 4197, 4211, 4220, 4227, 4228, 4234, 5418, 4257, 4260, 4276, 4279, 4297
, 4405, 4303, 5452, 4306, 4313, 4314, 4330, 4353, 4357, 4358, 4382, 4384, 4385, 4386, 4399, 4419, 4428, 4445, 44
57, 4456, 4532, 5326, 4470, 4465, 4472, 4482, 4508, 4553, 4579, 4598, 4636, 4689, 4705, 4728, 4752, 4751, 4755,
4769, 4784, 4810, 4874, 4900, 6138, 5893, 4949, 5267, 4988, 4999, 5008, 5029, 5044, 5090, 5122, 5201, 5245, 5255
, 5263, 5281, 5332, 5339, 5352, 5359, 5386, 5424, 5471, 5478, 5498, 5510, 5515, 5532, 5543, 5599, 5617, 5654, 57
26, 5730, 5792, 5809, 5828, 5819, 5834, 5928, 5950, 5962, 5978, 5988, 6060, 6067, 6077, 6092, 6174, 6175, 6264]
Community 4: Size 329, Nodes: [1702, 2237, 3637, 1952, 1968, 76, 80, 84, 105, 2027, 1041, 2019, 2021, 1332, 2069
, 2092, 470, 4945, 2116, 187, 201, 714, 2153, 2154, 2155, 2156, 2157, 2159, 2160, 621, 776, 294, 2519, 2304, 229
9, 2330, 825, 2376, 404, 835, 3485, 2397, 2416, 2425, 469, 471, 474, 476, 523, 2935, 2583, 1040, 635, 2968, 746,
678, 683, 1347, 2597, 2658, 2689, 766, 771, 2698, 1003, 1402, 1717, 2706, 775, 780, 802, 1327, 827, 829, 832, 17
38, 2768, 2762, 2776, 2804, 2802, 970, 1735, 2932, 1055, 2962, 2963, 2964, 2967, 1044, 1045, 2969, 2970, 3053, 1
273, 3122, 3123, 3124, 3127, 3137, 1534, 3170, 3223, 1375, 3227, 3228, 3281, 1444, 3261, 1443, 5490, 5491, 1495,
1496, 4793, 4818, 6061, 6062, 3363, 1572, 3357, 3359, 1611, 3383, 3386, 3394, 1675, 3420, 3424, 3425, 3426, 3427
, 3431, 3432, 3433, 3434, 3435, 3436, 3437, 3471, 1736, 3542, 1889, 3638, 3687, 3720, 3724, 3736, 3747, 3990, 38
37, 3855, 3857, 3858, 3871, 3887, 3875, 3876, 3877, 3878, 3908, 3909, 3911, 5958, 5959, 5960, 5115, 3964, 3991,
3992, 3993, 3994, 3997, 4000, 4021, 4099, 4100, 5304, 5159, 4271, 4311, 4441, 4461, 5404, 4573, 4641, 4546, 4547
, 4653, 4609, 4610, 4614, 4615, 4616, 4625, 4633, 4646, 4648, 4649, 4650, 4651, 4652, 4664, 4684, 4700, 4701, 47
02, 4703, 4704, 4722, 4727, 4729, 4730, 4735, 4824, 4854, 4788, 4814, 4877, 4951, 5939, 4961, 4984, 4985, 4986,
4987, 5037, 5038, 5042, 5066, 5361, 5149, 5323, 5257, 5292, 5296, 5397, 5398, 5408, 5409, 5434, 5435, 5436, 5419
, 5420, 5429, 5453, 5473, 5493, 5497, 5588, 5603, 5611, 5620, 5656, 5734, 5735, 5749, 5766, 5763, 5764, 5765, 57
89, 5790, 5791, 5805, 5827, 5845, 5846, 5896, 5956, 5972, 6059, 6013, 6072, 6079, 6082, 6083, 6139, 6150, 6151,
6153, 6249, 6170, 6197, 6198, 6199, 6200, 6201, 6202, 6203, 6204, 6205, 6206, 6207, 6208, 6209, 6210, 6211, 6212
, 6213, 6214, 6215, 6216, 6217, 6218, 6219, 6220, 6221, 6222, 6223, 6224, 6225, 6226, 6227, 6228, 6238, 6239, 62
40, 6245, 6259, 6260, 6261, 6268, 6269, 6270, 6282, 6283, 6291]
Community 5: Size 318, Nodes: [1842, 22, 23, 26, 32, 1954, 1699, 1700, 1703, 44, 46, 47, 48, 51, 52, 54, 56, 65,
1994, 1982, 86, 2003, 2004, 2005, 2008, 2009, 472, 1142, 211, 2016, 1361, 1787, 2059, 503, 131, 2095, 813, 3715,
1490, 2131, 2133, 2143, 494, 1358, 2161, 2165, 2202, 513, 2207, 2215, 2226, 2232, 279, 281, 282, 284, 287, 1789,
2238, 2239, 2240, 2242, 822, 432, 2331, 2314, 2321, 1792, 2384, 2377, 428, 429, 436, 448, 2439, 2448, 1488, 2478
, 580, 1288, 4516, 3171, 1275, 1427, 2539, 2540, 2542, 2537, 2544, 2547, 1326, 2566, 2599, 2637, 2642, 738, 2722
, 817, 2677, 2702, 1328, 889, 2951, 2961, 1397, 3777, 981, 2867, 2869, 1143, 2883, 2897, 1015, 2917, 2919, 5106,
2942, 2950, 2989, 3007, 3045, 3104, 3083, 1254, 1277, 1282, 1283, 1285, 3194, 3172, 3179, 1356, 1357, 3237, 3238

, 3239, 3240, 3241, 3230, 3289, 3242, 4172, 3376, 1563, 1574, 3372, 1597, 3382, 1652, 3414, 3418, 1674, 1677, 16
78, 1680, 1681, 3464, 3455, 3457, 3458, 1746, 3507, 1785, 1788, 3701, 3539, 3540, 3565, 3578, 3582, 1859, 3608,
3805, 1876, 3690, 3778, 3783, 3784, 3773, 3774, 3791, 3797, 3862, 3868, 3880, 3889, 3912, 3916, 4894, 3954, 3980
, 4373, 5164, 5437, 4039, 4062, 4064, 4069, 4071, 4075, 4093, 4098, 4122, 4107, 4120, 4146, 4163, 4191, 4192, 41
99, 4200, 4315, 4266, 4267, 4293, 4356, 4337, 4338, 4366, 4367, 4377, 4379, 4380, 4415, 4435, 4423, 4458, 4475,
4476, 4502, 4577, 5638, 4679, 4719, 4720, 4721, 4745, 4783, 4789, 4798, 4808, 4864, 4871, 4884, 4886, 4887, 4895
, 4897, 4898, 4928, 5891, 5892, 5894, 5848, 5849, 5851, 5853, 5937, 4991, 4993, 4994, 5045, 5078, 5079, 5080, 51
03, 5102, 5121, 5131, 5193, 5194, 5197, 5198, 5211, 5222, 5225, 5244, 5282, 5298, 5299, 5301, 5331, 5333, 5334,
5335, 5365, 5425, 5438, 5439, 5441, 5475, 5461, 5500, 5504, 5533, 5534, 5536, 5614, 5615, 5616, 5637, 5639, 5640
, 5646, 5657, 5698, 5699, 5715, 5900, 5915, 5922, 5957, 6066, 6084, 6085, 6086, 6154, 6156]
Community 6: Size 288, Nodes: [18, 1920, 957, 1459, 1924, 60, 1937, 1938, 1939, 1941, 1944, 1945, 1946, 66, 69,
70, 71, 72, 73, 74, 1956, 1970, 1974, 1976, 262, 85, 1987, 2081, 3219, 151, 153, 154, 160, 2109, 2110, 2111, 211
2, 2115, 5584, 1556, 223, 227, 840, 2184, 296, 308, 2287, 4154, 2345, 1733, 392, 2382, 2383, 2386, 2387, 399, 40
2, 403, 406, 407, 408, 2410, 2419, 1491, 2461, 512, 524, 1629, 602, 603, 749, 1870, 2587, 2611, 1581, 727, 5138,
2673, 795, 831, 860, 880, 2791, 952, 954, 956, 959, 2836, 3491, 961, 964, 966, 2845, 2848, 2849, 2850, 2864, 288
4, 2894, 1005, 2904, 2905, 2908, 2910, 1009, 1011, 1012, 1013, 1014, 1017, 3044, 2912, 2913, 2914, 2915, 1533, 1
033, 1035, 1037, 1038, 1039, 1651, 2955, 2956, 2957, 1081, 1720, 2978, 2979, 2980, 2983, 1457, 3025, 1172, 1174,
1178, 1179, 3050, 3051, 4711, 3064, 3100, 1385, 3117, 3709, 3208, 1369, 3220, 3221, 3222, 1379, 1381, 1384, 1386
, 1387, 3335, 3248, 3250, 3251, 3280, 3271, 1546, 3310, 1548, 1549, 1551, 1552, 1553, 1554, 1555, 3361, 3341, 33
73, 1579, 1580, 3398, 3399, 3400, 3401, 4109, 3446, 3475, 1714, 1715, 3492, 3493, 3494, 3607, 1871, 1874, 1875,
3685, 3686, 3708, 3710, 3711, 3722, 3723, 3725, 3745, 3746, 5541, 5795, 5796, 5797, 3780, 3788, 3852, 3923, 3924
, 3925, 3927, 3986, 5163, 4001, 4044, 4171, 4202, 4217, 4239, 4242, 4287, 4277, 4388, 4401, 4473, 4640, 4511, 45
44, 4545, 4574, 4580, 4592, 4593, 4601, 4602, 4603, 4604, 4605, 4606, 4607, 4628, 4668, 4676, 4685, 4693, 4710,
4712, 4713, 4734, 4743, 4744, 4959, 5027, 5028, 5036, 5058, 5095, 5143, 5144, 5145, 5180, 5227, 5247, 5868, 5362
, 5363, 5364, 5380, 5381, 5377, 5378, 5384, 5426, 5427, 5605, 5671, 5701, 5721, 5722, 5800, 5537, 5538, 5539, 55
40, 5869, 5870, 6046]
Community 7: Size 288, Nodes: [1219, 2524, 1942, 1064, 1981, 2051, 658, 2070, 2088, 446, 3337, 411, 168, 1264, 2
26, 2171, 2173, 2174, 2175, 2218, 267, 268, 414, 2230, 2231, 2233, 289, 293, 299, 443, 774, 2250, 300, 301, 303,
304, 306, 309, 764, 1061, 1093, 1095, 2255, 1018, 3677, 334, 2285, 364, 360, 362, 385, 2339, 2340, 2341, 2342, 3
93, 410, 415, 416, 417, 419, 420, 1256, 2839, 5013, 5015, 769, 3215, 676, 799, 803, 1019, 2434, 2435, 2436, 2437
, 2438, 2396, 2400, 439, 441, 442, 444, 445, 447, 2440, 460, 2463, 528, 530, 533, 535, 536, 1482, 2498, 2502, 25
04, 1132, 765, 819, 2551, 798, 656, 657, 661, 662, 663, 664, 686, 1162, 1309, 2588, 2591, 2592, 2638, 2639, 741,
2720, 2721, 2723, 2708, 768, 770, 772, 773, 2701, 2703, 2907, 797, 805, 806, 2738, 2739, 847, 1481, 2794, 2799,
1228, 1124, 2926, 2929, 2966, 1042, 1057, 1062, 1065, 1091, 1125, 1128, 1130, 1164, 1166, 1344, 3060, 3074, 4906
, 4907, 1251, 1252, 1257, 3114, 1260, 1261, 1263, 1266, 3121, 1345, 3164, 3175, 3184, 3186, 4132, 1398, 1399, 32
46, 3282, 3300, 1478, 1479, 1480, 1483, 1526, 3362, 1582, 1791, 3371, 1625, 1627, 3384, 1632, 1659, 3412, 3571,
1692, 1693, 1695, 1696, 1697, 1698, 3450, 3454, 1712, 1713, 3472, 3473, 3474, 3609, 3661, 3718, 3719, 3758, 3760
, 3761, 3762, 3763, 3775, 3821, 3834, 3835, 3836, 3838, 3854, 3885, 3906, 3907, 3938, 3939, 3940, 3941, 3959, 40
82, 4083, 4084, 4957, 5305, 4115, 4144, 4145, 4147, 4133, 4135, 5840, 4188, 4214, 4231, 4233, 4235, 5154, 4290,
4284, 4295, 4390, 4392, 4407, 4433, 4493, 4608, 4630, 4631, 4632, 4681, 4707, 4708, 4746, 4747, 4748, 4749, 4811
, 4831, 4832, 4905, 4954, 4955, 4956, 5098, 5250, 5265, 5266, 5268, 5394, 5462, 5465, 5466, 5733, 5901, 6018, 60
54, 6055]
Community 8: Size 260, Nodes: [1896, 520, 1912, 1656, 2007, 614, 1940, 77, 2026, 578, 1451, 2061, 2480, 1766, 21
34, 191, 2365, 2367, 2368, 2369, 2370, 2371, 2372, 197, 198, 199, 200, 203, 204, 205, 207, 2149, 2150, 2151, 597
3, 5974, 1072, 1891, 2246, 305, 710, 1498, 2256, 2258, 711, 2279, 843, 1587, 357, 359, 1334, 2333, 2334, 2315, 9
96, 1814, 3029, 502, 516, 517, 519, 522, 525, 526, 537, 538, 539, 541, 542, 543, 544, 545, 546, 2534, 2507, 2508
, 2509, 2512, 2513, 607, 608, 611, 615, 616, 1500, 876, 2633, 2607, 2615, 707, 1503, 2641, 709, 713, 1886, 800,
5628, 836, 839, 841, 844, 1707, 997, 2902, 2930, 1036, 1092, 1645, 3001, 3010, 3056, 1168, 3048, 3049, 3061, 306
9, 3125, 3166, 3243, 3270, 1445, 1446, 1447, 1448, 1449, 1450, 1452, 1497, 1499, 1501, 1502, 1505, 3339, 1797, 1
584, 1590, 3370, 1650, 1654, 3404, 3407, 3410, 3477, 3481, 3482, 3483, 1737, 3528, 3680, 3681, 1811, 3543, 3545,
3546, 3547, 3548, 3562, 3581, 3674, 3658, 3672, 3678, 3750, 3776, 3785, 3787, 3884, 3913, 3965, 4004, 4030, 4070
, 4125, 4143, 4149, 4150, 5653, 4205, 4219, 4221, 4236, 4269, 4425, 4449, 4451, 4481, 4483, 4486, 4488, 4642, 46
87, 4742, 4797, 4800, 4801, 4879, 4922, 4939, 4942, 4995, 5026, 5030, 5031, 5032, 5033, 5034, 5059, 5064, 5065,
5105, 5156, 5167, 5272, 5291, 5293, 5294, 5295, 5306, 5369, 5503, 5505, 5590, 5601, 5602, 5621, 5622, 5625, 5673
, 5674, 5691, 5736, 5737, 5660, 5661, 5662, 5663, 5664, 5665, 5666, 5667, 5668, 5669, 5760, 5762, 5767, 5768, 57
94, 5866, 5885, 5912, 5913, 5927, 5929, 5986, 5989, 5990, 5991, 6109, 6111, 6243]
Community 9: Size 248, Nodes: [1097, 1021, 960, 3874, 1961, 2006, 87, 88, 90, 96, 97, 1476, 108, 324, 365, 1486,
2039, 2040, 2042, 111, 1670, 1846, 2067, 2068, 2080, 161, 1390, 604, 2429, 2172, 231, 244, 2197, 2199, 1389, 164
4, 1306, 1321, 2251, 2259, 310, 312, 313, 316, 317, 319, 320, 2267, 321, 322, 323, 328, 349, 2291, 1322, 473, 17
16, 2296, 2297, 2298, 2325, 2318, 1741, 413, 2532, 2393, 463, 3309, 4829, 4830, 479, 2452, 2454, 2455, 2457, 245
8, 2459, 2460, 579, 581, 1330, 1311, 2516, 5237, 1682, 2531, 2533, 2538, 599, 601, 606, 1439, 2555, 2556, 1392,
2602, 1641, 2655, 737, 2660, 1863, 1709, 1351, 1152, 953, 2880, 2933, 2940, 2971, 1204, 3099, 3101, 3102, 3103,
3144, 3149, 1354, 1642, 3203, 3224, 1391, 1395, 3234, 1460, 1461, 3317, 3320, 1484, 3324, 3326, 3338, 1643, 1646
, 1648, 1649, 3397, 3402, 1663, 3430, 1808, 1840, 1855, 3600, 3697, 3741, 3744, 3789, 3808, 5170, 5172, 5173, 51
74, 3818, 3879, 3839, 5468, 3922, 6108, 4806, 6121, 6122, 3963, 3987, 4009, 4011, 4013, 4022, 4023, 4024, 4025,
4026, 4059, 4063, 4121, 4108, 4110, 4304, 4332, 4348, 4352, 4403, 4478, 4479, 4506, 4535, 4549, 4550, 4709, 4678
, 4673, 4674, 4682, 4758, 4815, 4908, 4936, 4938, 4981, 5124, 5125, 5063, 5083, 5084, 5161, 5157, 5217, 5218, 52
19, 5230, 5251, 5283, 5313, 5314, 5315, 5316, 5317, 5319, 5451, 5563, 5564, 5565, 5677, 5738, 5739, 5740, 5741,
5742, 5743, 5744, 5745, 5746, 5769, 5770, 5772, 5874, 5875, 5876, 5877, 5878, 5899, 5935, 5961, 6056, 6058, 6152
, 6180, 6182, 6244]
Community 10: Size 242, Nodes: [703, 1895, 11, 12, 16, 19, 92, 25, 27, 1022, 901, 1624, 1911, 1914, 2170, 1932,
106, 898, 94, 1776, 2015, 2017, 864, 5048, 2190, 1861, 235, 2198, 900, 2216, 260, 288, 967, 1620, 2244, 2245, 15
07, 881, 2309, 2427, 2456, 4455, 4980, 2892, 2953, 2529, 2530, 1426, 2579, 2595, 1027, 705, 2634, 2635, 2636, 26
40, 1594, 2625, 2626, 2628, 2643, 2644, 1424, 2697, 2704, 837, 850, 1333, 2744, 2747, 1860, 1336, 2757, 2771, 27
72, 2960, 896, 897, 899, 903, 905, 2785, 2798, 2855, 2856, 2858, 2859, 1829, 1031, 2952, 2954, 2946, 3018, 1134,
3011, 3047, 1331, 1211, 3080, 1220, 1265, 3177, 1767, 1413, 3285, 3286, 3287, 3951, 4635, 1506, 1508, 1510, 1513
, 1514, 1515, 1595, 1616, 1617, 1621, 1622, 1623, 1658, 1672, 3409, 1673, 3449, 3470, 3487, 3504, 1768, 1769, 17
71, 1773, 1774, 3550, 3587, 1856, 1858, 3713, 3714, 3743, 3807, 3809, 3810, 3811, 3796, 3851, 3882, 3943, 4092,
4005, 4006, 4007, 5165, 4042, 4065, 4066, 4263, 4155, 4156, 4157, 4229, 4238, 4270, 4406, 4307, 4309, 5277, 4354
, 4372, 4374, 4376, 4400, 4429, 4430, 4431, 4422, 4439, 4453, 4454, 4474, 5634, 4566, 4613, 4617, 6037, 4726, 47
39, 4754, 4771, 4775, 4776, 4777, 4866, 4914, 4930, 4933, 4935, 5010, 5011, 5050, 5077, 5096, 5104, 5260, 5261,
5262, 5318, 5353, 5379, 5387, 5388, 5389, 5390, 5396, 5407, 5454, 5455, 5467, 5472, 5479, 5546, 5548, 5566, 5567
, 5585, 5586, 5589, 5635, 5636, 5680, 5758, 5761, 5773, 5774, 5839, 5882, 5871, 5872, 5888, 6015, 6127]

