Social graphs section:

Function: make_empty_graph

1. Inputs: nothing
2. Output: graph
3. This function simply returns an empty graph.

Function: is_disjoint

4. Inputs: two sets of nodes
5. Output: true or false
6. This function goes through every node in the first set and checks it against every node in the second set.
7. If there are not matches, the function returns True.
8. If there is a match at any point, the function returns False.

Function: not_empty

9. Inputs: set of nodes
10. Output: true or false
11. This function simply returns whether or not a set is empty.
12. This is accomplished by checking whether the number of elements in the set is greater than 0 or not.

Function: is_equivalent

13. Inputs: graph, two sets of nodes
14. Output: true or false
15. This function compares the nodes in a graph to two given sets.
16. The two given sets are unioned to form a new set.
17. The set of nodes is obtained from the graph.
18. If the graph's set is equal to the unioned set, True is returned.
19. Otherwise, False is returned.

Function: is_partition

20. Inputs: graph, two sets of nodes
21. Output: true or false
22. This function checks whether the given graph is partitioned by the given nodes.
23. It calls the three functions above to check if disjoint, not empty, and equivalent.
24. If any return False, False is returned.
25. Otherwise, for every node in the first set, for every node in the second set, a previous function is called to check if the nodes are neighbors.
26. If they are, False is returned.
27. Otherwise, True.

Function: connect_all

28. Inputs: graph, set of nodes
29. Output: graph
30. This function completes a graph by adding edges.
31. For a node1 in the set of nodes, if the node is not in the graph's set of nodes, it is added.
    a. For a node2 in the set of nodes, if the node is not equal to node1 and the two nodes are not neighbors, then an edge is added between them.

32. The graph is returned.

Function: create_graph_and_partition

33. Inputs: set of nodes
34. Output: graph and 2 sets of nodes
35. An empty graph is made and two empty lists to contain nodes are initialized.
36. The input set is stored as a list.
37. For every node in the list of nodes, the node is added to the graph.
38. Modulo 2 is checked.
    a. If True, the node is added to a list.
    b. If False, the node is added to a different list.
39. Connect_all is called to connect the nodes and the graph and two lists, converted into sets, are returned.

Function: create_graph_without_partition

40. Inputs: set of nodes
41. Output: graph
42. An empty graph is made and two empty lists to contain nodes are initialized.
43. For every node in the set of nodes, the node is added to the graph.
44. Connect_all is called and the graph is returned.