Week 1 (Amortization and Fibonacci Heap):

1. Can a different definition of potential functions give different amortized cost for the same set of operations?
2. What is the importance of cascading cuts in DecreaseKey operation in Fibonacci Heap?
3. Given that children of min are already in a circular doubly linked list, is it possible to do Step 1 of ExtractMin (Take the min out and add its children to root list)in O(1) time (by splicing the list as a whole) instead of O(D(n))?
4. In DecreaseKey, why don't we mark the root node t the end (if the cascading cuts go all the way to root)?
5. In ExtractMin, can we start the "current" pointer anywhere other than min?
6. (General question) Can we say whether a time complexity found is the best for this problem?

Week 2 (Network Flow):

1. In the proof of Ford-Fulkerson, it is stated f(S,T) = f(S,V) – f(S,S). What does the cut (S,V) mean?
2. What is the logic behind considering minimum cut (a cut which has minimum capacity)? Shouldn't we consider a cut which has the maximum capacity while proving the Max-flow min-cut theorem?