Community 11: Size 232, Nodes: [826, 1900, 3619, 4408, 34, 1922, 338, 55, 58, 61, 62, 63, 64, 907, 1993, 1995, 1
996, 1997, 1998, 1999, 1194, 1196, 1198, 1959, 1960, 1962, 1963, 1977, 1978, 1983, 102, 1949, 1951, 1973, 98, 10
0, 103, 104, 2028, 2025, 1535, 135, 2094, 786, 1274, 2099, 2100, 2101, 2102, 164, 2127, 1343, 229, 2181, 1069, 2
61, 291, 297, 2265, 331, 332, 333, 337, 2294, 2329, 2336, 2352, 2343, 2364, 430, 435, 2417, 2418, 2420, 2421, 44
0, 1607, 2492, 3617, 5476, 484, 1764, 2494, 2495, 2496, 2477, 527, 1171, 1342, 2837, 3786, 3283, 2541, 748, 750,
2589, 739, 1609, 2664, 2665, 2674, 2675, 2676, 1024, 2714, 2716, 2717, 2718, 1867, 1873, 2778, 2805, 2806, 2787,
1213, 4411, 2889, 3004, 1341, 1192, 3071, 3113, 1262, 3126, 1294, 1296, 1297, 1298, 1763, 1765, 3163, 3193, 3204
, 3196, 3197, 3198, 3329, 1822, 1608, 3334, 3336, 1566, 3351, 3353, 3354, 3369, 1605, 1606, 3421, 3439, 3456, 34
97, 3500, 1751, 3513, 1758, 3520, 3523, 3525, 3527, 3595, 1864, 1869, 3615, 3618, 3626, 3688, 3689, 3691, 3692,
3867, 3703, 3742, 3766, 3770, 3859, 3860, 3869, 3870, 3872, 3873, 4080, 4008, 4027, 4029, 4041, 4040, 4050, 4052
, 4185, 4274, 4283, 4412, 4584, 4658, 4714, 4490, 4567, 4582, 4680, 4715, 4741, 4762, 4782, 4785, 4786, 4787, 48
13, 4865, 4962, 4971, 4972, 5012, 5073, 5168, 5178, 5207, 5320, 5351, 5460, 5488, 5489, 5529, 5624, 5678, 5679,
5708, 5709, 5751, 5542, 5867, 5943]
Community 12: Size 232, Nodes: [1915, 637, 870, 67, 2222, 2789, 3332, 3769, 1337, 2205, 2044, 2057, 137, 2093, 1
150, 4463, 1728, 171, 5483, 2158, 220, 1726, 398, 2378, 2379, 2380, 5484, 5501, 640, 4781, 475, 2482, 2483, 2484
, 2487, 2472, 518, 2671, 642, 910, 2553, 747, 636, 639, 641, 643, 869, 875, 877, 2581, 2686, 2667, 2668, 2700, 8
10, 1323, 1762, 2753, 2754, 885, 2773, 906, 911, 912, 2800, 2801, 963, 2851, 2854, 1058, 3109, 3046, 1724, 3165,
3199, 3200, 3201, 1338, 1725, 1727, 3178, 5494, 3256, 3318, 3325, 1529, 3330, 5916, 1612, 3377, 3378, 3385, 3388
, 1691, 3669, 3467, 3488, 3490, 3516, 3563, 3632, 3765, 3813, 3881, 3896, 3898, 3899, 5310, 6000, 6001, 4101, 41
18, 4176, 4201, 4203, 4256, 4298, 4305, 6039, 4375, 4378, 4572, 4575, 4576, 4586, 4637, 6036, 4796, 4917, 4918,
4919, 5850, 4934, 4975, 5049, 5814, 5815, 5118, 5129, 5130, 5200, 5270, 5288, 5307, 5376, 5395, 5459, 5480, 5481
, 5482, 5495, 5511, 5577, 5578, 5580, 5607, 5644, 5651, 5710, 5711, 5712, 5713, 5754, 5820, 5832, 5833, 5847, 58
55, 5879, 5883, 5887, 5917, 5923, 5919, 5920, 5944, 5951, 5964, 5965, 5994, 6002, 6003, 6004, 6005, 6022, 6026,
6027, 6028, 6029, 6030, 6033, 6034, 6035, 6042, 6043, 6044, 6045, 6047, 6048, 6076, 6088, 6089, 6099, 6105, 6106
, 6107, 6112, 6113, 6114, 6115, 6116, 6137, 6157, 6158, 6159, 6160, 6161, 6162, 6164, 6166, 6237, 6179, 6189, 61
90, 6191, 6192, 6194, 6289, 6266, 6274, 6275, 6276]
Community 13: Size 228, Nodes: [95, 159, 2106, 206, 230, 326, 922, 4180, 2212, 292, 1004, 2277, 2280, 2281, 2282
, 2283, 2284, 344, 693, 400, 418, 3306, 1002, 763, 2467, 2468, 2469, 496, 1471, 2473, 756, 561, 1631, 2552, 2603
, 2605, 2608, 2609, 2610, 2613, 2656, 1557, 2672, 757, 758, 761, 2809, 2810, 2811, 2812, 2813, 2814, 2815, 2816,
2817, 2823, 931, 941, 943, 6087, 950, 1747, 2824, 2825, 2827, 2890, 999, 1000, 1007, 1008, 1440, 1760, 2906, 291
6, 2928, 1221, 2994, 1117, 1119, 1120, 1121, 1122, 1177, 3082, 1218, 3981, 1279, 3128, 3129, 3130, 3131, 3174, 3
202, 1377, 3225, 3226, 5717, 1400, 3262, 3263, 3264, 3265, 3267, 3268, 3273, 3296, 3302, 3303, 1518, 1519, 3349,
3350, 3360, 1558, 1560, 3438, 1742, 1744, 1745, 3557, 3569, 3599, 1857, 3610, 3611, 3612, 3614, 3679, 3682, 3695
, 3712, 3748, 3794, 3895, 3897, 3962, 4105, 4165, 4183, 4273, 4321, 4322, 4323, 4334, 4371, 4397, 4398, 4402, 44
04, 4548, 4558, 4571, 4590, 4611, 4612, 4645, 4647, 4657, 4660, 4661, 4690, 4691, 4692, 4759, 4761, 4790, 4809,
4816, 4817, 4852, 4853, 4855, 4929, 4998, 5018, 5496, 6241, 6242, 5126, 5127, 5128, 5109, 5111, 5112, 5147, 5158
, 5213, 5220, 5221, 5345, 5385, 5527, 5528, 5655, 5705, 5706, 5707, 5716, 5718, 5719, 5771, 5802, 5824, 5863, 58
64, 5865, 5945, 5946, 5971, 6006, 6091, 6098, 6117, 6118, 6119, 6120, 6125, 6252, 6184, 6185, 6186, 6187, 6188,
6255, 6256, 6257, 6265, 6267]
Community 14: Size 218, Nodes: [1591, 1669, 1910, 1028, 75, 190, 1964, 1965, 1966, 1967, 133, 1193, 2085, 285, 5
171, 158, 2104, 2105, 2107, 181, 185, 186, 189, 2141, 2219, 916, 873, 1300, 270, 271, 272, 273, 274, 276, 277, 2
78, 600, 1583, 2248, 2249, 1227, 290, 2288, 2471, 2423, 4828, 2476, 2505, 548, 549, 551, 555, 556, 982, 2518, 62
0, 649, 2568, 834, 2715, 815, 816, 4695, 5224, 5629, 2756, 2759, 2760, 949, 2830, 2831, 2832, 2833, 2834, 2835,
1540, 2852, 3028, 3030, 3031, 2888, 2891, 3462, 1750, 2943, 2993, 3023, 1167, 1170, 1180, 1184, 5518, 5519, 5520
, 3078, 1216, 3216, 3299, 1474, 1493, 4891, 1542, 1543, 3346, 1585, 1588, 1592, 1593, 1668, 3413, 3416, 3453, 17
48, 3510, 3511, 3512, 3517, 3522, 1772, 3530, 1818, 3555, 3556, 3801, 3738, 5177, 3900, 5114, 5116, 5117, 3971,
3972, 3973, 3978, 4281, 4077, 4081, 4096, 4148, 4137, 4216, 4285, 4286, 4326, 4327, 4328, 4329, 4331, 4333, 4345
, 4346, 4347, 4349, 4350, 4351, 4355, 4340, 4360, 4362, 4393, 4394, 4395, 4396, 4487, 4536, 4683, 4763, 4764, 48
19, 4820, 4880, 4916, 4989, 4990, 5890, 5100, 5141, 5142, 5184, 5185, 5186, 5187, 5188, 5189, 5223, 5240, 5232,
5233, 5234, 5242, 5243, 5340, 5341, 5423, 5545, 5547, 5549, 5550, 5551, 5630, 5642, 5676, 5681, 5682, 5683, 5684
, 5685, 5686, 5687, 5689, 5690, 5720, 6292, 6293, 6294, 6295, 5914, 6300]
Community 15: Size 177, Nodes: [1899, 1906, 33, 35, 37, 40, 41, 42, 1988, 89, 914, 2011, 2012, 2013, 99, 116, 20
30, 2065, 2054, 2055, 1466, 1752, 872, 188, 214, 216, 1349, 1301, 1305, 2247, 2311, 2335, 2357, 1415, 2481, 2464
, 585, 1730, 647, 648, 650, 651, 652, 653, 2572, 2573, 2574, 2575, 1350, 689, 691, 692, 1380, 2734, 866, 1408, 2
763, 2765, 895, 1661, 2841, 1191, 1025, 3191, 5107, 5108, 1075, 1076, 1078, 1079, 1080, 1083, 2982, 1118, 2998,
2999, 3000, 1181, 1182, 1183, 1185, 1186, 1187, 1188, 1189, 1190, 3054, 1340, 3134, 1346, 1348, 3150, 3151, 3152
, 3160, 3189, 3288, 1403, 1405, 1406, 1410, 3272, 1462, 1463, 1464, 1465, 1467, 1468, 1469, 1470, 1472, 3313, 33
14, 3315, 5414, 1657, 1660, 1662, 1664, 3518, 3519, 3616, 3768, 3790, 3828, 3830, 3831, 3832, 3833, 3934, 3996,
5162, 4116, 4117, 4119, 4316, 4363, 4381, 4503, 4554, 4555, 4556, 4557, 4559, 4560, 4561, 4562, 4581, 4671, 4672
, 4883, 4858, 4859, 4860, 4861, 4862, 4863, 4946, 4947, 4948, 4950, 4952, 5889, 5086, 5094, 5183, 5246, 5346, 55
09, 5583, 5859, 5860, 5861, 5862, 6155, 6230, 6231]
Community 16: Size 165, Nodes: [14, 809, 39, 1378, 1923, 1925, 1927, 45, 50, 79, 82, 2022, 157, 2120, 196, 2142,
2144, 218, 221, 225, 1749, 2177, 2180, 2188, 341, 370, 862, 2289, 1734, 372, 373, 374, 375, 379, 380, 2316, 2317
, 2320, 2373, 1230, 1236, 2344, 951, 814, 455, 2441, 2442, 514, 2557, 655, 2569, 2565, 2567, 671, 3654, 4878, 85
9, 861, 863, 865, 868, 1775, 2740, 2741, 2742, 2743, 1145, 1147, 1148, 1149, 1153, 933, 948, 1208, 2870, 2927, 1
890, 1200, 1201, 1206, 1207, 3072, 3073, 3075, 3076, 3670, 4540, 1280, 1295, 1754, 1453, 1455, 1694, 3443, 3444,
3445, 3447, 5176, 3506, 3508, 3639, 3640, 3641, 3642, 3643, 3644, 3645, 3646, 3647, 3648, 3649, 3650, 3651, 3652
, 3653, 3655, 3656, 3657, 3659, 3660, 3662, 3663, 3664, 3665, 3666, 3667, 3668, 3694, 3844, 3933, 4102, 4151, 42
65, 4296, 4361, 4369, 5777, 5778, 5780, 4510, 4518, 4724, 4772, 4870, 4881, 4958, 5043, 5091, 5252, 5256, 5258,
5259, 5328, 5329, 5330, 5368, 5444, 5445, 5446, 5672, 5753, 5966, 5967, 6171, 6236]
Community 17: Size 158, Nodes: [1833, 1679, 81, 1984, 1985, 1989, 1990, 1991, 1419, 2058, 2113, 1165, 1575, 184,
1071, 1800, 2268, 354, 361, 1217, 2394, 2450, 1267, 575, 824, 823, 1827, 1854, 610, 2560, 2561, 2614, 1195, 744,
1223, 894, 1324, 1325, 848, 2755, 1292, 2821, 1001, 1100, 1291, 2934, 1016, 2944, 1131, 1155, 3034, 3035, 3036,
3055, 3057, 5192, 5311, 5312, 3115, 1268, 1269, 1270, 1271, 1286, 1293, 1404, 3274, 3277, 3278, 3279, 1420, 1422
, 1425, 1729, 1731, 3536, 1802, 1820, 3554, 3559, 3561, 1828, 1830, 1832, 3583, 3584, 3573, 3602, 3603, 3604, 36
20, 3698, 3759, 3795, 3918, 3921, 3967, 4043, 4140, 4159, 4208, 4212, 4218, 4246, 4292, 6279, 4320, 4537, 4538,
4432, 4466, 4570, 4644, 4662, 4804, 4807, 4833, 4834, 4835, 5019, 5020, 5022, 5054, 5055, 5076, 5097, 5133, 5134
, 5135, 5136, 5278, 5273, 5392, 5411, 5522, 5568, 5647, 5648, 5649, 5650, 5652, 5757, 5921, 6011, 6068, 6078, 61
69, 6234, 6235, 6284, 6285, 6286, 6287, 6288, 6296, 6297, 6298, 6299]
Community 18: Size 153, Nodes: [21, 1711, 1917, 1919, 1836, 3052, 2014, 1986, 2041, 110, 2029, 2031, 2033, 778,
2079, 5941, 3344, 700, 2138, 2166, 2191, 808, 1197, 295, 1850, 568, 2392, 464, 1509, 2485, 1547, 759, 804, 1843,
2616, 2618, 2619, 2620, 2719, 1454, 2707, 2709, 2710, 2711, 777, 781, 782, 783, 784, 785, 807, 811, 812, 1151, 2
729, 2730, 2731, 1367, 2777, 2853, 2948, 1098, 1099, 1101, 1102, 1103, 1104, 1105, 1106, 1169, 3081, 3084, 3218,

```
3249, 3253, 3232, 3292, 3307, 3308, 3342, 3345, 3406, 1710, 3702, 3570, 3574, 3575, 3596, 3597, 3598, 3683, 3732
, 3756, 3792, 3793, 5175, 3853, 3894, 4126, 4230, 4289, 4272, 4278, 4280, 4364, 4417, 4427, 5405, 4504, 4618, 46
19, 4688, 4812, 5688, 4912, 5000, 5040, 5067, 5093, 5153, 5181, 5347, 5552, 5612, 5613, 5631, 5632, 5633, 5626,
5783, 5784, 5793, 5804, 5837, 5854, 5873, 5910, 5976, 5977, 5979, 5983, 5984, 5985, 5993, 5995, 5996, 6007, 6031
, 6032, 6057, 6073, 6074, 6075]
Community 19: Size 102, Nodes: [13, 2074, 2761, 1943, 2203, 112, 1160, 2045, 2048, 2049, 4944, 1865, 5209, 659,
2220, 921, 632, 336, 2327, 633, 2405, 2479, 2510, 2545, 1429, 631, 874, 2576, 2577, 1628, 1887, 1156, 2764, 2803
, 1780, 2903, 1043, 3146, 1154, 1159, 1163, 4494, 4495, 4496, 4497, 4498, 4499, 1339, 5199, 1289, 3143, 3145, 31
95, 3297, 1666, 3514, 3533, 3534, 3535, 3812, 3814, 3816, 4138, 4139, 4175, 4177, 4178, 4299, 4310, 4387, 4500,
4578, 4600, 4841, 4791, 4792, 4903, 4932, 5152, 5327, 5559, 5560, 5561, 5562, 5573, 5574, 5575, 5576, 5581, 5582
, 5897, 6050, 6051, 6052, 6053, 6143, 6144, 6145, 6146, 6147, 6148, 6149]
Community 20: Size 101, Nodes: [1364, 2082, 2163, 1114, 1233, 302, 348, 1366, 2293, 2295, 2348, 405, 1565, 467,
1512, 2550, 1365, 2594, 3501, 731, 767, 2691, 2692, 2696, 2857, 1006, 2965, 1049, 1087, 1108, 1109, 1110, 1111,
1112, 1116, 3020, 3021, 3841, 1362, 1363, 1368, 3210, 3211, 3257, 1442, 3322, 1562, 1567, 3468, 3521, 3605, 3675
, 3846, 3920, 3953, 3968, 5645, 5449, 4141, 4206, 4389, 5779, 4569, 4594, 4595, 4596, 4597, 4659, 4716, 4794, 47
95, 4882, 4893, 4909, 5229, 5938, 4997, 5082, 5146, 5148, 5248, 5249, 5308, 5360, 5370, 5371, 5372, 5373, 5374,
5375, 5799, 5925, 5926, 5981, 6070, 6094, 6095, 6176, 6177, 6178, 6262]
Community 21: Size 86, Nodes: [3147, 24, 1979, 91, 1972, 243, 2194, 109, 114, 115, 117, 118, 119, 2097, 1531, 16
2, 163, 165, 166, 170, 2123, 2124, 1539, 236, 237, 239, 240, 242, 245, 2189, 2192, 286, 2241, 4153, 355, 968, 23
24, 499, 501, 506, 701, 702, 704, 2623, 2624, 4844, 2748, 971, 973, 974, 1066, 1126, 3148, 3305, 3301, 1532, 430
1, 1545, 1837, 3478, 3486, 3538, 1834, 1835, 1838, 3751, 3752, 3753, 3771, 3947, 4252, 4302, 4926, 4927, 4976, 5
024, 5492, 5727, 5841, 5842, 5843, 5844, 5947, 5948, 5949, 6123]
Community 22: Size 85, Nodes: [1898, 1901, 49, 2366, 2893, 3065, 3593, 3840, 3999, 2076, 2162, 2214, 318, 955, 4
31, 2422, 2424, 2426, 2428, 2784, 529, 2558, 2622, 942, 2654, 908, 2786, 2826, 2838, 2842, 984, 985, 987, 989, 9
91, 2895, 1026, 2937, 2996, 1852, 1205, 1849, 3067, 3068, 3070, 3316, 5140, 1847, 1848, 1851, 1853, 3767, 4028,
4097, 4167, 4213, 4225, 4226, 4243, 4336, 5516, 5517, 4489, 4491, 4585, 4643, 4740, 4768, 4920, 5366, 5342, 5457
, 5458, 5477, 5554, 5798, 5829, 5830, 5987, 6024, 6025, 6100, 6101, 6103, 6104]
Community 23: Size 11, Nodes: [588, 589, 590, 591, 592, 593, 594, 595, 596, 597, 598]
Community 24: Size 2, Nodes: [1683, 1684]
```

