

The graph is implemented using an adjacency list, where each node is a key in a HashMap, and its value is another HashMap that maps destination nodes to lists of edge labels. This structure allows efficient storage, especially for sparse graphs, since only existing connections are stored rather than an entire matrix. I chose this representation because it supports fast lookups, dynamic insertion and removal of nodes and edges, and efficient traversal of a node's neighbors.