Homework 0, Question 1

In [41]:

```
import networkx as nx
from collections import Counter
```

In [42]:

```
G = nx.read_edgelist("wiki-Vote.txt",create_using=nx.DiGraph())
```

a) Number of nodes in the network

In [43]:

```
len(G.nodes())
```

Out[43]:

```
7115
```

b) Number of nodes with a self edge

In [44]:

```
len(G.nodes_with_selfloops())
```

Out[44]:

```
0
```

c) Number of directed edges

In [45]:

```
len(G.edges())-len(G.nodes_with_selfloops())
```

Out[45]:

```
103689
```

Since there are no self-loops, the number of directed edges is the total number of edges

d) Number of undirected edges

Convert G to an undirected graph H and count the edges.

In [46]:

```
H = nx.Graph(G)
len(H.edges())
```

Out[46]:

```
100762
```

e) Number of reciprocated edges

```
In [47]: edge_cnt = Counter()
         for u,v in G.edges_iter():
             if u < v:
                 edge_cnt[(u,v)] += 1
             elif u > v:
                 edge_cnt[(v,u)] += 1

         recip = 0
         for e,cnt in edge_cnt.items():
             if cnt == 2: recip += 1

         recip

Out[47]: 2927
```

f) Number of nodes with zero out-degree

```
In [48]: len([n for n in G.nodes() if len(nx.DiGraph.successors(G,n)) == 0])

Out[48]: 1005
```

g) Number of nodes with zero in-degree

```
In [49]: len([n for n in G.nodes() if len(nx.DiGraph.predecessors(G,n)) == 0])

Out[49]: 4734
```

h) Number of nodes with out-degree > 10

```
In [50]: len([n for n in G.nodes() if len(nx.DiGraph.successors(G,n)) > 10])

Out[50]: 1612
```

h) Number of nodes with in-degree < 10

```
In [51]: len([n for n in G.nodes() if len(nx.DiGraph.predecessors(G,n)) < 0])

Out[51]: 0
```