# Eulerian Path Algorithm

Connect node with out-degree < in-degree to node with out-degree < in-degree. So that we will have an Eulerian cycle.

Why will you return to *u*?

Walk from some arbitrary node *u* until you return to *u*, creating a doubly liked list of the path you visit.

**Repeat** until all edges used:

*How can find such a node quickly?

- Start from some node *w* on the current tour with unused edges*.
- Walk along unused edges until you return to *w*, inserting the visited nodes after *w* into the current tour list.