Quiz 2 (100 scores + 20 bonus scores)

1. Minimum Number of Coins problem: If we want to make change for V cents, and we have infinite supply of each of {d[0], d[1], .. , d[m-1]} valued coins, what is the minimum number of coins to make the change?
   Please modify the following recursive implementation with exponential time complexity to a top-down dynamic programing with linear time complexity, by adding table[] to store the solutions of subproblems. (20 scores)

```
int minCoins(int V)
{
        if (V == 0)
                return 0;
        int res = INT_MAX;
        for (int i=0; i<m; i++)   // Try each type of coin. There are m types of coins.
        {
                if (d[i] <= V)
                {
                        int sub_res = minCoins(V-d[i]);
                        if (sub_res + 1 < res)
                                res = sub_res + 1;
                }
        }
        return res;
}
```

2. Minimum Number of Coins problem: Suppose you have an unlimited supply of three types of coins: d1 = 1 penny, d2 = 2 pennies, and d3 = 5 pennies, please fill out the given table using the bottom-up dynamic programming algorithm to find the fewest number of coins to make the change of 13. (10 scores)

   For the last two entries of the table, please explain how the algorithm computes the values? (10 scores)
   Please give the exact coins making the fewest number of the change by checking the table, and explain? (5 scores)

0	1	2	3	4	5	6	7	8	9	10	11	12	13

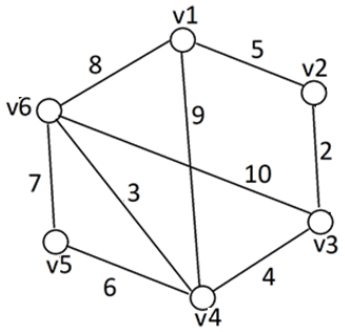
3. Knapsack problem: Assume that we have a knapsack with a max weight capacity W = 7. Our objective is to fill the knapsack with items such that the total value is maximum. The following table contains the items along with their value and weight. Note that no duplicate one for each item.

Item i	1	2	3	4
Value	10	6	11	2
Weight	3	2	4	1

   please fill out the given table using the bottom-up dynamic programming algorithm to find the maximum value of the knapsack. (10 scores)
   For the last entry of the table, please explain how the algorithm computes the value? (5 scores)
   Please give the exact items making the maximum value of the knapsack by checking the table, and explain? (10 scores)

[weight][item]	0	1	2	3	4	5	6	7
0								
1								
2								
3								
4								

4. Given graph G, use Prim's Algorithm and Kruskal's Algorithm to compute its Minimum Spanning Tree, respectivelt. For each algorithm, please write down the edge picked in each step. For example, Step 1: (v1, v2). (30 scores)



5. Bonus question: Given a distance D, please give a dynamic programming algorithm (either top-down or bottom-up) that counts the total number of ways to cover the distance with 1, 2, or 3 steps. (20 scores)

Example: Input: D = 3 Output: 4

- 1 step + 1 step + 1 step
- 1 step + 2 step
- 2 step + 1 step
- 3 step