3.1. Given a graph, each edge has a length. In one move, you can go through several edges, the sum of the lengths on which does not exceed D. Go from s to t in the minimum number of moves. $O(m \log n$.

3.2. How to find all vertices, the weight of the path to which can be arbitrarily small?

3.3. Modify Dijkstra's algorithm to run in $O(M + m)$ if all shortest paths are integers at most M.

3.4. Given a weighted graph. Remove the maximum number of edges, so that the distance from s to t is no more than d.

3.5. Suppose there are two regular games on the graph, but, unlike the usual sum of games, on each move the player makes a move in both games. If at least one token is at a dead end, then the player loses. Who will win if both players play optimally? Time $O(n + m)$.

3.6. Consider a regular game on a graph, but the first player wants the game to end (he doesn't care who wins), and the second wants it to go on forever. Determine who will win.

3.7. There is a strip of n cells. Players take turns making moves. The move consists in placing the domino on two adjacent free cells. If a player cannot place a domino, he loses. Who will win? Time $O(n^2)$.

3.8. There is a strip of n cells. Players take turns making moves. The move consists in placing the piece on an empty cell. In this case, you cannot put a piece on a cell if there is already a piece in the adjacent cell. Whoever cannot make a move loses. Who won? Time $O(n^2)$.

3.9. There is a strip of n cells. Players take turns making moves. The move consists in placing the piece on an empty cell. If after a player's move three pieces are in a row, then he wins. Who will win? Time $O(n^2)$.