## Обработка данных

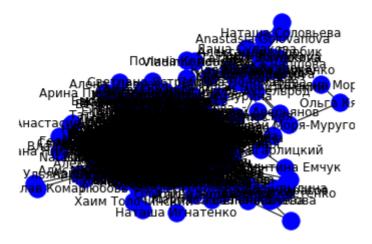
Чистила данные я прямо в Гефи, в Data Laboratory. Отсортировала Labels так, что стали видны сразу все Deleted, и я их удалила вручную. Там же я сделала красивые визуализации (лежат отдельными файлами в репозитории).

Дальше нужно было соединить таблицу узлов с таблицей ребер. Это получилось сделать в R.

```
library(tidyverse)
edges = read_csv('VK_Nika_edges.csv')
nodes = read_csv('VK_Nika_nodes.csv')
edges %>%
left_join(nodes, by = c("Source" = "Id")) %>%
left_join(nodes, by = c("Target" = "Id")) %>%
rename(label_source = "Label.y", label_target = "Label") %>%
select(Source, Target, Id, label_source, label_target) -> clean_graph
write.csv(clean_graph,'clean_graph_data.csv') # здесь записываем данные в новый сsv-файл
```

#### Переходим в Питон (Pycharm)

```
import networkx as nx
import collections
import matplotlib.pyplot as plt
import csv
import pandas as pd
from networkx import number of nodes
from networkx import number of edges
from networkx import density
from networkx import number connected components
from networkx import weakly connected components
from networkx import periphery
from networkx.algorithms.community.centrality import girvan newman
import community as louvain community
import networkx.algorithms.community as nx comm
from collections import Counter
data = pd.read csv("clean graph data.csv")
labels = dict(zip(data['Source'], data['label source']))
G=nx.from pandas edgelist(data, 'Source', 'Target')
pos=nx.spring layout(G)
nx.draw_networkx_labels(G, pos, labels)
nx.draw(G, pos, with labels=False, node color='blue')
plt.show()
```



Некрасиво. Все красивые визуализации я сделала в Гефи.

### ▼ Метрики

```
#Весь граф
print('Количество узлов:', number of nodes(G))
print('Количество peбep:', number_of_edges(G))
print('Плотность:', density(G))
print('Компоненты связности:', number connected components(G))
     Количество узлов: 511
     Количество ребер: 8048
     Плотность: 0.0617627873067035
     Компоненты связности: 1
print ('Диаметр: ', nx.diameter(G))
print ('Средняя длина пути: ', nx.average_shortest_path_length(G))
print ('Кластеризация: ', nx.transitivity(G))
print ('Кластерный коэффициент: ', nx.average_clustering(G))
print ('Количество узлов на периферии: ', len(nx.periphery(G)))
     Диаметр:
     Средняя длина пути: 2.53628026553087
     Кластеризация: 0.3581921893646605
     Кластерный коэффициент: 0.47380426736870446
     Количество узлов на периферии:
```

График распределения степеней тоже автоматически выгрузила из Гефи, лежит отдельно в папке. Смысл его в том, что у самые большие значения degree - всего у нескольких человек, а у больших групп людей degree ниже. (Самое большое значение - 223 - у одного человека, который и является самым крупным узлом на графе, если ранжировать величину узлов по degree).

# ▼ Поиск сообществ разными алгоритмами

```
#Girvan-Newman algorithm

cm = nx_comm.girvan_newman(G)

communities = next(cm)

print(len(communities))

2

#Louvain algorithm

partition = louvain_community.best_partition(G)

modularity = louvain_community.modularity(partition, G)

print('Количество сообществ:', max(Counter(partition).values()))

С> Количество сообществ: 6
```

### Интерпретация

Гефи выдал очень хороший результат разделения на сообщества (оставила базовый resolution 1.0), все совершенно так, как и есть в жизни. Самый отдельный кластер - первая школа, где я училась до 9 класса. Видно, что вторая школа дала мне огромное количество новых связей, плавно перетекая в университет. Выделились даже более маленькие сообщества разных "тусовок".