4. Applying the Leiden Algorithm

In [6]:

```python
import igraph as ig
import leidenalg as la

# Convert NetworkX graph to igraph
ig_graph = ig.Graph.TupleList(graph.edges(), directed=False)

# Apply Leiden algorithm
partition = la.find_partition(ig_graph, la.ModularityVertexPartition)

# Count the size of each community
community_sizes = {}
for node, community_id in enumerate(partition.membership):
    community_sizes.setdefault(community_id, 0)
    community_sizes[community_id] += 1

# Sort communities by size
sorted_communities = sorted(community_sizes.items(), key=lambda x: x[1], reverse=True)

# Get the top 100 communities
top_100_communities = sorted_communities[:100]

print("Top 100 Leiden communities:")
for i, (community_id, size) in enumerate(top_100_communities):
    nodes_in_community = [node for node, comm_id in enumerate(partition.membership) if comm_id == community_id]
    print(f"Community {i+1}: Size {size}, Nodes: {nodes_in_community}")
```

```
Top 100 Leiden communities:
Community 1: Size 742, Nodes: [16, 21, 46, 49, 50, 104, 108, 110, 256, 268, 274, 317, 330, 331, 332, 333, 334, 3
35, 337, 339, 340, 363, 369, 371, 372, 377, 379, 382, 385, 393, 420, 430, 432, 434, 437, 438, 440, 456, 457, 458
, 460, 464, 467, 468, 518, 560, 562, 568, 577, 581, 597, 609, 664, 671, 672, 686, 687, 708, 720, 729, 734, 735,
736, 737, 738, 740, 747, 750, 752, 756, 760, 770, 776, 780, 800, 815, 831, 837, 856, 859, 899, 906, 910, 921, 92
7, 928, 930, 931, 932, 933, 934, 938, 944, 949, 952, 957, 961, 968, 986, 987, 988, 989, 994, 995, 1009, 1014, 10
30, 1042, 1046, 1048, 1049, 1053, 1059, 1063, 1069, 1078, 1085, 1087, 1090, 1096, 1098, 1099, 1102, 1105, 1107,
1109, 1118, 1126, 1127, 1128, 1129, 1130, 1131, 1133, 1134, 1144, 1145, 1147, 1148, 1150, 1151, 1152, 1153, 1155
, 1160, 1161, 1162, 1163, 1164, 1165, 1169, 1170, 1171, 1181, 1206, 1211, 1212, 1216, 1224, 1232, 1235, 1254, 12
57, 1258, 1260, 1277, 1280, 1281, 1289, 1299, 1304, 1305, 1306, 1307, 1308, 1309, 1324, 1335, 1343, 1356, 1360,
1361, 1362, 1363, 1369, 1370, 1373, 1379, 1382, 1384, 1386, 1387, 1416, 1439, 1447, 1448, 1470, 1473, 1475, 1529
, 1541, 1546, 1548, 1556, 1603, 1608, 1618, 1624, 1625, 1640, 1657, 1663, 1685, 1709, 1711, 1722, 1766, 1784, 17
85, 1786, 1791, 1795, 1813, 1819, 1820, 1821, 1822, 1824, 1827, 1830, 1832, 1834, 1843, 1858, 1879, 1881, 1888,
1896, 1897, 1905, 1915, 1936, 1940, 1941, 1942, 1945, 1949, 1950, 1960, 1974, 1979, 1989, 1991, 2010, 2038, 2066
, 2067, 2082, 2083, 2084, 2086, 2088, 2089, 2106, 2109, 2111, 2112, 2114, 2122, 2166, 2179, 2191, 2193, 2200, 22
03, 2207, 2219, 2237, 2246, 2247, 2250, 2259, 2269, 2279, 2281, 2282, 2293, 2294, 2295, 2299, 2309, 2328, 2337,
2342, 2343, 2345, 2350, 2351, 2352, 2354, 2357, 2360, 2361, 2362, 2365, 2367, 2368, 2382, 2384, 2402, 2405, 2423
, 2434, 2452, 2453, 2457, 2461, 2511, 2551, 2556, 2595, 2597, 2598, 2601, 2611, 2639, 2644, 2653, 2667, 2671, 27
22, 2724, 2737, 2741, 2745, 2757, 2758, 2766, 2768, 2769, 2811, 2816, 2817, 2819, 2821, 2823, 2839, 2870, 2871,
2873, 2874, 2890, 2895, 2896, 2917, 2928, 2934, 2935, 2936, 2943, 2950, 2985, 2986, 3007, 3008, 3009, 3010, 3011
, 3020, 3039, 3044, 3045, 3046, 3047, 3049, 3051, 3053, 3054, 3057, 3060, 3061, 3069, 3071, 3081, 3109, 3110, 31
11, 3112, 3113, 3125, 3126, 3129, 3135, 3139, 3143, 3150, 3151, 3162, 3164, 3166, 3173, 3189, 3191, 3204, 3216,
3241, 3254, 3286, 3301, 3304, 3307, 3315, 3319, 3320, 3322, 3323, 3324, 3325, 3326, 3328, 3335, 3337, 3339, 3340
, 3344, 3345, 3347, 3348, 3353, 3363, 3364, 3365, 3370, 3374, 3375, 3376, 3393, 3415, 3420, 3441, 3443, 3450, 34
```

