

1. Briefly describe how you constructed the path list in `returnPath()`. Which end of the list did you append edges to, and why?

- Briefly describe how you constructed the path list in `returnPath()`.
 - First, we need to check if the input 'endVertex' is not null.
 - Second, we need to pop the Edges out which are in a `parentEdge` variable. In `ShortestPaths` class, there is a `parentEdge` Hashmap variable which contains Vertex and Edge. The Edge which is in the `parentEdge` shows the previous Vertex in the shortest path. Once we get the Edge and then push it to the 'path' variable. And then repeat this process until the null point because the start Vertex has no parent Edge.
- Which end of the list did you append edges to, and why?
 - `returnPath()` function has to return path in order from a starting Vertex to an end Vertex. Therefore, when we put an element into `LinkedList<Edge>` path variable, we need to call `push()` function of a `ArrayList` class. Because we are following the shortest path backward direction, so the first Edge goes to the end and the last Edge goes to the first of the `ArrayList`.