Problem 1:

Approach
Use BFS. Since each edge is unweighted, the BFS will find the shortest path, because it checks nearest neighbors first, then expands to the next nearest neighbors until it finds the node it's searching for.

What is the order of complexity of this function?
O(V+E)

Problem 2:

Approach
Approach is just to add n-1 number of node between the source and destination to represent edge of weight n, since each edge is equally weighted. Then run BFS as in problem 1 to find the shortest path between the two edges, and that represents the weighted degree of separation.

What are the potential weaknesses of this approach?
The potential weaknesses is you cannot add an edge with a weight of 0 or a double.

If the staff had decided to rate each friendship as an integer from 1 to N, where N is an integer greater than 1, what would the order of complexity of this function be?
This function inserts at most N dummy edges and at most N-1 extra nodes for each edge, so at the end the graph will have at most NE dummy edges and (N-1)E dummy vertices. So complexity would be O (number_of_nodes + number_of_edges) = O((N-1)E+V + NE+E) = O(NE+V).

Problem 3:

Approach
Start with any node in the graph, and use BFS or DFS and modify it to mark each node reachable found as a group. Now see if there then go through the loop again picking a new (unmarked) start and mark everything reachable, and so on until no more nodes are left

What is the size of the largest group? How many students are not Facebook friends with anyone in 6.00? Are the 6.00 students well-connected on Facebook?
Size of largest group is 94.
14 students are not Facebook friends with anyone in 6.00
Overall it seems that 6.00 students are quite well connected with around 80 % of students in one facebook network.