Minimal Spanning Tree (Prim's Algorithm)

Terms and Definitions

Minimal Spanning Tree (MST): A set of edges in a connected, undirected, edge-weighted graph that connects all the vertices without any cycles and with the lowest possible edge length.

Algorithm Outline

Given $G = (V, E)$, we will keep track of which node has been visited by the following procedure, and start at an arbitrary vertex s:

```
while not all vertices has been visited:
    mark s as visited
    u = s's closest neighbor that has not been visited[1]
    add edge connecting s and u to the MST
    s = u
```

Algorithm Pre-Condition and Pitfalls

Note that this algorithm only works correctly for **undirected graphs**. A different algorithm must be used for directed graphs.
Also note that this program does not terminate if the graph is not connected. A way to fix this is to slightly modify the algorithm to generate a **Minimal Spanning Forest**, or a collection of MST's. The modification is that if all of s's neighbors have been visited, let s be a random vertex that has not been visited and continue the loop.

Performance

The algorithm is greedy in nature, and the runtime depends on implementation, the main difference is using a priority queue or linear search, and the runtime difference are:

Priority Queue	$O(\|E\| \log \|V\|)$
Linear Search	$O(\|V\|^2)$

Due to the cost of building a priority queue, it is generally preferred to just use a linear search for dense graphs (where $|E| \approx |V|^2$), but the priority queue implementation is generally preferred for sparse graphs.

Sample Problems

https://uva.onlinejudge.org/external/100/10034.pdf

[1] Use a priority queue or linear search (see Performance for more details)