Algorithm for Constructing Euler Tours

**Input:** A connected graph $G = (V, E)$ with even degrees for all vertices, i.e. $d_G(v)$ is even for all $v \in V$.

**Output:** An eulerian tour in $G$.

Step 1 Choose any starting vertex.

Step 2 Begin tracing a trail randomly from the starting vertex until you get stuck, i.e. cannot continue to do so without repeating an edge. (By the property of the given graph, you should get stuck at the starting vertex).

Step 3 If the trail covers all edges in $G$ then we are done. STOP. Otherwise, name this trail an approximate Euler tour and go to Step 4.

Step 4 Choose a new starting vertex which is on the approximate Euler tour generated so far, and which also has at least one unused edge incident to it.

Step 5 Begin tracing a trail randomly from the starting vertex, avoiding edges already on the current approximate Euler tour generated so far. Continue until you get stuck, i.e. cannot do so without repeating an edge. (By the property of the given graph, you should get stuck at the vertex you started out with in Step 4).

Step 6 Create a new approximate Euler tour by concatenating the current approximate Euler tour with the trail obtained from Step 5, in the following way:

(a) Trace the current approximate Euler tour till you get to the starting vertex of Step 5.

(b) Follow through with the trail just obtained in Step 5, returning to its starting vertex.

(c) Finish tracing the current approximate Euler tour.

Step 7 If this new approximate Euler tour covers all edges in $G$ i.e. if it is an Euler tour of $G$ then STOP. Otherwise, name this trail the current approximate Euler tour and repeat Step 4.