

1. Carregar o Grafo a partir do arquivo CSV.

In [3]:

```

import pandas as pd
import networkx as nx
import unidecode
import matplotlib.pyplot as plt
from networkx.algorithms import bipartite

base_autores = pd.read_csv("D:\\\\Users\\\\VitorAlmeida\\\\Programas_Unifor\\\\Grafos_Publicos\\\\autores_titulos.csv")
base_autores = base_autores[["AUTORES","TITULO"]]

def ajustar_texto(texto):
    #passo de ETL, pensar como melhorar e encontrar nomes parecidos
    texto_aj = unidecode.unidecode(texto)
    texto_aj = texto_aj.lower()
    return texto_aj

base_autores["AUTORES"] = base_autores["AUTORES"].apply(ajustar_texto)
base_autores["TITULO"] = base_autores["TITULO"].apply(ajustar_texto)
base_autores = base_autores.drop_duplicates() #retirar duplicatas

#transformar as colunas pandas para listas para colocar no grafo
lista_autores = base_autores['AUTORES'].tolist()
lista_titulos = base_autores['TITULO'].tolist()
lista_arestas = base_autores.values.tolist()

```

2. Representar os dados como um Grafo Bipartido entre os trabalhos publicados e os autores dos trabalhos.

In [17]:

```

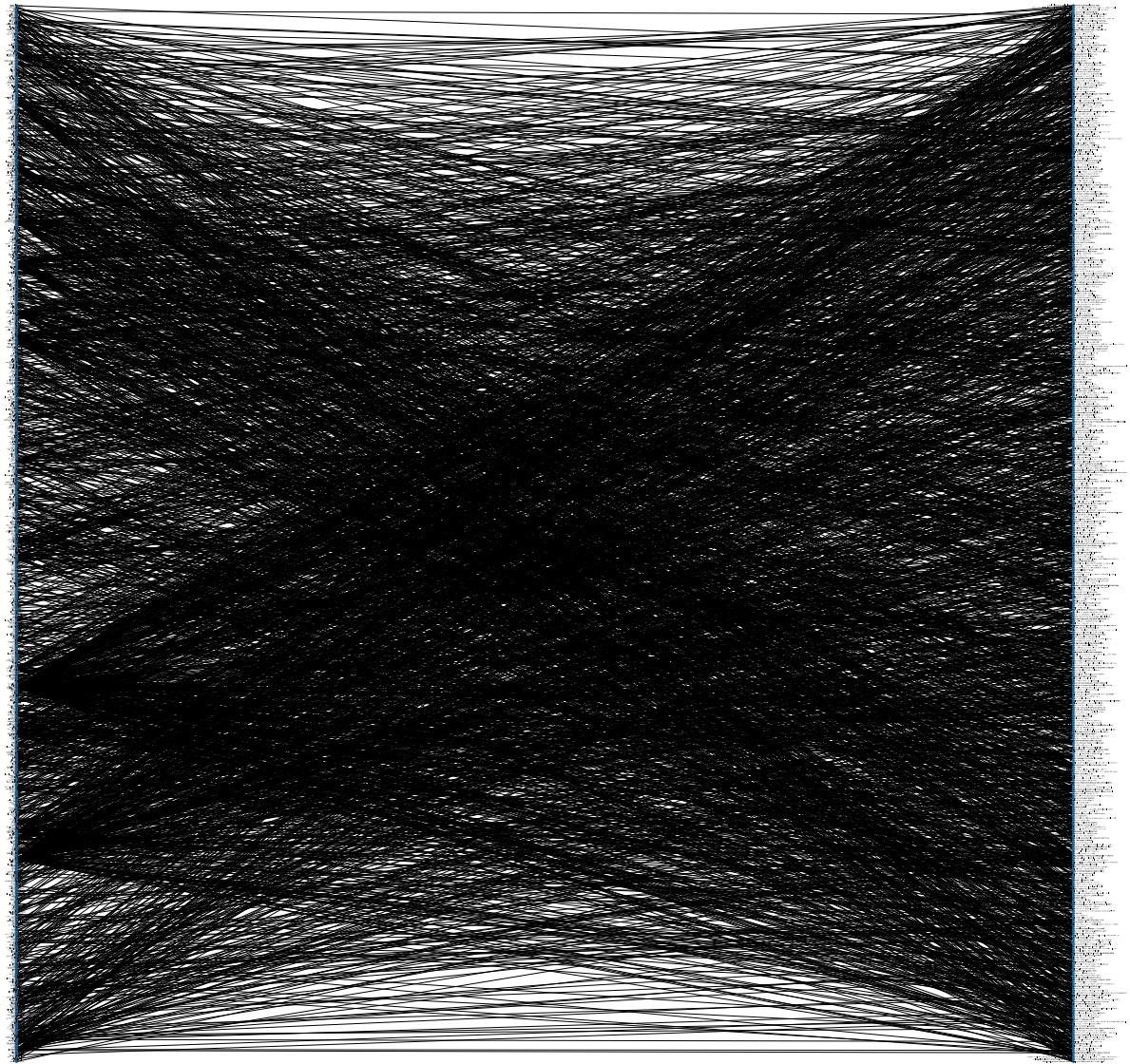
grafo_bi = nx.Graph()

grafo_bi.add_nodes_from(lista_autores, bipartite=0)
grafo_bi.add_nodes_from(lista_titulos, bipartite=1)
grafo_bi.add_edges_from(lista_arestas)

top = nx.bipartite.sets(grafo_bi)[0]
pos = nx.bipartite_layout(grafo_bi, top)

plt.figure(figsize=(150,150))
nx.draw(grafo_bi, pos=pos, with_labels=True, node_color=['green','green','green'],
        plt.savefig('grafo_bi_partido.png', format='png')
        plt.show()
        plt.clf()

