Eulerian Tour

Stephen Huan
Edited by: Udbhav Muthakana

January 8, 2020

1 Definition

An ***Eulerian tour*** is path through a graph that uses each edge exactly once. If it starts and ends at the same node, then it is called an ***Eulerian circuit***.

1.1 Identification

For a graph to have an Eulerian tour, it has to be connected (strongly connected for a directed graph) and either:

1. Every node has an even degree (the number of incoming and outgoing edges).

2. Every node but two has an even degree.

In the second case, the two nodes are the start and end nodes of the circuit. Detecting strongly connected components can be done in linear time with Kosaraju-Sharir.

2 Implementation

Wikipedia recommends Hierholzer's algorithm but I found it confusing and convoluted. Just do a postorder iterative DFS on the graph, adding nodes to the tour once their children have been processed.

3 Applications

In order to find the lowest common ancestor (LCA) between two nodes in a tree, one approach is to run a Eulerian tour, doubling up edges between parent and children, and then do a range minimum query (RMQ) on the resulting tour between the two nodes.

3.1 Sample Implementation

Eulerian tour

```
def eulerian_tour(graph):
    # remove nodes with no edges
    graph = {k: v for k, v in graph.items() if len(v) > 0}
    count = sum(len(graph[v]) % 2 for v in graph)
    if count != 0 and count != 2:
        return # no such tour
    start = [v for v in graph if len(graph[v]) % 2 == 1][0] if count ==
    ↪  2 else 0

    stk = [(start, 0)]
    tour = deque([])
    seen = set()
    while len(stk) > 0:
        n, p = stk[-1]
        # done with children
        if len(graph[n]) == p:
            tour.appendleft(n)
            stk.pop()
        # has children left to process
        else:
            child = graph[n][p]
            # edges have a unique id - (vertex, id)
            if child[1] not in seen:
                stk.append((child[0], 0))
                seen.add(child[1])
                stk[-2] = (n, p + 1)
            else:
                stk[-1] = (n, p + 1)
    return list(map(lambda x: x + 1, tour))
```

4 Sample Problems

1. USACO Training: Riding the Fence

   Solution: Direct implementation of an Eulerian Tour.

5 Works Cited

1. Wikipedia - “Eulerian path”