1a) The modified QuickSelect algorithm can be efficiently used and assures that we choose the best students while severing ties at random.

1) First, preprocess the student data into a list of tuples, where each tuple contains the student ID and their CS301 overall numeric grade.

2) Apply the QuickSelect algorithm to the list, but modify it to find the m-th highest grade instead of the m-th lowest element. To do this, reverse the comparison in the partition step.

3) Once the QuickSelect algorithm finds the m-th highest grade, partition the list again using the grade as the pivot. With this, the list will be divided into three sections:

a. Students with grades higher than the m-th grade.

b. Students with the m-th grade.

c. Students with grades lower than the m-th grade.

4) The students in the first partition (a) will be sent to the visit. If there are still spots available (i.e., m is larger than the number of students in partition (a)), randomly select the remaining students from the second partition (b) until all spots are filled.

1b) Preprocessing the student data: O(n)

QuickSelect algorithm: Average case of O(n)

Partitioning the list again and selecting students: O(n)

So, in the average case, the run time complexity of the algorithm is O(n) (preprocessing) + O(n) (QuickSelect) + O(n) (partitioning and selecting) = O(3n). We can simplify this to O(n). Hence, the tight upper bound for the algorithm is O(n).

In the worst case, we have a time complexity of O(n^2), but this can be avoided by using a randomized pivot selection strategy.

2) Given your friends F = {f1, f2, ..., fn} and for each pair of your friends fi, fj ∈ F, whether fi and fj know each other or not, and an integer k (1 ≤ k ≤ n), is it possible to send k different messages to your friends such that no two friends who know each other receive the same message?

3a) True, at most 2 rotations are performed. The number of rotations is limited to a constant number.

3b) False, A red-black tree insertion can require Θ(log n) node recoloring in the worst case. An example of a balanced binary search tree that uses rules to keep its balance is a red-black tree.

Certain qualities may be violated during insertion, in which case recoloring and rotations may be required to restore the properties. In the worst situation, it's possible that we'll need to carry out recoloring procedures that spread from the tip of the tree to the root. The worst-case scenario for node recoloring can also be Θ(log n), as the height of a red-black tree is Θ(log n).

3c) False, a pre-order tree walk takes Θ(n). Any binary tree, including red-black trees, can be pre-order traversed by visiting each node a single time, with the time complexity being proportional to the number of nodes in the tree. Pre-order traversal has a temporal complexity of Θ(n) because the tree has n nodes. A red-black tree has a height of O(log n), but as each node must still be visited once, the traversal difficulty is unaffected by this.

4a) "NONDETERMINISTICALLY POLYNOMIAL" is what NP stands for. It is a complexity class in computational complexity theory, used to classify decision problems.

4b) If it is feasible to check whether a proposed solution g is valid or incorrect for the problem P in polynomial time $O(n^k)$ by a deterministic Turing machine, the term "NP" is used to describe the problem P.