52, 3453, 3458, 3459, 3460, 3463, 3467, 3468, 3469, 3500, 3501, 3552, 3558, 3575, 3581, 3582, 3583, 3638, 3677, 3690, 3691, 3698, 3707, 3745, 3775, 3776, 3777, 3778, 3783, 3792, 3793, 3827, 3848, 3852, 3853, 3854, 3855, 3856, 3868, 3874, 3875, 3876, 3877, 3878, 3880, 3883, 3884, 3894, 3918, 3945, 3946, 3951, 3952, 3953, 3955, 3957, 3986, 3988, 3992, 4000, 4006, 4007, 4008, 4010, 4030, 4035, 4038, 4043, 4046, 4053, 4074, 4076, 4096, 4097, 4133, 4135, 4136, 4137, 4139, 4164, 4169, 4170, 4187, 4189, 4190, 4228, 4230, 4231, 4232, 4233, 4236, 4237, 4238, 4239, 4240, 4266, 4270, 4271, 4275, 4278, 4279, 4285, 4287, 4288, 4289, 4297, 4299, 4300, 4318, 4326, 4336, 4356, 4359, 4371, 4376, 4381, 4384, 4385, 4390, 4404, 4405, 4406, 4417, 4427, 4428, 4432, 4455, 4461, 4467, 4486, 4491, 4492, 4501, 4527, 4528, 4531, 4536, 4537, 4538, 4577, 4590, 4620, 4621, 4622, 4631, 4632, 4633, 4634, 4640, 4648, 4649, 4650, 4656, 4657, 4663, 4664, 4665, 4667, 4669, 4672, 4695, 4706, 4715, 4728, 4729, 4742, 4744, 4745, 4746, 4770, 4771, 4772, 4785, 4786, 4787, 4788, 4789, 4814, 4815, 4816, 4820, 4821, 4845, 4846, 4847, 4969, 4976, 4977, 4990, 4991, 4992, 4993, 5014, 5020, 5022, 5034, 5056, 5057, 5083, 5091, 5118, 5127, 5133, 5148, 5149, 5150, 5151, 5152, 5153, 5161, 5162, 5163, 5164, 5173, 5176, 5178, 5181, 5231, 5232, 5237, 5238, 5239, 5272, 5274, 5279, 5291, 5293, 5294, 5296, 5297, 5318, 5326, 5327, 5344, 5346, 5364, 5368, 5369, 5370, 5378, 5386, 5387, 5408, 5409, 5422, 5423, 5424, 5428, 5429, 5430, 5431, 5433, 5434, 5479, 5496, 5526, 5569, 5571, 5578, 5579, 5615, 5630, 5635, 5646, 5660, 5713, 5753, 5757, 5825, 5854, 5863, 5871, 5876, 5895, 5896, 5922, 5923, 5924, 5953, 5962, 5963, 5982, 6007, 6008, 6041, 6046, 6047, 6048, 6115, 6124, 6125, 6126, 6176, 6204, 6229, 6234, 6235, 6236, 6285, 6286]
Community 2: Size 729, Nodes: [17, 20, 45, 107, 132, 253, 342, 344, 349, 367, 368, 375, 376, 421, 422, 447, 450, 451, 454, 455, 462, 469, 495, 524, 550, 564, 588, 622, 627, 635, 677, 680, 691, 696, 703, 705, 710, 712, 713, 714, 717, 719, 721, 733, 755, 762, 764, 766, 773, 782, 784, 788, 790, 798, 810, 823, 825, 826, 827, 836, 850, 863, 877, 886, 912, 920, 922, 923, 924, 926, 951, 958, 966, 967, 974, 975, 996, 1004, 1005, 1008, 1017, 1020, 1027, 1028, 1031, 1032, 1033, 1034, 1035, 1037, 1039, 1040, 1041, 1045, 1050, 1054, 1055, 1056, 1061, 1064, 1065, 1067, 1068, 1072, 1076, 1086, 1088, 1091, 1093, 1094, 1095, 1097, 1106, 1114, 1115, 1136, 1141, 1158, 1174, 1176, 1178, 1179, 1180, 1184, 1185, 1186, 1187, 1190, 1198, 1202, 1203, 1204, 1205, 1207, 1220, 1225, 1231, 1233, 1234, 1238, 1240, 1263, 1264, 1266, 1268, 1269, 1270, 1271, 1272, 1273, 1274, 1284, 1285, 1287, 1288, 1292, 1294, 1325, 1345, 1348, 1350, 1357, 1388, 1395, 1397, 1405, 1412, 1414, 1417, 1426, 1428, 1429, 1430, 1431, 1433, 1437, 1441, 1442, 1443, 1452, 1453, 1454, 1455, 1456, 1457, 1458, 1459, 1460, 1469, 1471, 1479, 1484, 1488, 1489, 1497, 1503, 1506, 1517, 1518, 1519, 1520, 1530, 1531, 1533, 1535, 1536, 1539, 1547, 1559, 1560, 1561, 1571, 1572, 1573, 1576, 1577, 1583, 1584, 1585, 1587, 1593, 1601, 1609, 1671, 1676, 1677, 1680, 1681, 1682, 1683, 1684, 1687, 1689, 1703, 1706, 1723, 1724, 1725, 1726, 1727, 1731, 1732, 1734, 1735, 1736, 1743, 1745, 1746, 1747, 1748, 1750, 1754, 1757, 1758, 1765, 1776, 1799, 1845, 1859, 1875, 1883, 1893, 1894, 1898, 1899, 1900, 1918, 1943, 1963, 1972, 1975, 1976, 1978, 1980, 1984, 1985, 1990, 1992, 2002, 2004, 2005, 2006, 2008, 2025, 2035, 2036, 2047, 2048, 2049, 2054, 2090, 2091, 2108, 2124, 2158, 2164, 2165, 2167, 2176, 2187, 2192, 2199, 2208, 2243, 2244, 2265, 2298, 2310, 2319, 2330, 2334, 2335, 2346, 2347, 2372, 2376, 2387, 2388, 2421, 2426, 2458, 2460, 2462, 2463, 2464, 2465, 2476, 2483, 2486, 2494, 2503, 2515, 2520, 2534, 2535, 2536, 2559, 2560, 2561, 2564, 2565, 2568, 2593, 2607, 2608, 2626, 2630, 2631, 2632, 2638, 2640, 2641, 2645, 2649, 2670, 2674, 2675, 2676, 2678, 2679, 2683, 2699, 2700, 2708, 2709, 2717, 2718, 2719, 2720, 2723, 2725, 2736, 2738, 2739, 2740, 2754, 2771, 2776, 2779, 2812, 2813, 2814, 2826, 2829, 2830, 2833, 2837, 2849, 2867, 2877, 2878, 2914, 2931, 2933, 2942, 2944, 2945, 2961, 2972, 3001, 3036, 3038, 3040, 3042, 3048, 3055, 3068, 3074, 3082, 3083, 3086, 3087, 3090, 3091, 3092, 3099, 3115, 3138, 3140, 3145, 3154, 3155, 3156, 3158, 3174, 3175, 3177, 3217, 3222, 3224, 3225, 3227, 3228, 3229, 3245, 3247, 3248, 3255, 3287, 3302, 3310, 3311, 3312, 3313, 3314, 3321, 3330, 3341, 3357, 3372, 3383, 3400, 3402, 3403, 3404, 3432, 3454, 3457, 3465, 3490, 3528, 3542, 3554, 3557, 3559, 3560, 3578, 3580, 3625, 3645, 3653, 3654, 3665, 3674, 3700, 3703, 3704, 3708, 3709, 3710, 3711, 3712, 3713, 3753, 3754, 3755, 3756, 3757, 3758, 3816, 3817, 3818, 3819, 3820, 3830, 3850, 3858, 3860, 3861, 3862, 3864, 3865, 3872, 3889, 3890, 3891, 3892, 3893, 3895, 3896, 3899, 3954, 3989, 3996, 4001, 4003, 4004, 4005, 4059, 4064, 4065, 4107, 4108, 4109, 4116, 4125, 4128, 4130, 4131, 4145, 4147, 4168, 4209, 4210, 4211, 4212, 4222, 4272, 4273, 4276, 4277, 4281, 4282, 4283, 4295, 4298, 4309, 4310, 4311, 4349, 4397, 4414, 4415, 4416, 4419, 4421, 4422, 4423, 4425, 4430, 4431, 4433, 4454, 4463, 4488, 4489, 4493, 4513, 4514, 4515, 4522, 4532, 4533, 4539, 4543, 4553, 4554, 4601, 4630, 4636, 4637, 4638, 4643, 4644, 4645, 4666, 4687, 4688, 4689, 4698, 4699, 4700, 4716, 4717, 4727, 4734, 4737, 4747, 4756, 4757, 4812, 4813, 4854, 4855, 4857, 4858, 4867, 4869, 4872, 4902, 4903, 4904, 4907, 4922, 4923, 4928, 4929, 4930, 4954, 4955, 4959, 4960, 4961, 4962, 4963, 4967, 4986, 4994, 4996, 4997, 5002, 5024, 5025, 5026, 5027, 5061, 5094, 5135, 5159, 5187, 5188, 5190, 5241, 5242, 5249, 5259, 5260, 5282, 5295, 5310, 5312, 5313, 5417, 5418, 5419, 5435, 5449, 5450, 5451, 5467, 5483, 5491, 5492, 5495, 5505, 5514, 5544, 5546, 5547, 5550, 5555, 5562, 5563, 5577, 5625, 5627, 5629, 5647, 5714, 5715, 5716, 5722, 5723, 5724, 5725, 5726, 5735, 5754, 5763, 5774, 5778, 5779, 5781, 5790, 5837, 5848, 5856, 5870, 5878, 5882, 5913, 5914, 5939, 5948, 5981, 5983, 5984, 6035, 6039, 6040, 6054, 6056, 6080, 6094, 6098, 6100, 6109, 6120, 6121, 6172, 6174, 6219, 6224, 6225, 6270, 6271, 6272, 6273, 6292]
Community 3: Size 558, Nodes: [0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 23, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 38, 39, 40, 41, 42, 43, 44, 47, 48, 51, 52, 53, 54, 55, 56, 57, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96, 97, 98, 99, 100, 101, 105, 109, 111, 112, 113, 114, 115, 116, 117, 118, 119, 120, 121, 122, 123, 124, 125, 126, 127, 128, 129, 130, 131, 133, 134, 135, 136, 137, 138, 139, 140, 141, 142, 145, 146, 147, 148, 149, 150, 151, 152, 153, 154, 155, 156, 157, 158, 159, 160, 161, 162, 163, 164, 165, 166, 167, 168, 169, 170, 171, 172, 173, 174, 175, 176, 177, 178, 179, 180, 181, 182, 183, 185, 186, 187, 188, 189, 190, 191, 192, 193, 194, 195, 196, 197, 199, 200, 201, 202, 203, 204, 205, 206, 207, 208, 209, 210, 212, 213, 214, 215, 216, 217, 218, 219, 220, 221, 222, 223, 224, 225, 226, 227, 228, 229, 230, 231, 232, 233, 234, 235, 236, 237, 238, 239, 240, 241, 242, 243, 244, 245, 246, 249, 250, 251, 254, 255, 257, 259, 260, 263, 264, 265, 266, 267, 271, 272, 273, 277, 278, 279, 280, 281, 282, 283, 284, 285, 286, 290, 291, 292, 293, 295, 296, 297, 298, 299, 300, 301, 302, 303, 304, 305, 306, 307, 308, 309, 310, 311, 312, 313, 314, 315, 318, 319, 320, 321, 322, 323, 324, 325, 326, 327, 357, 380, 395, 396, 397, 398, 400, 401, 402, 403, 404, 405, 406, 407, 408, 410, 411, 412, 413, 414, 415, 416, 417, 439, 446, 448, 465, 471, 472, 473, 474, 475, 476, 477, 478, 479, 480, 481, 482, 483, 484, 485, 486, 487, 488, 490, 491, 492, 493, 494, 496, 498, 499, 500, 501, 502, 503, 504, 505, 506, 508, 509, 510, 511, 512, 513, 514, 515, 516, 517, 519, 520, 521, 522, 523, 525, 526, 527, 528, 530, 531, 532, 533, 534, 535, 536, 537, 538, 539, 540, 541, 542, 543, 544, 545, 546, 547, 548, 549, 552, 553, 554, 556, 565, 566, 567, 569, 570, 571, 572, 573, 574, 575, 576, 578, 579, 580, 582, 583, 584, 585, 587, 589, 590, 591, 592, 593, 594, 601, 602, 603, 604, 605, 606, 607, 608, 610, 611, 612, 613, 614, 615, 617, 618, 620, 621, 623, 624, 634, 641, 648, 649, 650, 651, 652, 653, 654, 655, 656, 658, 660, 661, 662, 663, 666, 667, 668, 669, 670, 675, 676, 679, 681, 682, 683, 684, 685, 725, 727, 749, 751, 753, 754, 758, 763, 765, 767, 778, 779, 783, 785, 787, 791, 792, 804, 828, 846, 904, 905, 997, 1135, 1192, 1193, 1195, 1445, 1641, 1654, 1825, 1862, 1983, 2013, 2046, 2055, 2138, 2152, 2153, 2160, 2170, 2206, 2217, 2373, 2374, 2390, 2391, 2398, 2403, 2414, 2415, 2437, 2439, 2440, 2441, 2442, 2444, 2446, 2572, 2657, 2666, 2701, 2702, 2704, 2806, 2820, 2844, 2855, 2898, 2940, 2941, 2946, 2948, 3152, 3153, 3563, 3975, 3987, 4226, 4227, 4241, 4296, 4429, 4443, 4595, 4596, 4654, 4655, 4675, 4723, 4724, 4725, 4861, 4862, 5090, 5382, 5652, 5733, 5925, 5934, 5997, 6044]
Community 4: Size 549, Nodes: [19, 106, 262, 338, 351, 353, 354, 356, 362, 364, 365, 366, 409, 428, 429, 442, 459, 461, 470, 507, 551, 586, 596, 619, 629, 631, 639, 688, 701, 771, 805, 819, 852, 853, 855, 857, 879, 881, 901,

