hw4 problem 3

Brian Wang-Chen

March 2024

1 Graph Implementation

Ways to represent a Graph:

1. As a collection of edges
2. As an adjacency list, in which each node is associated with a list of its outgoing edges
3. As an adjacency matrix, which explicitly represents, for every pair ¡A,B¿of edges, whether there is a link from A to B, and how many

Describe Implementation for the 3 representations above and your representation. Explain the advantages of that representation, and briefly explain why you chose the representation

- The first implementation, collection of edges which would simply be a collection data structure like a set that stores each edge including: parent node, child node, edge label. This implementation has the advantage of time complexity of o(1) when it comes to adding new edges. It also uses less data to store the graph compared to the other 2 represenions with a data complexity of o(E), E representing edges.
- The second implementation, a adjacency list is the one I used to represent my graph. It would store the graph in a Map, where each key is a node, and it value would be a set of edges. Each element in the set can be represented by a abstractMap that stores it first value as the child node which the parent node is connecting to and the edge label. The advantages of using this is the fact that it has o(1) time complexity when adding a new vertex or edge. However this representation does take up more data of o(V + E), v is vertices and E is edges. I chose this representation along with using a tree set for my set of outgoing edges because, it sorts the keys in lexicographical order, has o(logn) time for basic operations like addNOde, addEdge, and personally for me when I drew it out it was the easist one for me to implement.
- The third implementation is a adjacency matrix where every element is a a edge¡A, B¿ and a indicator of if there is a link between the 2 nodes and how many. Every element is 2 nodes and it will be a edge if there is a link between them and it will store each edge between them. The advantages of using a adjacency matrix is that it has a ease of access, however it has disadvantage which it stores unnecessary information because it stores every possible edge including the ones that do not have a edge between them.