1. Please describe any algorithm you know that use dynamic programming to solve all pairs shortest paths problem. And please describe the time complexity need for the algorithm you answer. (15 points)

2. We can implement the quick sort algorithm easily by recursive method, but the operation system needs to maintain one stack to store the recursive information. We can easily modify the quick sort code to decrease the total space needed in the stack by the following codes. (20 points)

   ```
   QuickSort2(A, p, r)   //A: the list, p: start position, r: end position
   1>  while p<r
   2>  do Partition and sort the left subarray
   3>      q = Partition(A, p, r) // q: pivot position after partition
   4>      QuickSort2(A, p, q-1)
   5>      p = q+1
   ```

   A. Argue that QuickSort2(A, 1, NumberOf(A) ) correctly sorts the array A (8 points).
   B. Describe a scenario in which the stack depth of QuickSort2 is O(n) on an n-element input array (5 points).
   C. Modify the code for QuickSort2 so that the worst-case stack depth is O(lgn) (7 points)

3. Given a set S of n integers and another integer x, determines whether or not there exist two elements in S whose sum is exactly x. Please describe a $\Theta$(n logn) time complexity algorithm to solve this problem. (15 points)

4. We know that finding a smallest vertex cover is an NP-complete problem. Can you show that, in a given graph G=(V,E), find out a maximum subset S of E such that any two nodes in S are not adjacent in G, is a NP-Complete problem. (20 points)

5. In the longest-common-subsequence problem, we are given two sequences X = <x1, x2, …, xm> and Y = <y1, y2, …, yn> and try to find a maximum-length common subsequence of X and Y. The LCS problem can be solved efficiently using dynamic programming. Please show the optimal substructure of an LCS and prove its correctness.    (20 points)

6. Show that if a node in a binary search tree has two children, then its successor has no left child and its predecessor has no right child. ( hint: Successor means the next node according to in-order traversal) (10 points)