902, 909, 918, 948, 963, 969, 970, 973, 980, 982, 985, 991, 993, 998, 1002, 1077, 1116, 1132, 1157, 1175, 1182, 1188, 1197, 1241, 1244, 1246, 1247, 1250, 1251, 1252, 1253, 1276, 1290, 1320, 1326, 1336, 1338, 1339, 1344, 1365, 1378, 1398, 1399, 1400, 1401, 1402, 1403, 1404, 1406, 1408, 1409, 1422, 1436, 1472, 1476, 1487, 1552, 1553, 1554, 1555, 1557, 1599, 1606, 1607, 1611, 1614, 1617, 1619, 1621, 1622, 1626, 1627, 1628, 1631, 1638, 1639, 1642, 1643, 1644, 1647, 1650, 1651, 1655, 1658, 1667, 1696, 1728, 1740, 1744, 1768, 1779, 1787, 1797, 1812, 1838, 1847, 1848, 1849, 1850, 1851, 1853, 1854, 1855, 1856, 1857, 1871, 1872, 1874, 1912, 1922, 1926, 1952, 1966, 1977, 2026, 2027, 2028, 2031, 2032, 2034, 2044, 2051, 2052, 2056, 2060, 2061, 2062, 2064, 2065, 2079, 2081, 2097, 2103, 2131, 2186, 2212, 2263, 2270, 2303, 2308, 2311, 2327, 2344, 2353, 2379, 2404, 2424, 2425, 2438, 2448, 2449, 2454, 2469, 2492, 2507, 2518, 2521, 2523, 2525, 2526, 2527, 2528, 2539, 2542, 2544, 2545, 2547, 2548, 2567, 2579, 2615, 2617, 2651, 2652, 2654, 2655, 2656, 2682, 2684, 2685, 2686, 2687, 2688, 2694, 2705, 2706, 2707, 2710, 2714, 2715, 2721, 2726, 2727, 2728, 2729, 2730, 2731, 2733, 2734, 2735, 2743, 2744, 2746, 2747, 2755, 2763, 2784, 2785, 2787, 2788, 2789, 2790, 2796, 2797, 2799, 2801, 2807, 2808, 2810, 2825, 2827, 2858, 2859, 2882, 2883, 2938, 2947, 2956, 2966, 2989, 2991, 2993, 2994, 3012, 3024, 3025, 3028, 3029, 3031, 3032, 3033, 3034, 3043, 3094, 3096, 3127, 3142, 3163, 3165, 3242, 3294, 3298, 3350, 3351, 3355, 3356, 3359, 3362, 3386, 3388, 3389, 3390, 3391, 3397, 3399, 3410, 3416, 3417, 3423, 3424, 3426, 3427, 3428, 3449, 3456, 3475, 3476, 3479, 3482, 3483, 3484, 3485, 3486, 3521, 3522, 3523, 3526, 3537, 3538, 3539, 3540, 3541, 3544, 3545, 3546, 3550, 3553, 3556, 3565, 3566, 3569, 3570, 3571, 3589, 3594, 3608, 3613, 3614, 3615, 3619, 3621, 3623, 3633, 3634, 3642, 3647, 3655, 3656, 3666, 3667, 3668, 3680, 3693, 3694, 3695, 3697, 3759, 3795, 3796, 3797, 3799, 3800, 3801, 3802, 3803, 3804, 3805, 3806, 3808, 3809, 3810, 3811, 3814, 3815, 3825, 3826, 3866, 3867, 3886, 3928, 4009, 4033, 4066, 4069, 4070, 4071, 4132, 4148, 4149, 4156, 4158, 4159, 4161, 4162, 4167, 4177, 4179, 4180, 4188, 4219, 4235, 4247, 4250, 4251, 4253, 4254, 4257, 4263, 4264, 4265, 4305, 4306, 4324, 4327, 4345, 4368, 4386, 4402, 4411, 4424, 4437, 4446, 4447, 4448, 4449, 4452, 4453, 4469, 4473, 4517, 4518, 4520, 4530, 4550, 4551, 4555, 4556, 4558, 4559, 4560, 4561, 4563, 4568, 4569, 4571, 4587, 4588, 4589, 4600, 4662, 4670, 4685, 4739, 4743, 4748, 4749, 4750, 4751, 4752, 4754, 4755, 4758, 4759, 4766, 4767, 4769, 4780, 4781, 4782, 4784, 4800, 4826, 4905, 4966, 4971, 5054, 5055, 5096, 5140, 5141, 5165, 5167, 5168, 5186, 5220, 5221, 5266, 5281, 5321, 5322, 5345, 5348, 5351, 5372, 5459, 5488, 5489, 5490, 5497, 5499, 5506, 5548, 5557, 5592, 5595, 5596, 5603, 5612, 5613, 5614, 5621, 5622, 5650, 5664, 5711, 5712, 5761, 5769, 5788, 5805, 5836, 5840, 5841, 5857, 5858, 5864, 5931, 5932, 5952, 5985, 5986, 6038, 6092, 6093, 6099, 6105, 6106, 6107, 6237, 6283]
Community 5: Size 390, Nodes: [18, 24, 198, 270, 345, 348, 352, 392, 558, 559, 642, 689, 698, 741, 768, 797, 829, 830, 841, 843, 858, 876, 890, 891, 896, 947, 954, 955, 1006, 1019, 1038, 1043, 1057, 1112, 1138, 1146, 1173, 1177, 1189, 1236, 1255, 1283, 1291, 1303, 1330, 1331, 1359, 1377, 1385, 1418, 1434, 1450, 1463, 1465, 1466, 1467, 1468, 1477, 1483, 1495, 1496, 1498, 1499, 1501, 1515, 1523, 1524, 1525, 1526, 1528, 1563, 1605, 1653, 1668, 1679, 1693, 1700, 1704, 1719, 1749, 1753, 1790, 1794, 1865, 1880, 1911, 1913, 1933, 1962, 1968, 1970, 1971, 1993, 1994, 1995, 1999, 2000, 2016, 2017, 2021, 2058, 2085, 2116, 2133, 2140, 2142, 2143, 2144, 2147, 2172, 2173, 2174, 2175, 2177, 2178, 2232, 2234, 2286, 2292, 2338, 2369, 2407, 2416, 2430, 2432, 2445, 2451, 2481, 2506, 2508, 2510, 2512, 2543, 2557, 2570, 2605, 2610, 2612, 2613, 2614, 2775, 2791, 2792, 2794, 2795, 2803, 2815, 2869, 2880, 2932, 2967, 2996, 2997, 3058, 3059, 3076, 3077, 3078, 3079, 3080, 3088, 3093, 3100, 3103, 3104, 3105, 3108, 3116, 3118, 3178, 3215, 3305, 3354, 3367, 3382, 3387, 3392, 3401, 3419, 3430, 3446, 3471, 3480, 3487, 3488, 3489, 3497, 3499, 3579, 3588, 3591, 3644, 3706, 3812, 3831, 3832, 3833, 3882, 3887, 3888, 3912, 3913, 3927, 3959, 3961, 3962, 3963, 3969, 3970, 3974, 3982, 3984, 3997, 4014, 4034, 4040, 4054, 4055, 4099, 4100, 4101, 4102, 4126, 4129, 4134, 4206, 4207, 4208, 4217, 4248, 4259, 4291, 4294, 4325, 4331, 4332, 4333, 4335, 4364, 4366, 4367, 4369, 4375, 4380, 4391, 4393, 4394, 4395, 4396, 4401, 4407, 4408, 4434, 4440, 4465, 4466, 4474, 4476, 4477, 4478, 4483, 4490, 4502, 4504, 4509, 4526, 4573, 4580, 4581, 4582, 4583, 4586, 4692, 4696, 4697, 4701, 4702, 4714, 4731, 4741, 4842, 4843, 4844, 4911, 4912, 4913, 4914, 4915, 4916, 4917, 4931, 4932, 4933, 4934, 4935, 4983, 4984, 5008, 5009, 5010, 5011, 5012, 5013, 5015, 5016, 5017, 5018, 5019, 5052, 5053, 5088, 5089, 5110, 5111, 5112, 5129, 5174, 5182, 5183, 5185, 5191, 5192, 5193, 5203, 5204, 5222, 5224, 5250, 5255, 5256, 5258, 5273, 5280, 5283, 5307, 5308, 5309, 5311, 5314, 5347, 5363, 5384, 5385, 5389, 5390, 5391, 5394, 5395, 5401, 5456, 5457, 5458, 5461, 5462, 5464, 5466, 5484, 5585, 5589, 5606, 5607, 5616, 5619, 5620, 5626, 5634, 5669, 5670, 5709, 5771, 5803, 5804, 5842, 5843, 5844, 5869, 5872, 5873, 5879, 5915, 5926, 5942, 5943, 5944, 6009, 6010, 6021, 6022, 6027, 6028]
Community 6: Size 387, Nodes: [22, 252, 261, 269, 275, 276, 287, 288, 289, 381, 383, 384, 386, 387, 388, 389, 390, 391, 399, 419, 444, 529, 561, 598, 633, 693, 695, 697, 699, 700, 702, 706, 707, 716, 728, 742, 745, 746, 807, 808, 809, 812, 813, 814, 816, 818, 821, 849, 851, 862, 864, 865, 866, 872, 893, 911, 1003, 1011, 1012, 1015, 1025, 1070, 1074, 1108, 1154, 1168, 1222, 1223, 1226, 1229, 1248, 1256, 1275, 1279, 1322, 1346, 1351, 1375, 1381, 1421, 1424, 1425, 1474, 1494, 1516, 1590, 1632, 1635, 1636, 1661, 1662, 1666, 1670, 1673, 1674, 1675, 1730, 1739, 1767, 1769, 1770, 1772, 1773, 1800, 1801, 1804, 1823, 1833, 1869, 1916, 1917, 1919, 1920, 1921, 1935, 1937, 1954, 1973, 1986, 2024, 2029, 2042, 2057, 2071, 2100, 2157, 2161, 2182, 2231, 2235, 2239, 2267, 2280, 2283, 2325, 2326, 2356, 2366, 2370, 2393, 2409, 2491, 2497, 2513, 2517, 2530, 2533, 2540, 2577, 2600, 2618, 2619, 2620, 2621, 2622, 2623, 2624, 2625, 2627, 2634, 2635, 2636, 2660, 2703, 2742, 2783, 2822, 2832, 2836, 2846, 2866, 2868, 2887, 2888, 2908, 2909, 2910, 2911, 2912, 2939, 2998, 2999, 3119, 3146, 3147, 3148, 3149, 3208, 3249, 3251, 3252, 3278, 3295, 3297, 3303, 3306, 3308, 3334, 3368, 3418, 3437, 3442, 3455, 3516, 3525, 3555, 3572, 3592, 3602, 3639, 3670, 3673, 3683, 3692, 3717, 3718, 3719, 3720, 3721, 3760, 3761, 3770, 3779, 3851, 3897, 3949, 3991, 3994, 3995, 4021, 4023, 4024, 4026, 4028, 4032, 4052, 4056, 4088, 4103, 4110, 4118, 4119, 4140, 4141, 4143, 4172, 4218, 4223, 4224, 4258, 4290, 4307, 4308, 4315, 4316, 4317, 4320, 4321, 4340, 4346, 4360, 4383, 4387, 4435, 4441, 4442, 4458, 4487, 4498, 4499, 4500, 4566, 4592, 4646, 4677, 4718, 4719, 4738, 4753, 4817, 4848, 4850, 4873, 4874, 4875, 4876, 4877, 4878, 4879, 4880, 4881, 4882, 4883, 4884, 4885, 4886, 4887, 4888, 4889, 4890, 4891, 4893, 4894, 4895, 4896, 4897, 4898, 4899, 4918, 4944, 4964, 5044, 5058, 5059, 5060, 5068, 5069, 5070, 5082, 5166, 5200, 5298, 5299, 5300, 5340, 5374, 5375, 5376, 5377, 5416, 5421, 5486, 5516, 5520, 5532, 5533, 5534, 5549, 5568, 5575, 5588, 5590, 5591, 5594, 5665, 5677, 5710, 5731, 5747, 5748, 5750, 5751, 5770, 5772, 5789, 5800, 5801, 5802, 5852, 5860, 5861, 5875, 5885, 5920, 5935, 5937, 5938, 5990, 5999, 6000, 6001, 6002, 6003, 6004, 6005, 6065, 6066, 6067, 6068, 6104, 6122, 6123, 6140, 6141, 6184, 6201, 6202]
Community 7: Size 371, Nodes: [294, 341, 350, 358, 359, 361, 435, 445, 563, 731, 781, 795, 871, 874, 887, 895, 907, 915, 925, 939, 943, 950, 953, 960, 977, 981, 992, 1100, 1111, 1142, 1156, 1172, 1210, 1213, 1230, 1267, 1372, 1374, 1380, 1383, 1393, 1444, 1451, 1508, 1511, 1532, 1540, 1542, 1543, 1544, 1545, 1581, 1582, 1600, 1646, 1660, 1669, 1707, 1721, 1738, 1761, 1782, 1806, 1807, 1808, 1809, 1810, 1811, 1831, 1891, 1901, 1902, 1929, 1931, 2003, 2039, 2059, 2105, 2107, 2110, 2113, 2126, 2154, 2155, 2171, 2180, 2181, 2183, 2196, 2205, 2220, 2222, 2224, 2238, 2241, 2242, 2248, 2266, 2296, 2300, 2304, 2305, 2306, 2307, 2321, 2323, 2332, 2336, 2339, 2355, 2375, 2400, 2406, 2428, 2429, 2467, 2477, 2478, 2479, 2493, 2496, 2498, 2499, 2516, 2642, 2643, 2648, 2673, 2711, 2732, 2756, 2759, 2760, 2761, 2762, 2764, 2777, 2778, 2781, 2782, 2798, 2802, 2834, 2835, 2847, 2848, 2850, 2851, 2852, 2853, 2919, 2920, 2921, 2922, 2924, 2925, 2926, 2937, 2949, 2960, 2969, 2970, 2971, 2983, 2984, 2988, 2995, 3000, 3002, 3003, 3004, 3005, 3030, 3102, 3121, 3210, 3212, 3213, 3250, 3261, 3270, 3271, 3272, 3273, 3274, 3275, 3276, 3277, 3279, 3280, 3283, 3284, 3288, 3290, 3292, 3293, 3300, 3346, 3371, 3405, 3515, 3561, 3562, 3593, 3601, 3622, 3624, 3651, 3661, 3662, 3663, 3664, 3672, 3762, 3821, 3914, 3915, 3941, 3983, 4045, 4047, 4048, 4050, 4113, 4165, 4171, 4191, 4193, 4194, 4244, 4245, 4246, 4313, 4334, 4426, 4495, 4496, 4505, 4506, 4507, 4508, 4521,

