Programming Assignment 2
Shortest Path

TODO Items

1. Implement the `dijkstra` function in hw2.cpp
   - Dijkstra can be viewed as a spfa variant using priority queue
   - You will need to study how to manipulate `std::priority_queue` or write you own priority queue (e.g., binary heap)
   - There are many resources online and you need to study them yourself
2. Implement the `spfa-queue` function in graph.cpp
   - This is essentially what you did in icp8
3. Implement the `spfa-stack` function in graph.cpp
   - Replace the queue in spfa with stack
4. All algorithms start from vertex 0 and find shortest paths to all other vertices
   - The distance vector returned from each function must match exactly
5. Run your code through all 10 graphs (graph[1-10].txt) and report the runtime (3x10 tables)
6. Email me your **graph.cpp** together with your **uid** and **name** by 3:30 PM 3/26 (before class)

Data Format

```
V E              # number of vertices and edges
u1 v1 w1     # E lines to follow; u→v with weight w
u2 v2 w2
u3 v3 w3
…
uE vE wE
```

```
# each vertex is indexed in the range [0, V)
# each weight is in the rage [0, 10]
```