Homework 8

Tianhe Lei

1. a) False, greedy algorithm is not optimal in Integer Knapsack Problem and the time complexity is Θ(nC) in Dynamic Programming.

   b) True, class P is a subset of class NP.

   c) False, NP-complete problem is NP problem, but not every P problem.

   d) True, Knapsack Decision Problem is NP-Complete and every NP problem can be polynomially reduced to it.

   e) True, when C < 1000, Θ(nC) becomes Θ(n) which could be solved in polynomial time.

   f) False, NP problem shall be solved in polynomial time.

   g) False, they cannot be solved in polynomial time, they are not NP problems.

   h) True, TSP is NP-complete, and NP-complete problem is NP-hard problem.

2. a) First, do BFS on unvisited student to get the number of connected graph C, mark the level of travelling, this is O(|V|). If k is less than C, the answer is yes.

   Else, for each connected graph, count the number of even-level student and odd-level student, add the greater one to the current event size. Event size start from C. When the event size is greater than k, return yes.

   If event size is still less than k when every connected graph is fully visited, return no. This is O(|V|).

   Overall, this could be done in polynomial time, it is a NP problem.

   b) So from (a), there is just one connected graph, and we just need to go through the line, calculate the number of odd index and even index students. If one of the number is greater than k, return yes. At the end of travel, return no. The algorithm visits each student once at most. This is O(|V|).

   c) Visit each student from the max height level, if the student is a leaf, max event size +=1, mark it counted in event.

If the student does not have a child student counted in event, count it in event, max event size +=1.

When the root node is visited, return the max event size.

d) For general undirected graphs, this problem could be reduced to SAT. Every student could be in the event or not, and if the students known to each other show up in the same event, result is no. Else, the result is yes.

For one possible combination, check the relationship from each connected graph, let students from that graph be 1, and do an add operation to student among this combination. If the result is greater than 1, the combination does not work, else, the combination works.

As a result, the problem is NP-complete.

3. a) It is NP-complete because it could be reduced to Subset Problem. The problem could be stated as: Is there a subset of packages S that the weight is less or equal to W. If answer is yes, then S is a subset with the whole kW. Each truck is independent. And repeat for k times to get the answer.

   b) Imagine a sequence of packages like w = 1, 5, 4 and truck size is 5.
   In greedy algorithm, it will be sent by 3 trucks.
   But in method from (a), it will be sent by 2 trucks.

4. a) weight is 7

   b) If d(I,j), d(i,k) and d(k,j) exist, since there is no cycle in this undirected graph, each of them must be 1 or 2.
   There are cases like 2=1+1, 1<1+2, 2<1+2, so d(i, j) ≤ d(i, k) + d(k, j).

   c) Euler Tour: 1-2-1-3-4-7-5-7-4-3-6-8-6-3-1

   d) after skip: 1-2-3-4-7-5-6-8-1
   length:1+2+1+1+1+2+1+2= 11

   e) not optimal, because it goes to the lowest numbered node, ignoring other information in the graph. May not use the 1-edge as much as possible.

   There are paths like 1-3-6-8-5-7-4-2-1, length 10.