4523, 4540, 4641, 4671, 4676, 4679, 4730, 4732, 4733, 4773, 4792, 4793, 4794, 4795, 4859, 4860, 4892, 4920, 4925
, 4927, 5051, 5097, 5128, 5134, 5225, 5226, 5233, 5247, 5248, 5254, 5350, 5397, 5415, 5420, 5439, 5441, 5442, 54
47, 5508, 5509, 5510, 5511, 5512, 5513, 5523, 5524, 5525, 5556, 5570, 5583, 5584, 5598, 5599, 5600, 5610, 5611,
5623, 5648, 5675, 5680, 5682, 5683, 5684, 5685, 5692, 5695, 5696, 5697, 5699, 5700, 5701, 5702, 5718, 5737, 5738
, 5739, 5740, 5741, 5742, 5743, 5749, 5755, 5756, 5758, 5786, 5791, 5792, 5793, 5834, 5838, 5839, 5888, 5889, 58
90, 5954, 5967, 5968, 6006, 6023, 6029, 6030, 6031, 6032, 6037, 6075, 6076, 6077, 6078, 6079, 6142, 6143, 6144,
6145, 6146, 6153, 6154, 6155, 6159, 6185, 6186, 6223, 6233, 6275, 6276, 6293, 6294, 6298, 6299, 6300]
Community 8: Size 318, Nodes: [316, 343, 424, 431, 452, 497, 557, 637, 640, 678, 690, 718, 786, 824, 839, 842, 8
67, 868, 875, 882, 897, 935, 942, 945, 956, 990, 1018, 1021, 1022, 1023, 1024, 1026, 1051, 1079, 1081, 1110, 111
7, 1119, 1122, 1124, 1166, 1183, 1199, 1209, 1217, 1259, 1282, 1286, 1329, 1334, 1337, 1355, 1411, 1423, 1446, 1
491, 1500, 1521, 1566, 1569, 1595, 1629, 1652, 1664, 1694, 1698, 1699, 1742, 1756, 1826, 1867, 1870, 1873, 1877,
1923, 1964, 2007, 2014, 2019, 2022, 2063, 2077, 2118, 2125, 2127, 2129, 2134, 2139, 2151, 2168, 2169, 2189, 2195
, 2201, 2209, 2271, 2272, 2273, 2274, 2278, 2297, 2312, 2333, 2341, 2359, 2363, 2385, 2514, 2549, 2637, 2650, 27
51, 2767, 2818, 2856, 2875, 2884, 2893, 2915, 2916, 2918, 3019, 3021, 3022, 3026, 3037, 3056, 3070, 3075, 3089,
3097, 3101, 3161, 3299, 3309, 3317, 3318, 3331, 3422, 3494, 3507, 3536, 3547, 3567, 3585, 3586, 3587, 3595, 3597
, 3598, 3603, 3604, 3605, 3609, 3610, 3612, 3648, 3684, 3702, 3724, 3725, 3727, 3728, 3766, 3767, 3807, 3813, 38
81, 3916, 3917, 3919, 3920, 3921, 3922, 3923, 3924, 3960, 4020, 4022, 4042, 4049, 4051, 4075, 4122, 4124, 4138,
4142, 4150, 4151, 4152, 4153, 4155, 4242, 4256, 4267, 4268, 4286, 4358, 4363, 4410, 4450, 4451, 4460, 4470, 4471
, 4503, 4534, 4574, 4668, 4691, 4735, 4740, 4774, 4783, 4798, 4799, 4802, 4803, 4809, 4839, 4840, 4841, 4865, 48
66, 4868, 4908, 4910, 4919, 4968, 4982, 4998, 4999, 5000, 5048, 5050, 5072, 5073, 5074, 5093, 5098, 5099, 5119,
5120, 5121, 5160, 5175, 5189, 5227, 5228, 5230, 5243, 5244, 5245, 5246, 5268, 5269, 5270, 5271, 5328, 5329, 5330
, 5331, 5332, 5334, 5335, 5336, 5337, 5338, 5366, 5367, 5426, 5427, 5448, 5485, 5494, 5540, 5551, 5560, 5604, 56
17, 5618, 5681, 5744, 5773, 5777, 5780, 5784, 5785, 5794, 5816, 5821, 5822, 5823, 5824, 5826, 5827, 5831, 5867,
5868, 5883, 5887, 5892, 5911, 5921, 5936, 5994, 5995, 5996, 6069, 6070, 6071, 6087, 6091, 6175, 6203, 6227, 6228
, 6274]
Community 9: Size 303, Nodes: [102, 418, 425, 441, 555, 599, 625, 694, 715, 739, 777, 789, 793, 860, 870, 873, 8
78, 894, 900, 914, 984, 1044, 1058, 1137, 1139, 1143, 1196, 1201, 1410, 1415, 1432, 1527, 1604, 1610, 1615, 1659
, 1672, 1751, 1760, 1793, 1828, 1835, 1852, 1864, 1878, 1885, 1904, 1927, 1939, 1955, 1956, 1957, 1958, 1961, 19
65, 1982, 2009, 2011, 2012, 2053, 2104, 2130, 2145, 2146, 2188, 2202, 2251, 2255, 2290, 2324, 2371, 2386, 2389,
2399, 2411, 2419, 2427, 2435, 2450, 2489, 2538, 2553, 2574, 2575, 2576, 2578, 2580, 2591, 2594, 2596, 2659, 2664
, 2680, 2681, 2716, 2748, 2780, 2809, 2876, 2886, 2923, 2927, 2929, 2951, 2952, 2953, 2954, 2955, 2979, 3013, 30
14, 3015, 3016, 3062, 3107, 3128, 3167, 3192, 3197, 3199, 3200, 3201, 3202, 3209, 3214, 3218, 3220, 3230, 3231,
3236, 3237, 3240, 3262, 3263, 3264, 3265, 3266, 3267, 3327, 3332, 3395, 3470, 3473, 3502, 3504, 3505, 3527, 3584
, 3590, 3599, 3600, 3606, 3618, 3627, 3628, 3629, 3631, 3632, 3637, 3640, 3699, 3769, 3794, 3798, 3829, 3844, 38
59, 3934, 3935, 3936, 3937, 3938, 3939, 3972, 3985, 4057, 4123, 4182, 4234, 4243, 4249, 4301, 4347, 4348, 4379,
4389, 4439, 4484, 4485, 4494, 4497, 4516, 4519, 4546, 4565, 4584, 4591, 4594, 4693, 4704, 4720, 4775, 4804, 4805
, 4806, 4807, 4808, 4810, 4818, 4819, 4965, 4970, 5003, 5021, 5032, 5033, 5035, 5036, 5038, 5062, 5117, 5130, 51
42, 5143, 5144, 5145, 5155, 5156, 5157, 5158, 5216, 5275, 5285, 5290, 5302, 5303, 5333, 5373, 5379, 5396, 5400,
5402, 5425, 5452, 5519, 5564, 5565, 5566, 5573, 5574, 5580, 5602, 5608, 5609, 5645, 5666, 5667, 5668, 5691, 5698
, 5703, 5728, 5729, 5730, 5746, 5818, 5835, 5845, 5901, 5933, 5969, 5970, 5971, 5972, 5977, 5978, 5979, 5980, 59
87, 5998, 6049, 6061, 6062, 6095, 6131, 6132, 6133, 6170, 6171, 6182, 6187, 6188, 6189, 6190, 6191, 6208, 6209,
6220, 6221, 6222, 6280, 6290, 6291]
Community 10: Size 271, Nodes: [58, 84, 103, 247, 346, 370, 378, 426, 427, 449, 453, 626, 645, 657, 692, 704, 70
9, 723, 799, 861, 919, 999, 1000, 1001, 1016, 1066, 1071, 1080, 1123, 1245, 1261, 1293, 1328, 1332, 1340, 1390,
1391, 1407, 1427, 1462, 1507, 1578, 1678, 1701, 1733, 1737, 1741, 1752, 1759, 1818, 1868, 1876, 1932, 1987, 1988
, 1996, 2001, 2033, 2050, 2068, 2093, 2094, 2095, 2096, 2132, 2210, 2216, 2221, 2226, 2229, 2230, 2233, 2236, 22
85, 2301, 2302, 2383, 2408, 2417, 2433, 2447, 2459, 2468, 2475, 2484, 2522, 2537, 2546, 2554, 2555, 2563, 2573,
2586, 2587, 2589, 2590, 2603, 2633, 2662, 2665, 2677, 2692, 2693, 2695, 2696, 2697, 2698, 2805, 2857, 2872, 2885
, 3006, 3067, 3072, 3095, 3141, 3205, 3206, 3221, 3260, 3333, 3378, 3379, 3381, 3384, 3385, 3396, 3398, 3406, 34
07, 3447, 3466, 3617, 3636, 3714, 3715, 3716, 3822, 3823, 3824, 3838, 3843, 3849, 3857, 3873, 3905, 3940, 3944,
3956, 3958, 3964, 3965, 3966, 3971, 3973, 4058, 4105, 4106, 4117, 4127, 4146, 4154, 4173, 4174, 4176, 4178, 4303
, 4312, 4355, 4362, 4445, 4468, 4472, 4511, 4512, 4529, 4541, 4547, 4548, 4552, 4575, 4585, 4613, 4614, 4615, 46
16, 4651, 4652, 4690, 4765, 4768, 4832, 4849, 4900, 4901, 4906, 4989, 4995, 5075, 5076, 5077, 5078, 5079, 5080,
5081, 5084, 5085, 5086, 5087, 5201, 5202, 5213, 5214, 5215, 5217, 5218, 5219, 5292, 5365, 5393, 5403, 5404, 5405
, 5453, 5482, 5487, 5500, 5501, 5507, 5515, 5527, 5636, 5637, 5678, 5694, 5734, 5736, 5764, 5766, 5767, 5768, 57
96, 5811, 5832, 5833, 5853, 5886, 5891, 5894, 5904, 5940, 5941, 5946, 5947, 6033, 6034, 6042, 6043, 6055, 6096,
6112, 6113, 6114, 6116, 6117, 6127, 6128, 6129, 6130, 6139, 6152]
Community 11: Size 258, Nodes: [489, 659, 744, 775, 796, 806, 883, 884, 913, 941, 1010, 1073, 1167, 1214, 1215,
1218, 1219, 1249, 1311, 1312, 1313, 1314, 1315, 1316, 1319, 1321, 1342, 1347, 1349, 1366, 1367, 1392, 1394, 1464
, 1502, 1534, 1549, 1550, 1575, 1613, 1645, 1665, 1705, 1720, 1729, 1839, 1840, 1860, 1925, 1930, 1938, 2045, 20
80, 2087, 2148, 2162, 2163, 2185, 2218, 2277, 2340, 2381, 2410, 2482, 2509, 2558, 2571, 2588, 2592, 2689, 2713,
2753, 2786, 2845, 2860, 2861, 2862, 2863, 2864, 2865, 2906, 2930, 3027, 3035, 3041, 3134, 3223, 3246, 3268, 3269
, 3329, 3358, 3444, 3462, 3474, 3481, 3493, 3506, 3508, 3524, 3576, 3596, 3611, 3643, 3682, 3686, 3688, 3696, 37
01, 3735, 3736, 3737, 3738, 3739, 3740, 3741, 3742, 3749, 3768, 3771, 3784, 3785, 3828, 3839, 3840, 3841, 3842,
3885, 3900, 3906, 3907, 3908, 3909, 3925, 3929, 3933, 3967, 3968, 3976, 3977, 3978, 3979, 3980, 4029, 4063, 4067
, 4081, 4082, 4083, 4084, 4085, 4086, 4089, 4090, 4091, 4092, 4093, 4098, 4157, 4163, 4213, 4214, 4255, 4269, 42
74, 4323, 4337, 4339, 4350, 4351, 4352, 4361, 4399, 4403, 4412, 4413, 4456, 4457, 4535, 4639, 4647, 4653, 4736,
4833, 4834, 4835, 4836, 4837, 4838, 4909, 4979, 5023, 5065, 5066, 5067, 5095, 5122, 5125, 5126, 5169, 5170, 5171
, 5172, 5177, 5234, 5235, 5236, 5240, 5359, 5360, 5380, 5381, 5388, 5475, 5480, 5535, 5536, 5567, 5640, 5641, 56
54, 5655, 5656, 5657, 5658, 5676, 5693, 5752, 5760, 5795, 5813, 5855, 5881, 5910, 5955, 5956, 5957, 5958, 5959,
5991, 5992, 5993, 6050, 6051, 6073, 6074, 6101, 6102, 6118, 6119, 6160, 6161, 6162, 6163, 6164, 6205, 6206, 6207
]
Community 12: Size 257, Nodes: [329, 355, 373, 433, 643, 674, 732, 801, 802, 820, 833, 840, 848, 946, 971, 1060,
1089, 1120, 1121, 1140, 1200, 1221, 1242, 1295, 1296, 1297, 1300, 1419, 1438, 1490, 1538, 1579, 1580, 1592, 1597
, 1602, 1616, 1623, 1634, 1637, 1712, 1713, 1714, 1715, 1716, 1717, 1718, 1775, 1796, 1805, 1836, 1866, 1914, 19
47, 1969, 1997, 1998, 2128, 2204, 2225, 2249, 2268, 2413, 2431, 2455, 2456, 2466, 2487, 2490, 2500, 2541, 2550,
2585, 2602, 2800, 2879, 2889, 2963, 2965, 2977, 3023, 3052, 3073, 3085, 3122, 3159, 3176, 3253, 3338, 3343, 3352
, 3360, 3361, 3369, 3472, 3520, 3529, 3530, 3531, 3532, 3533, 3534, 3535, 3630, 3641, 3649, 3650, 3652, 3658, 36
59, 3660, 3671, 3689, 3743, 3744, 3772, 3773, 3999, 4011, 4012, 4013, 4017, 4044, 4215, 4225, 4260, 4261, 4262,
4280, 4314, 4319, 4353, 4372, 4373, 4374, 4377, 4378, 4392, 4400, 4438, 4459, 4480, 4481, 4482, 4510, 4542, 4564
, 4603, 4604, 4605, 4606, 4607, 4608, 4612, 4678, 4686, 4703, 4705, 4711, 4713, 4726, 4764, 4791, 4946, 4948, 49
72, 4980, 4985, 5071, 5092, 5136, 5137, 5138, 5184, 5205, 5206, 5207, 5208, 5209, 5210, 5211, 5212, 5229, 5251,
5253, 5276, 5305, 5319, 5320, 5325, 5413, 5414, 5440, 5476, 5477, 5478, 5498, 5521, 5522, 5545, 5553, 5554, 5559

```
, 5561, 5593, 5605, 5628, 5633, 5642, 5643, 5644, 5651, 5717, 5719, 5720, 5745, 5775, 5776, 5782, 5849, 5850, 58
51, 5865, 5866, 5897, 5898, 5899, 5900, 5906, 5907, 5908, 5909, 5918, 5919, 5988, 5989, 6045, 6052, 6088, 6089,
6090, 6110, 6111, 6165, 6166, 6167, 6168, 6169, 6177, 6178, 6210, 6211, 6212, 6213, 6214, 6215, 6216]
Community 13: Size 226, Nodes: [25, 328, 394, 436, 632, 636, 644, 647, 711, 722, 730, 772, 822, 835, 845, 916, 9
17, 965, 1083, 1265, 1368, 1371, 1376, 1435, 1492, 1574, 1586, 1620, 1686, 1708, 1755, 1882, 1928, 1946, 1948, 1
953, 1981, 2098, 2101, 2240, 2260, 2329, 2349, 2392, 2394, 2395, 2396, 2397, 2524, 2531, 2606, 2616, 2628, 2629,
2646, 2647, 2663, 2772, 2773, 2774, 2804, 2897, 2899, 2900, 2901, 2902, 2903, 2904, 2905, 2907, 2980, 2990, 3017
, 3117, 3193, 3194, 3195, 3203, 3239, 3244, 3281, 3296, 3373, 3377, 3380, 3464, 3492, 3495, 3496, 3498, 3509, 35
74, 3620, 3722, 3723, 3734, 3774, 3782, 3835, 3863, 3943, 3950, 3981, 4027, 4120, 4192, 4302, 4341, 4388, 4409,
4444, 4464, 4475, 4544, 4545, 4549, 4609, 4610, 4611, 4623, 4635, 4673, 4674, 4722, 4790, 4801, 4831, 4856, 4870
, 4871, 4926, 5028, 5029, 5030, 5031, 5102, 5103, 5104, 5106, 5107, 5108, 5109, 5123, 5124, 5252, 5349, 5398, 53
99, 5410, 5443, 5444, 5502, 5503, 5504, 5517, 5518, 5529, 5530, 5537, 5539, 5727, 5765, 5846, 5847, 5859, 5902,
5903, 5912, 5927, 5928, 5929, 5930, 5975, 5976, 6026, 6057, 6058, 6059, 6060, 6082, 6083, 6084, 6085, 6086, 6097
, 6199, 6200, 6238, 6239, 6240, 6241, 6242, 6243, 6244, 6245, 6246, 6247, 6248, 6249, 6250, 6251, 6252, 6253, 62
54, 6255, 6256, 6257, 6258, 6259, 6260, 6261, 6262, 6263, 6264, 6265, 6266, 6267, 6268, 6269, 6278, 6279, 6287,
6288, 6289, 6295, 6297]
Community 14: Size 224, Nodes: [143, 463, 616, 628, 638, 794, 803, 847, 869, 889, 908, 959, 972, 979, 1062, 1084
, 1104, 1149, 1310, 1327, 1389, 1413, 1440, 1493, 1505, 1509, 1510, 1512, 1513, 1514, 1562, 1564, 1565, 1567, 15
68, 1690, 1691, 1695, 1697, 1774, 1798, 1803, 1886, 1887, 1889, 1890, 1892, 1906, 1907, 1908, 1909, 1910, 1951,
1967, 2041, 2043, 2078, 2115, 2117, 2119, 2120, 2123, 2135, 2136, 2137, 2159, 2184, 2190, 2194, 2211, 2213, 2214
, 2215, 2223, 2245, 2256, 2257, 2287, 2317, 2331, 2348, 2418, 2488, 2495, 2501, 2502, 2504, 2505, 2658, 2668, 26
69, 2749, 2750, 2752, 2881, 2957, 2958, 2959, 2962, 2964, 2973, 2974, 2975, 2976, 2978, 2981, 2982, 2987, 3098,
3114, 3120, 3196, 3198, 3342, 3408, 3409, 3412, 3413, 3414, 3448, 3518, 3543, 3675, 3676, 3729, 3730, 3731, 3732
, 3733, 3786, 3787, 3788, 3789, 3790, 3791, 3869, 3870, 3871, 3926, 3947, 3948, 4025, 4068, 4087, 4111, 4216, 42
20, 4221, 4304, 4344, 4357, 4382, 4398, 4418, 4576, 4602, 4628, 4629, 4681, 4682, 4694, 4796, 4822, 4823, 4864,
4921, 4945, 4949, 4950, 4952, 4953, 4978, 5113, 5114, 5115, 5116, 5131, 5132, 5139, 5179, 5180, 5264, 5265, 5284
, 5286, 5301, 5306, 5445, 5446, 5460, 5528, 5649, 5671, 5686, 5687, 5688, 5704, 5705, 5706, 5707, 5708, 5762, 58
06, 5807, 5808, 5809, 5810, 5812, 5814, 5877, 5917, 5964, 5965, 5966, 5973, 5974, 6053, 6072, 6108, 6147, 6148,
6149, 6150, 6151]
Community 15: Size 218, Nodes: [37, 248, 336, 347, 374, 443, 595, 724, 726, 811, 834, 885, 888, 892, 929, 962, 9
64, 978, 1075, 1101, 1113, 1194, 1227, 1228, 1278, 1318, 1323, 1354, 1358, 1396, 1461, 1504, 1537, 1558, 1570, 1
589, 1591, 1594, 1596, 1612, 1633, 1656, 1692, 1702, 1762, 1763, 1764, 1777, 1778, 1780, 1781, 1788, 1789, 1792,
1829, 1841, 1842, 1844, 1846, 1861, 1863, 1924, 1959, 2015, 2020, 2030, 2037, 2092, 2149, 2197, 2198, 2228, 2264
, 2275, 2276, 2288, 2291, 2313, 2314, 2315, 2318, 2320, 2322, 2420, 2443, 2470, 2471, 2472, 2473, 2562, 2566, 26
04, 2609, 2765, 2793, 2824, 2828, 2838, 2840, 2841, 2842, 2843, 2891, 2892, 2894, 3018, 3063, 3065, 3066, 3084,
3123, 3124, 3157, 3207, 3211, 3232, 3243, 3259, 3366, 3411, 3451, 3491, 3503, 3510, 3511, 3512, 3513, 3514, 3519
, 3549, 3551, 3568, 3573, 3616, 3626, 3635, 3726, 3750, 3751, 3752, 3834, 3932, 4039, 4041, 4094, 4095, 4112, 41
15, 4175, 4181, 4183, 4184, 4185, 4229, 4292, 4293, 4328, 4329, 4420, 4562, 4567, 4572, 4680, 4683, 4712, 4853,
4863, 4924, 4956, 4957, 4958, 4987, 4988, 5001, 5004, 5005, 5006, 5037, 5039, 5040, 5041, 5042, 5043, 5045, 5046
, 5047, 5049, 5100, 5101, 5267, 5304, 5339, 5371, 5406, 5411, 5412, 5432, 5481, 5576, 5597, 5659, 5721, 5815, 58
62, 5893, 5945, 5960, 5961, 6156, 6157, 6158, 6179, 6180, 6181, 6183, 6217, 6218, 6296]
Community 16: Size 128, Nodes: [423, 630, 665, 743, 844, 880, 898, 903, 983, 1013, 1082, 1159, 1191, 1208, 1237,
1243, 1262, 1301, 1317, 1352, 1353, 1478, 1480, 1481, 1482, 1485, 1486, 1522, 1649, 1783, 1814, 1815, 1816, 1817
, 1884, 1944, 2018, 2040, 2076, 2121, 2358, 2474, 2480, 2485, 2529, 2599, 2854, 2992, 3144, 3219, 3233, 3234, 32
35, 3282, 3289, 3316, 3349, 3548, 3577, 3607, 3646, 3657, 3669, 3685, 3687, 3746, 3747, 3763, 3764, 3765, 3780,
3781, 3845, 3846, 3847, 3901, 3902, 3903, 3904, 3930, 3931, 3942, 3990, 3993, 4036, 4144, 4166, 4186, 4284, 4342
, 4343, 4479, 4525, 4557, 4579, 4593, 4617, 4618, 4619, 4624, 4625, 4626, 4627, 4707, 4708, 4709, 4710, 4721, 47
77, 4778, 4981, 5146, 5147, 5154, 5323, 5324, 5341, 5392, 5558, 5572, 5759, 5787, 5819, 5820, 5830, 5916, 6063,
6064]
Community 17: Size 109, Nodes: [646, 774, 838, 854, 936, 937, 940, 976, 1029, 1047, 1239, 1298, 1364, 1420, 1710
, 1837, 2075, 2099, 2252, 2253, 2254, 2284, 2377, 2380, 2436, 2519, 2532, 2552, 2690, 2691, 2968, 3064, 3106, 31
30, 3131, 3132, 3133, 3136, 3137, 3190, 3291, 3336, 3425, 3429, 3431, 3433, 3434, 3435, 3461, 3705, 4019, 4037,
4104, 4121, 4195, 4196, 4197, 4198, 4199, 4200, 4201, 4202, 4203, 4204, 4205, 4252, 4354, 4779, 4797, 4811, 4851
, 4947, 4951, 5194, 5257, 5407, 5436, 5437, 5438, 5454, 5455, 5463, 5586, 5587, 5601, 5624, 5639, 5679, 5732, 58
17, 5874, 5880, 5884, 5905, 5949, 5950, 5951, 6103, 6136, 6137, 6138, 6195, 6196, 6197, 6198, 6230, 6231, 6232,
6284]
Community 18: Size 101, Nodes: [466, 757, 759, 817, 1007, 1052, 1125, 1333, 1341, 1449, 1551, 1588, 1598, 1630,
1648, 1688, 1895, 1934, 2023, 2141, 2150, 2289, 2316, 2412, 2831, 2913, 3050, 3226, 3285, 3394, 3421, 3445, 3517
, 3748, 3836, 3837, 3998, 4062, 4160, 4322, 4330, 4338, 4370, 4462, 4570, 4578, 4658, 4659, 4660, 4661, 4776, 48
52, 4973, 4974, 4975, 5007, 5105, 5223, 5287, 5288, 5289, 5352, 5353, 5354, 5355, 5356, 5357, 5358, 5361, 5362,
5383, 5465, 5493, 5531, 5552, 5653, 5672, 5673, 5674, 5689, 5690, 5783, 5797, 5798, 5799, 5828, 5829, 6024, 6025
, 6036, 6081, 6134, 6135, 6173, 6192, 6193, 6194, 6226, 6277, 6281, 6282]
Community 19: Size 85, Nodes: [258, 360, 748, 761, 769, 1103, 1802, 2069, 2070, 2072, 2073, 2074, 2102, 2227, 22
58, 2261, 2262, 2364, 2378, 2401, 2422, 3168, 3169, 3170, 3171, 3172, 3179, 3180, 3181, 3182, 3183, 3184, 3185,
3186, 3187, 3188, 3238, 3478, 4002, 4015, 4016, 4018, 4031, 4597, 4642, 4684, 4760, 4761, 4762, 4763, 4824, 4825
, 4827, 4828, 4829, 4830, 4936, 4937, 4938, 4939, 4940, 4941, 4942, 4943, 5063, 5064, 5195, 5196, 5197, 5198, 51
99, 5315, 5316, 5317, 5638, 6011, 6012, 6013, 6014, 6015, 6016, 6017, 6018, 6019, 6020]
Community 20: Size 75, Nodes: [144, 184, 211, 600, 673, 832, 1036, 1092, 1302, 1771, 1903, 2156, 2569, 2581, 258
2, 2583, 2584, 2661, 2672, 2712, 2770, 3160, 3256, 3257, 3258, 3436, 3438, 3439, 3440, 3477, 3564, 3678, 3679, 3
681, 3879, 3898, 3910, 3911, 4060, 4061, 4072, 4073, 4077, 4078, 4079, 4080, 4114, 4365, 4436, 4524, 4598, 4599,
5261, 5262, 5263, 5277, 5278, 5468, 5469, 5470, 5471, 5472, 5473, 5474, 5538, 5541, 5542, 5543, 5581, 5582, 5631
, 5632, 5661, 5662, 5663]
Community 21: Size 2, Nodes: [5342, 5343]
```

