Space complexity for the entire assignment is O(V+E). All variables that scale with input are either some variations of adjacency lists, heaps, or lists the size of the number of vertices. Hence, all variables are bound by O(V+E).

Functions buildHeap and buildHeapDisreg loop a maximum of E+V times. V is bounded by E (since each node at least has 1 edge), hence have time complexity of O(E).

augmentGraph loops a maximum of O+V times (number of lines in the file), hence also has time complexity of O(E).

reposition has time complexity of log(V) as it moved nodes up or down the heap, and can hence only loop log(V) times.

quickestPath and quickestSafePath have time complexity of O(E log V). The outer while loop in both loops V times (which is bounded by E). There is another loop inside, but that only runs a total of E times as well (for each adjacent node). reposition is called inside the main function which runs logV times, hence giving total time complexity of O(E log V).

addService loops maximum of V times and hence has time complexity of O(V).

quickestDetourPath performs dijkstra's twice, which is bound by time complexity of O(ElogV). It reverse self.a to create self.b, which has the same complexity as buildGraph of O(E). Collating the results has a time complexity of O(E), as it iterates the returnlists, and they cannot contain more items than E. Hence, quickestDetour has a time complexity of O(ElogV).