HW_4

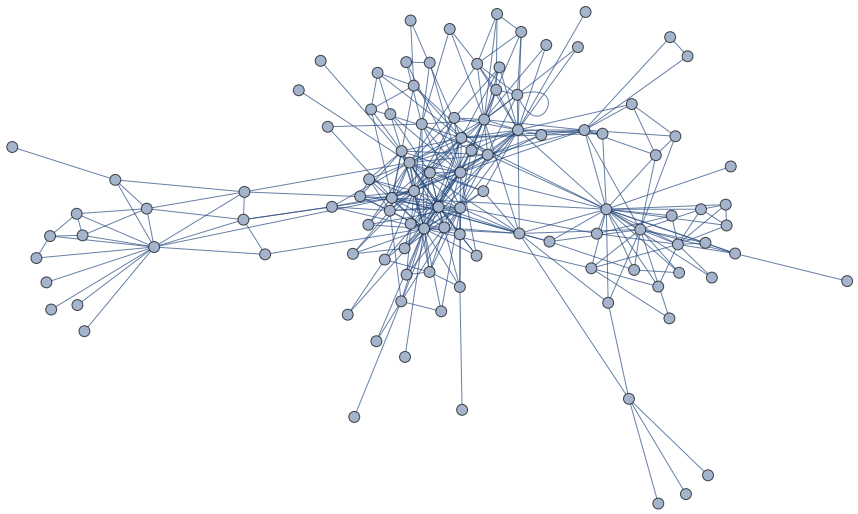
shahaf zohar – 205978000

Roey Salah – 206115438

In[927]:=

```
Needs["GraphUtilities`"];
SetDirectory[NotebookDirectory[]];
file = Rest[Import["stormofswords.csv"]];
tribes = Import["tribes.csv"];
nodes = Flatten[tribes[[All, 1]]];
edges = #[[1]] ⟷ #[[2]] & /@ file [[All, {1, 2}]];
T = Graph[nodes, edges]
communitiesWithout = FindGraphCommunities[T]
G = Graph[nodes, edges, EdgeWeight → weights];
communitiesWith = FindGraphCommunities[G];
```

Out[933]=



Out[934]=

```
{{Aerys, Amory, Balon, Bronn, Chataya, Doran, Elia, Ellaria, Gregor, Ilyn, Jaime, Joffrey,
  Kevan, Lancel, Loras, Mace, Margaery, Meryn, Myrcella, Oberyn, Olenna, Podrick,
  Pycelle, Qyburn, Renly, Sandor, Shae, Tommen, Tyrion, Tywin, Varys, Walton},
 {Aemon, Alliser, Bowen, Craster, Cressen, Dalla, Davos, Eddison, Gilly, Grenn,
  Hodor, Janos, Jojen, Jon, Karl, Mance, Meera, Melisandre, Orell, Qhorin,
  Rattleshirt, Salladhor, Samwell, Shireen, Stannis, Styr, Val, Ygritte},
 {Arya, Bran, Brienne, Brynden, Catelyn, Cersei, Eddard, Edmure, Hoster,
  Jeyne, Lothar, Luwin, Lysa, Marillion, Nan, Petyr, Ramsay, Rickard,
  Rickon, Robb, Robert Arryn, Roose, Roslin, Sansa, Theon, Walder},
 {Aegon, Barristan, Belwas, Daario, Daenerys, Drogo, Illyrio, Irri,
  Jon Arryn, Jorah, Kraznys, Missandei, Rakharo, Rhaegar,
  Robert, Viserys, Worm}, {Anguy, Beric, Gendry, Thoros}}
```


Printed by Wolfram Mathematica Student Edition

Without Weights:

In[937]:=
```
Print["Modularity without weights: ", N[GraphAssortativity[T, communitiesWithout]]]
```
```
Modularity without weights: 0.629439
```

With Weights

In[938]:=
```
Print["Modularity with weights: ", N[GraphAssortativity[G, communitiesWith]]]
```
```
Modularity with weights: 0.784158
```

Positive and Negative

In[939]:=
```
G1 = ExampleData[{"NetworkGraph", "ZacharyKarateClub"}];
G2 = ExampleData[{"NetworkGraph", "LesMiserables"}];
G3 = ExampleData[{"NetworkGraph", "AmericanCollegeFootball"}];
G4 = ExampleData[{"NetworkGraph", "DiscussionRecall"}];
G5 = ExampleData[{"NetworkGraph", "ProteinInteraction"}];
G6 = ExampleData[{"NetworkGraph", "EurovisionVotes"}];
```

In[945]:=
```
N[GraphAssortativity[G1, VertexList[G1] → VertexDegree[G1]]]
```
Out[945]=
```
-0.475613
```

In[946]:=
```
N[GraphAssortativity[G2, VertexList[G2] → VertexDegree[G2]]]
```
Out[946]=
```
-0.0242187
```

In[947]:=
```
N[GraphAssortativity[G3, VertexList[G3] → VertexDegree[G3]]]
```
Out[947]=
```
0.16398
```

In[948]:=
```
N[GraphAssortativity[G4, VertexList[G4] → VertexDegree[G4]]]
```
Out[948]=
```
0.0138798
```

In[949]:=
```
N[GraphAssortativity[G5, VertexList[G5] → VertexDegree[G5]]]
```
Out[949]=
```
-0.161545
```

Printed by Wolfram Mathematica Student Edition

In[950]:=
```
N[GraphAssortativity[G6, VertexList[G6] → VertexDegree[G6]]]
```
Out[950]=
```
0.0138142
```

Explain why do you think these networks have this phenomenon

The assortativity coefficient measures the degree to which nodes in a network tend to be connected to nodes with similar degrees . A positive assortativity coefficient indicates that nodes with high degree are more likely to be connected to other nodes with high degree, while nodes with low degree are more likely to be connected to other nodes with low degree . A negative assortativity coefficient indicates the opposite pattern, with nodes of high and low degree being more likely to be connected to nodes of low and high degree, respectively .