Overlapping Algorithm

Clique Percolation

In [8]:
```
import networkx as nx
```

```
# Function to read a subset of the edges from the file
def read_subset_of_edges(file_path, num_edges):
    edges = []
    with open(file_path, 'r') as f:
        for _ in range(num_edges):
            line = f.readline().strip()
            if not line:
                break
            if line.startswith('#'):
                continue
            edges.append(tuple(map(int, line.split())))
    return edges

# Path to the edge list file
file_path = 'p2p-Gnutella08.txt'

# Number of edges to read
num_edges = 2000  # Adjust this number to control the subset size

# Read a subset of the edges
edges = read_subset_of_edges(file_path, num_edges)

# Create a graph from the subset of edges
graph = nx.Graph()
graph.add_edges_from(edges)

# Apply Clique Percolation algorithm with k-clique size, k
k = 4
communities = list(nx.algorithms.community.k_clique_communities(graph, k))

# Convert communities to list of lists for better readability
communities = [list(c) for c in communities]

# Display the number of communities found
print(f"Number of communities found: {len(communities)}")

# Display the first 100 communities
for i, community in enumerate(communities[:100], 1):  # Adjust the number of communities to display
    print(f"Community {i}: {community}")
```

```
Number of communities found: 8
Community 1: [353, 7, 367, 145, 177, 123]
Community 2: [249, 174, 5, 30]
Community 3: [123, 421, 390, 423]
Community 4: [145, 123, 124, 390]
Community 5: [123, 367, 15, 143]
Community 6: [176, 148, 124, 248]
Community 7: [423, 401, 427, 127]
Community 8: [264, 266, 366, 367, 177, 147]
```

Label Propagation Paper Link: Finding overlapping communities in networks by label

In [13]:

```
import networkx as nx
import random

# Function to read a subset of the edges from the file
def read_subset_of_edges(file_path, num_edges):
    edges = []
    with open(file_path, 'r') as f:
        for line in f:
            if line.startswith('#'):
                continue  # Skip comment lines
            edges.append(tuple(map(int, line.split())))
            if len(edges) >= num_edges:
                break
    return edges

# Path to the edge list file
file_path = 'p2p-Gnutella08.txt'

# Number of edges to read
num_edges = 2000  # Adjust this number to control the subset size

# Read a subset of the edges
edges = read_subset_of_edges(file_path, num_edges)

# Create a graph from the subset of edges
graph = nx.Graph()
graph.add_edges_from(edges)
```

```python
# Implement the Label Propagation Algorithm
def label_propagation(graph):
    # Initialize each node with a unique label
    labels = {node: node for node in graph.nodes()}
    nodes = list(graph.nodes())
    random.shuffle(nodes)  # Randomize the order of nodes

    while True:
        updated = False
        # For each node, update its label based on the most frequent label of its neighbors
        for node in nodes:
            if graph.degree(node) == 0:
                continue  # Skip isolated nodes
            neighbor_labels = [labels[neighbor] for neighbor in graph.neighbors(node)]
            most_frequent_label = max(set(neighbor_labels), key=neighbor_labels.count)
            if labels[node] != most_frequent_label:
                labels[node] = most_frequent_label
                updated = True

        # If no labels were updated, the algorithm has converged
        if not updated:
            break

    # Group nodes by labels
    communities = {}
    for node, label in labels.items():
        if label not in communities:
            communities[label] = []
        communities[label].append(node)

    return list(communities.values())

# Detect communities using Label Propagation
communities = label_propagation(graph)

# Display the number of communities found
print(f"Number of communities found: {len(communities)}")

# Display the first few communities
for i, community in enumerate(communities[:10], 1):  # Adjust the number of communities to display
    print(f"Community {i}: {community}")
```

```
Number of communities found: 130
Community 1: [0, 1, 2, 6, 10, 248, 914, 915, 916, 917, 918, 919, 920, 921, 922]
Community 2: [3, 703, 826, 1097, 1287, 1591, 1895, 1896, 1897, 1898, 1899]
Community 3: [4, 5, 7, 9, 144, 258, 491, 1021, 1418, 1669, 1900, 1901, 1902, 1903, 121, 127, 128, 179, 249, 264,
353, 424, 426, 145, 176, 177, 753, 754, 762, 2064, 3002, 124, 147, 246, 251, 15, 20, 123, 143, 367, 427, 718, 36
8, 717, 856, 975, 1908, 1909, 1910, 30, 960, 1911, 126, 174, 36, 101, 83, 148, 351, 946, 2000, 2001, 369, 422, 5
59, 2018, 113, 1787, 1317, 120, 122, 578, 658, 1428, 1451, 2057, 2058, 2059, 2060, 2061, 2062, 146, 390, 423, 20
63, 132, 696, 698, 755, 947, 2077, 1246, 2076, 130, 134, 137, 139, 140, 629, 1728, 141, 142, 149, 150, 2098, 265
, 667, 285, 3337, 5048, 5171, 5338, 5940, 5941, 5942, 1245, 238, 155, 167, 172, 266, 173, 697, 700, 2122, 192, 1
95, 558, 736, 1412, 314, 2218, 2219, 2181, 2182, 2183, 2184, 2185, 2193, 983, 513, 586, 873, 2206, 2207, 2208, 2
209, 2210, 263, 268, 269, 2224, 2225, 2226, 2227, 2228, 279, 281, 282, 286, 287, 311, 712, 955, 1197, 1227, 2243
, 298, 352, 666, 378, 3677, 2278, 339, 364, 1755, 340, 342, 343, 344, 345, 347, 881, 354, 357, 358, 366, 371, 37
7, 2338, 391, 393, 394, 395, 396, 397, 401, 409, 421, 2394, 2395, 2396, 2397, 2398, 2399, 2400, 2401, 2402, 2403
, 2404, 2405, 2406, 2407, 2408, 940, 2409, 2410, 2411, 2412, 2413, 2414, 2415, 425, 695]
Community 4: [8, 520, 665, 852, 1394, 1786, 1842, 1904, 1905, 1906, 1907]
Community 5: [247, 776, 900, 1068, 1069, 1070, 1071, 1072, 1073, 1074]
Community 6: [250, 51, 114, 1726, 1349, 1389, 1644, 4180, 4181]
Community 7: [252, 808, 1475, 2211, 2212, 2213, 2215, 2216]
Community 8: [11, 12, 14, 18, 19, 809]
Community 9: [13, 2074, 152, 2053, 2056, 151, 153, 154, 156, 157, 159, 160, 161, 2109, 2110, 2111, 2112, 2113, 2
114, 2115]
Community 10: [16, 66, 70, 71, 72, 73, 74, 1969, 1970, 1971, 1972, 1973, 1974, 1975]
```

