ASSIGNMENT 2: Spanning tree finding on a weighted graph

Tasks:
1. Modify the implementations of graphs (both adjacency matrix and list) from assignment 1, so that weights can be added to the edges of the graph
2. Implement Prim's Algorithm to find the minimum spanning tree for a weighted graph
3. Implement Kruskal's Algorithm to find the minimum spanning tree a weighted graph

Tips:
1. Use rand() function from stdlib.h to generate random weights for your graphs and check outputs
2. Can you use any of the above algorithms (modify inputs cleverly) to find the maximum spanning from for a graph as well
3. You can use clock() function from time.h to measure how much time your program (or, part of your program) takes to compute.

Bonus:
a. You can run the code on 20 different sizes (increasing the number of node from 10 to 1000, or beyond) of graph and measure the time your code takes to run each code