```



<Figure size 432x288 with 0 Axes>

3. Gerar um Grafo Monopartido ponderado com apenas os autores.

In [19]:

```
plt.figure(figsize=(150,150))
grafo_mono_autores = bipartite.projected_graph(grafo_bi, top)
nx.draw(grafo_mono_autores, with_labels=True)
plt.savefig('grafo_so_autores.png', format='png')
plt.show()
plt.clf()
```



<Figure size 432x288 with 0 Axes>

4. Identificar os autores com maior colaboração com outros autores.

In [6]:

```
degX, degY = bipartite.degrees(grafo_mono_autores, lista_autores)
dict_grafo_autores = dict(degY)

top_20_autores = dict(sorted(dict_grafo_autores.items(), key=lambda x: x[1], reverse=True))
print(top_20_autores)
```

```
{'victor hugo costa de albuquerque': 390, 'placido rogerio pinheiro': 193, 'joel j p c rodrigues': 107, 'joao jose vasco peixoto furtado': 90, 'joao batista furlan duarte': 77, 'maria elizabeth sucupira furtado': 74, 'nabor das chagas mendonca': 69, 'raimundo holanda filho': 69, 'pedro p reboucas filho': 69, 'victor hugo c de albuquerque': 66, 'adriano bessa albuquerque': 58, 'andre luis vasconcelos coelho': 54, 'maria andreia formico rodrigues': 51, 'j m r s tavares': 48, 'pedro pedrosa reboucas filho': 48, 'khan muhammad': 46, 'marco tulio oliveira valente': 42, 'deepak gupta': 39, 'wanqing wu': 37, 'americo tadeu falcone sampaio': 37}
```

5. Identificar os autores com maior quantidade de artigos compartilhados.

In [7]:

```
G = nx.from_pandas_edgelist(base_autores, 'AUTORES', 'TITULO')
```

```
top_20_autores = dict(sorted(dict(G.degree()).items(), key=lambda x: x[1], reverse=True))
print(top_20_autores)
```

```
{'victor hugo costa de albuquerque': 157, 'placido rogerio pinheiro': 123, 'joao jos  
e vasco peixoto furtado': 41, 'maria elizabeth sucupira furtado': 38, 'maria andreia  
formico rodrigues': 36, 'joel j p c rodrigues': 34, 'the crosscutting impact of the  
aasd brazilian research community': 33, 'j m r s tavares': 30, 'andre luis vasconcel  
os coelho': 29, 'adriano bessa albuquerque': 28, 'joao batista furlan duarte': 25,  
'cloud computing and reservoir project': 23, 'raimir holanda filho': 22, 'pedro p re  
boucas filho': 22, 'nabor das chagas mendonca': 19, 'khan muhammad': 19, 'mirian cal  
iope dantas pinheiro': 16, 'victor hugo c de albuquerque': 16, 'pedro pedrosa rebouc  
as filho': 15, 'isabelle tamanini': 13}
```

6. Aplicar diferentes cortes e analisar.

In []:

7. Plotar os Grafos gerados.

In []:

8. Gerar um gráfico com a Frequência x Valor Link

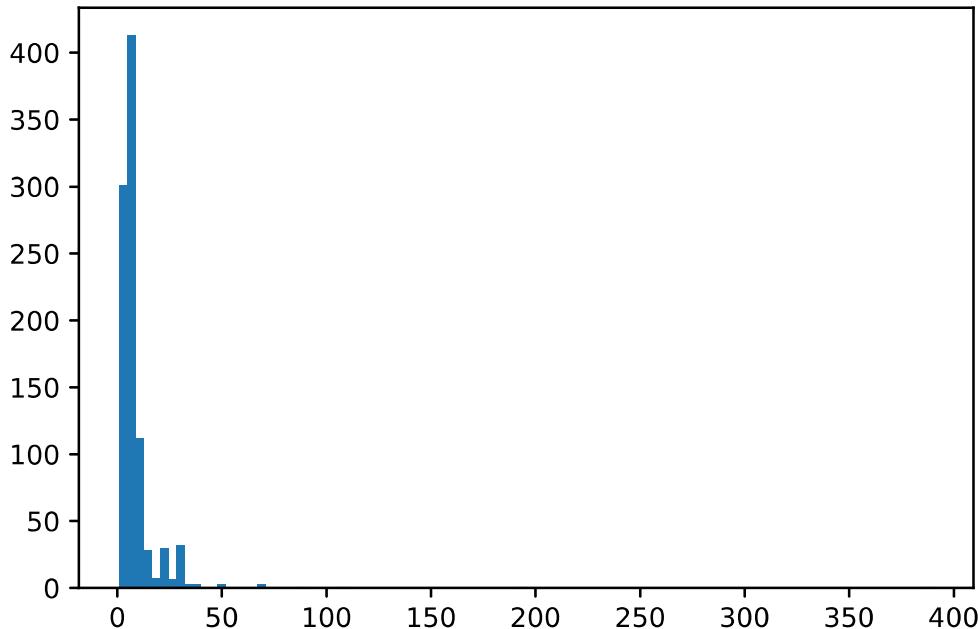
In [15]:

```
dict_grafo_autores_lista_valores = list(dict_grafo_autores.values())
```

```
plt.hist(dict_grafo_autores_lista_valores, bins=100)
```

```
#plt.figure(figsize=(20,5))
#plt.ylabel("Values")
#sns.boxplot().set_title('Distribuição de Graus e frequência')
#sns.countplot(x = distribuicao_graus)
```

Out[15]:



9. Calcular as seguintes propriedades da rede : Densidade, quantidade de nós isolados, diâmetro, distância média entre os nós, coeficiente de clustering.

In [9]:

```

densidade_autores = bipartite.density(grafo_mono_autores, lista_autores)
print("Densidade : ", densidade_autores)

#print(top_20_autores)
dic_autores_grau_zero = {}

for key, value in dict_grafo_autores.items():
    if (value == 1):
        dic_autores_grau_zero[key] = value

print("Nós Isolados: ", len(dic_autores_grau_zero))

diametro = nx.diameter(grafo_mono_autores)
print("Diâmetro : ", diametro)

```

```

Densidade : -0.0015134039680302447
Nós Isolados:  20
Diâmetro :  7

In [10]:
clus = nx.clustering(grafo_mono_autores)
print("Clusters : ", clus)

```

| Nó | Clustering |
|------------------------------------|---------------------|
| ssa albuquerque' | 0.102843315184513 |
| 'f o de franca' | 0 |
| 'raimir holanda' | 0.5054945 |
| 'giovanna sannino' | 0.5714285714285714 |
| 'juan carlos alvarado alcocer' | 0.36594202898550726 |
| 'pedro h vilela' | 1.0 |
| 'dorasilvia pontes lima' | 1.0 |
| 'igor silva' | 1.0 |
| 'victor hugo c de albuquerque' | 0.14871794871794872 |
| 'lakshmanaprabu sk' | 0.803030303030303 |
| 'sayed chhattan shah' | 1.0 |
| 'jose de aguiar moraes filho' | 1.0 |
| 'marli cardia' | 1.0 |
| 'paulo henrique monteiro borba' | 1.0 |
| 'sarah mesquita lima' | 1.0 |
| 'stephanie braga' | 1.0 |
| 'deepak gupta' | 0.21187584345479082 |
| 'francisco a de franco neto' | 1.0 |
| 'andre nunes de souza' | 1.0 |
| 'maikol magalhaes rodrigues' | 1.0 |
| 'raul victor m da nobrega' | 0.580952380952381 |
| 'hernan a makse' | 0.6944444444444444 |
| 's p p da silva' | 1.0 |
| "claudio sant\'anna" | 1.0 |
| 'raul v m da nobrega' | 1.0 |
| 'jose marques soares' | 1.0 |
| 'francisca marcia goncalves' | 1.0 |
| 'tayana conte' | 1.0 |
| 'alex ramos' | 0.7 |
| 'joao paolo c m oliveira' | 1.0 |
| 'glauber vasconcelos' | 1.0 |
| 'tanveer hussain' | 0.5128205128205128 |
| 'teresa yamana' | 1.0 |
| 'francisco j dos santos' | 0 |
| 'ibrahim el-henawy' | 1.0 |
| 'niels h kjeldsen' | 1.0 |
| 'roberto costa' | 1.0 |
| 'paulo masiero' | 1.0 |
| 'd j s goncalves' | 0.8055555555555556 |
| 'sudeep varshney' | 1.0 |
| 'andre luis carvalho coelho' | 1.0 |
| 'ronaldo parente de menezes' | 1.0 |
| 'victor hugo costa de albuquerque' | 1.0 |