Network representations, basic network algorithms

You are given three networks in Pajek format that was presented in lectures.

- A simple toy network for testing (tiny)
- Zachary karate club network (small)
- A part of Google web graph (large)

I. Adjacency list representation

1. **(code)** Assume that all networks are undirected. Implement your own adjacency list representation of the networks as an array of lists and represent all three networks.

2. **(code)** Now, assume that all networks are directed and extend your network representation accordingly.

3. **(answer)** Does your network representation allow for multiple links between the nodes, loops on nodes and isolated nodes?

II. Basic network statistics

1. **(code)** Compute basic statistics of all three networks. Namely, number of nodes n and links m, average node degree $\langle k\rangle = 2m/n$ and undirected density $\rho = m/\binom{n}{2}$. Are the results expected or are they surprising?

2. **(code)** Compute number of isolated nodes and number of pendant nodes (i.e. degree-1 nodes), and maximum node degree $k_{\max}$. How do the values of $k_{\max}$ compare to $\langle k\rangle$?

3. **(answer)** What is the time complexity of the computations above?

III. Network connected components

1. **(answer)** Study the following algorithm for computing (weakly) connected components by any order link traversal. Does the algorithm implement breadth-first or depth-first search? Why? What is the time complexity of the algorithm?

```
input   graph G, nodes N
output  network components {C}
    {C} ← empty list
    while not N empty do
        {C}.add(component(G, N, N.next()))
    return {C}
```

```
input   graph G, nodes N, node i
output  weak component C
    C ← empty list
    S ← empty stack
    N.remove(S.push(i))
    while not S empty do
        C.add(i ← S.pop())
        for neighbors j ∈ Γ_i do
            if N.remove(j) then
                S.push(j)
    return C
```

2. **(code)** Implement the algorithm and compute number of (weakly) connected components and size of the largest (weakly) connected component of all three networks. Are the results expected or are they surprising?