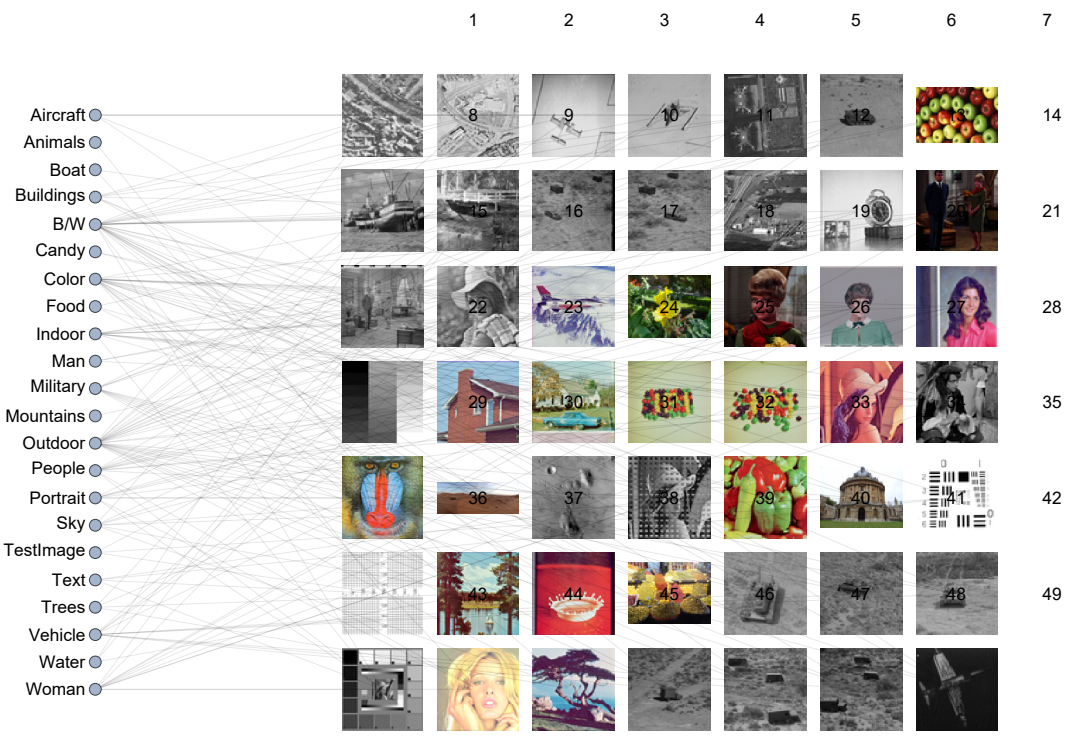
The assortativity coefficient measures the degree to which nodes in a network tend to be connected to nodes with similar degrees . A positive assortativity coefficient indicates that nodes with high degree are more likely to be connected to other nodes with high degree, while nodes with low degree are more likely to be connected to other nodes with low degree . A negative assortativity coefficient indicates the opposite pattern, with nodes of high and low degree being more likely to be connected to nodes of low and high degree, for example G1:

- G1 - (Zachary Karate Club): Assortativity coefficient of 0.475, indicating a tendency for high-degree nodes to be connected to other high-degree nodes and low-degree nodes to be connected to other low-degree nodes. This phenomenon may be due to the natural tendency for individuals to form social connections with others who are similar to themselves in terms of social status, influence, or popularity.

Find one more network with other scalar / enumerative mixing, positive or negative

In[951]:=
```
Gnew = ExampleData[{"NetworkGraph", "TaggedTestImages"}];
Graph[Gnew, VertexLabels → "Name"]
```

Out[952]=



In[953]:=
```
communitiesNew = FindGraphCommunities[Gnew]
```

Out[953]=
```
{{1, 2, 3, 4, 5, 8, 9, 12, 17, 23, 24, 30, 34, 37, 45, 49, Aircraft, Boat,
  Buildings, Mountains, Outdoor, Sky, Trees, Water}, {6, 10, 11, 22, 31, 32,
  35, 36, 40, 41, 42, 43, 46, 47, 48, B/W, Military, TestImage, Text, Vehicle},
 {13, 14, 15, 16, 19, 20, 21, 25, 26, 27, 28, 29, 44, Animals, Candy, Indoor,
  Man, People, Portrait, Woman}, {7, 18, 33, 38, 39, Color, Food}}
```

In[954]:=
```
PropertyList[Gnew]
```

Out[954]=
```
{GraphHighlight, GraphHighlightStyle, GraphLayout,
 GraphStyle, EdgeShapeFunction, EdgeStyle, VertexCoordinates,
 VertexShapeFunction, VertexShape, VertexSize, VertexStyle}
```

In[955]:=
```
bc = BetweennessCentrality[Gnew];
N[GraphAssortativity[Gnew, VertexList[Gnew] → bc]]
```

Out[956]=
```
-0.283495
```