Propagation

In [12]:

```python
import networkx as nx
import random

# Function to read a subset of the edges from the file
def read_subset_of_edges(file_path, num_edges):
    edges = []
    with open(file_path, 'r') as f:
        for _ in range(num_edges):
            line = f.readline().strip()
            if not line or line.startswith('#'):
                continue  # Skip empty lines and comments
            try:
                edges.append(tuple(map(int, line.split())))
            except ValueError:
```

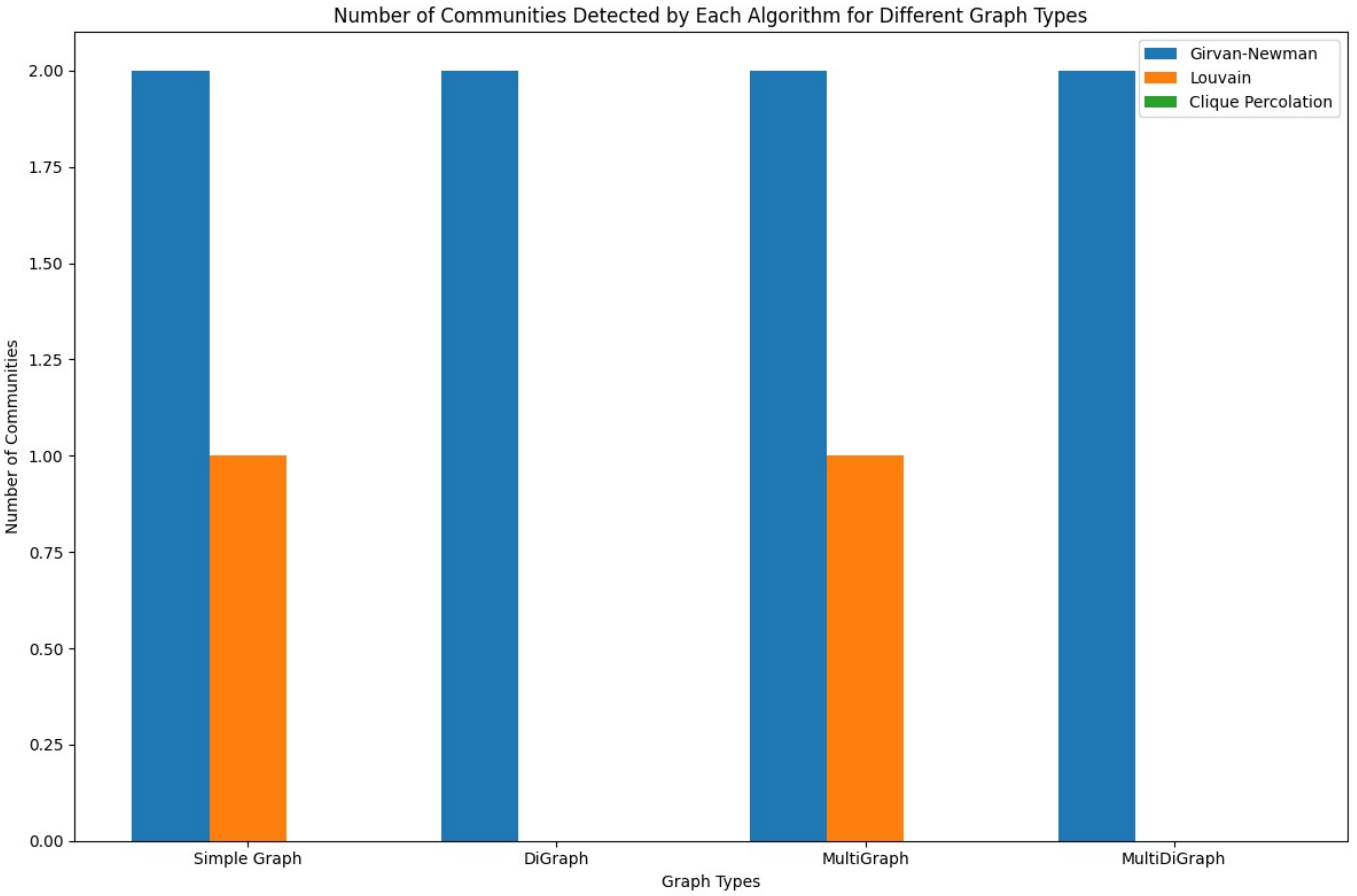
```
                print(f"Skipping invalid line: {line}")
    return edges

# Path to the edge list file
file_path = 'p2p-Gnutella08.txt'

# Number of edges to read
num_edges = 2000  # Adjust this number to control the subset size

# Read a subset of the edges
edges = read_subset_of_edges(file_path, num_edges)

# Create a graph from the subset of edges
graph = nx.Graph()
graph.add_edges_from(edges)

# Label Propagation Algorithm implementation
def label_propagation(graph):
    # Initialize each node with a unique label
    labels = {node: node for node in graph.nodes()}
    nodes = list(graph.nodes())
    random.shuffle(nodes)  # Shuffle nodes to ensure randomness

    while True:
        updated = False
        # For each node, update its label based on the most frequent label of its neighbors
        for node in nodes:
            if graph.degree(node) == 0:
                continue  # Skip isolated nodes
            neighbor_labels = [labels[neighbor] for neighbor in graph.neighbors(node)]
            most_frequent_label = max(set(neighbor_labels), key=neighbor_labels.count)
            if labels[node] != most_frequent_label:
                labels[node] = most_frequent_label
                updated = True

        # If no labels were updated, the algorithm has converged
        if not updated:
            break

    # Group nodes by labels to form communities
    communities = {}
    for node, label in labels.items():
        if label not in communities:
            communities[label] = []
        communities[label].append(node)

    return list(communities.values())

# Detect communities using Label Propagation
communities = label_propagation(graph)

# Display the number of communities found
print(f"Number of communities found: {len(communities)}")

# Display the first few communities
for i, community in enumerate(communities[:10], 1):  # Adjust the number of communities to display
    print(f"Community {i}: {community}")
```

```
Number of communities found: 126
Community 1: [0, 1, 2, 6, 10]
Community 2: [3, 826, 1097, 1287, 1895, 1896, 1897, 1898, 1899]
Community 3: [4, 5, 7, 9, 144, 258, 491, 1021, 1418, 1669, 1900, 1901, 1902, 1903, 121, 127, 128, 179, 249, 264,
353, 424, 426, 145, 176, 177, 753, 754, 762, 2064, 3002, 124, 147, 246, 251, 15, 20, 123, 143, 367, 427, 718, 36
8, 717, 856, 975, 1908, 1909, 1910, 30, 960, 1911, 126, 174, 33, 34, 35, 36, 37, 40, 41, 42, 101, 152, 83, 148,
351, 946, 2000, 2001, 369, 422, 559, 2018, 113, 1787, 1317, 120, 122, 578, 658, 1428, 1451, 2057, 2058, 2059, 20
60, 2061, 2062, 146, 390, 423, 2063, 132, 696, 698, 755, 947, 2077, 1246, 2076, 629, 141, 142, 149, 150, 2098, 2
65, 667, 3337, 5048, 5171, 5338, 5940, 5941, 5942, 1245, 238, 151, 153, 154, 155, 156, 157, 158, 159, 161, 813,
1688, 2104, 2105, 2106, 2107, 2108, 167, 172, 266, 173, 697, 700, 2122, 195, 558, 736, 1412, 314, 2181, 2182, 21
83, 2184, 2185, 2193, 983, 513, 586, 873, 2206, 2207, 2208, 2209, 2210, 261, 263, 268, 269, 2224, 2225, 2226, 22
27, 2228, 311, 298, 352, 666, 378, 3677, 2278, 339, 364, 1755, 343, 345, 354, 357, 358, 366, 371, 377, 2338, 391
, 393, 394, 395, 396, 397, 401, 409, 421, 2394, 2395, 2396, 2397, 2398, 2399, 2400, 2401, 2402, 2403, 2404, 2405
, 2406, 2407, 2408, 940, 2409, 2410, 2411, 2412, 2413, 2414, 2415, 425, 695]
Community 4: [8, 520, 665, 852, 1394, 1786, 1842, 1904, 1905, 1906, 1907]
Community 5: [703, 1561, 162, 163, 164, 165, 166, 168, 170, 171, 531, 1165, 1390, 1865, 2123, 2124]
Community 6: [1591, 2177, 231, 1891, 303, 309, 341, 764, 909, 1061, 1093, 1095, 2253, 2255, 2319, 370, 862, 1805
, 2288, 2289, 2290, 2291, 372, 373, 374, 375, 379, 2315, 2316, 2317, 2318, 2320, 2373]
Community 7: [247, 776, 900, 1068, 1069, 1070, 1071, 1072, 1073, 1074]
Community 8: [248, 914, 915, 916, 917, 918, 919, 920, 921, 922]
Community 9: [250, 1923, 1928, 51, 55, 56, 57, 58, 59, 64, 1950, 91, 102, 106, 1946, 1947, 1948, 1949, 1951, 195
2, 1968, 2194, 98, 99, 103, 104, 107, 2026, 2027, 2028, 2025, 114, 1726, 1349, 1389, 1644, 4180, 4181]
Community 10: [252, 808, 1475, 2211, 2212, 2213, 2214, 2215, 2216]
```

Bar Chart

In [4]:
```
import matplotlib.pyplot as plt
import networkx as nx
import community  # Louvain algorithm for community detection
from networkx.algorithms.community import girvan_newman, k_clique_communities
from itertools import islice

# Read the dataset and create a graph
graph = nx.read_edgelist("p2p-Gnutella08.txt")

# Limit the graph to a subgraph with fewer nodes for faster processing
graph = graph.subgraph(list(graph.nodes)[:10])

# Define the graph types
graph_types = ['Simple Graph', 'DiGraph', 'MultiGraph', 'MultiDiGraph']

# Initialize lists to store the number of communities detected by each algorithm for different graph types
girvan_newman_communities = []
louvain_communities = []
clique_communities = []

# Function to apply Girvan-Newman algorithm
def apply_girvan_newman(graph):
    comp = girvan_newman(graph)
    limited_levels = list(islice(comp, 1))  # Limit to the first iteration
    return len(limited_levels[0])

# Function to apply Louvain algorithm
def apply_louvain(graph):
    partition = community.best_partition(graph)
    return len(set(partition.values()))

# Function to apply Clique Percolation algorithm
def apply_clique_percolation(graph, k=3):
    cliques = list(k_clique_communities(graph, k))
    return len(cliques)

# Apply Girvan-Newman algorithm to detect communities
for graph_type in graph_types:
    if graph_type == 'Simple Graph':
        g_newman = nx.Graph(graph)
    elif graph_type == 'DiGraph':
        g_newman = nx.DiGraph(graph)
    elif graph_type == 'MultiGraph':
        g_newman = nx.MultiGraph(graph)
    elif graph_type == 'MultiDiGraph':
        g_newman = nx.MultiDiGraph(graph)

    communities_count = apply_girvan_newman(g_newman)
    girvan_newman_communities.append(communities_count)

# Apply Louvain algorithm to detect communities (only for undirected graph types)
for graph_type in graph_types:
    if graph_type == 'Simple Graph':
        g_louvain = nx.Graph(graph)
        communities_count = apply_louvain(g_louvain)
        louvain_communities.append(communities_count)
    elif graph_type == 'MultiGraph':
        g_louvain = nx.MultiGraph(graph)
        communities_count = apply_louvain(g_louvain)
        louvain_communities.append(communities_count)
    else:
        louvain_communities.append(None)  # Louvain algorithm is not applicable

# Apply Clique Percolation algorithm to detect communities (only for undirected graph types)
for graph_type in graph_types:
    if graph_type == 'Simple Graph':
        g_clique = nx.Graph(graph)
        communities_count = apply_clique_percolation(g_clique)
        clique_communities.append(communities_count)
    elif graph_type == 'MultiGraph':
        g_clique = nx.MultiGraph(graph)
        communities_count = apply_clique_percolation(g_clique)
        clique_communities.append(communities_count)
    else:
        clique_communities.append(None)  # Clique Percolation is not applicable

# Create a bar chart
fig, ax = plt.subplots(figsize=(12, 8))

bar_width = 0.25
```

```
index = range(len(graph_types))

# Plotting bars for each algorithm
bar1 = ax.bar(index, girvan_newman_communities, bar_width, label='Girvan-Newman')
bar2 = ax.bar([i + bar_width for i in index],
              [count if count is not None else 0 for count in louvain_communities],
              bar_width, label='Louvain')
bar3 = ax.bar([i + 2 * bar_width for i in index],
              [count if count is not None else 0 for count in clique_communities],
              bar_width, label='Clique Percolation')

# Adding labels and title
ax.set_xlabel('Graph Types')
ax.set_ylabel('Number of Communities')
ax.set_title('Number of Communities Detected by Each Algorithm for Different Graph Types')
ax.set_xticks([i + bar_width for i in index])
ax.set_xticklabels(graph_types)
ax.legend()

# Display the bar chart
plt.tight_layout()
plt.show()
```



The End 11508 Muzamil Khan