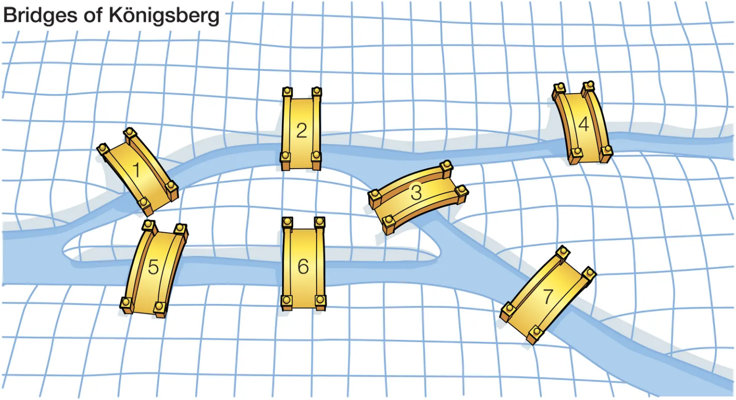
Proof For Closed Eulerian Trails

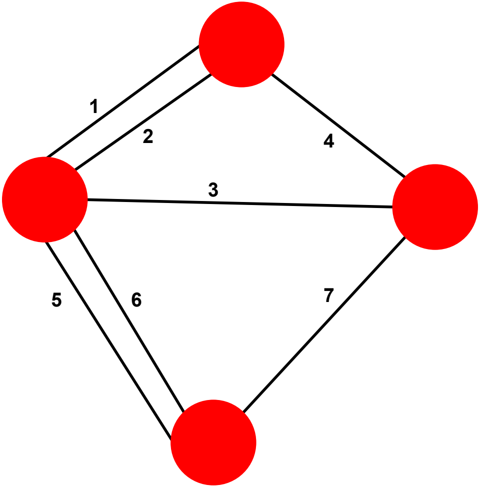
Graham Swain

September 13, 2022
Math 479

Königsberg Bridge Problem



Königsberg Bridge Problem



Definition

A **walk** is a sequence of vertices in which each consecutive pair of vertices are adjacent.

Definition

A **walk** is a sequence of vertices in which each consecutive pair of vertices are adjacent.

Definition

A **trail** is a walk where edges are not allowed to be repeated.

Definition

A **walk** is a sequence of vertices in which each consecutive pair of vertices are adjacent.

Definition

A **trail** is a walk where edges are not allowed to be repeated.

- It is called a **closed trail** when it begins and ends at the same vertex.

Definition

A **walk** is a sequence of vertices in which each consecutive pair of vertices are adjacent.

Definition

A **trail** is a walk where edges are not allowed to be repeated.

- It is called a **closed trail** when it begins and ends at the same vertex.

Definition

If a trail uses all the edges in a graph it is called a **Eulerian trail**.

Definition

A **walk** is a sequence of vertices in which each consecutive pair of vertices are adjacent.

Definition

A **trail** is a walk where edges are not allowed to be repeated.

- It is called a **closed trail** when it begins and ends at the same vertex.

Definition

If a trail uses all the edges in a graph it is called a **Eulerian trail**.

- A **closed Eulerian trail** is when an Eulerian trail begins and ends on the same vertex.

Definition

A graph is **connected** if, and only if, there is a walk between any two vertices v and w.

Definition

A graph is **connected** if, and only if, there is a walk between any two vertices v and w.

Definition

The **degree** of a vertex, v, is the number of edges incident with v.

Theorem (Euler)

A connected graph has a closed Eulerian trail if and only if all of its vertices have even degree.

Proof.

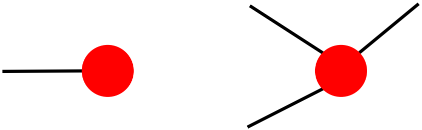
$\Rightarrow$ Assume a connected graph G has a closed Eulerian trail, denoted as C. Prove that every vertex has an even degree.

Take an arbitrary vertex v. As we traverse C, each time we enter v we must be able to exit on a distinct edge. Thus every vertex must be adjacent to $2k$ edges, where k is the number of times a vertex is visited. Therefore every vertex has even degree.

Proof.

$\Rightarrow$ Assume a connected graph G has a closed Eulerian trail, denoted as C. Prove that every vertex has an even degree.

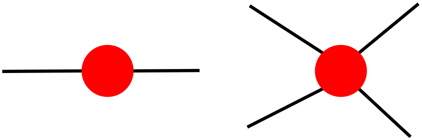
Take an arbitrary vertex v. As we traverse C, each time we enter v we must be able to exit on a distinct edge. Thus every vertex must be adjacent to $2k$ edges, where k is the number of times a vertex is visited. Therefore every vertex has even degree.



Proof.

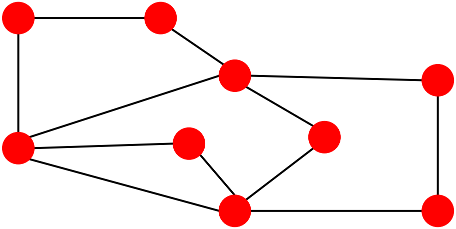
$\Rightarrow$ Assume a connected graph G has a closed Eulerian trail, denoted as C. Prove that every vertex has an even degree.

Take an arbitrary vertex v. As we traverse C, each time we enter v we must be able to exit on a distinct edge. Thus every vertex must be adjacent to $2k$ edges, where k is the number of times a vertex is visited. Therefore every vertex has even degree.



Proof (cont).

$\Leftarrow$ Assume G is a connected graph with every vertex having even degree. Prove that it has a closed Eulerian trail.

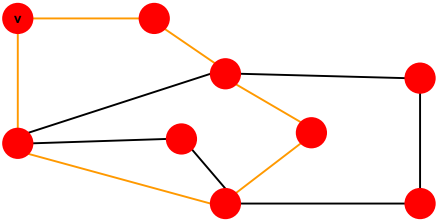


Proof (cont).

Starting at an arbitrary vertex v, begin constructing a trail. Continue until we reach a vertex we cannot leave. If this vertex is not v, we entered on one edge and do not have a corresponding exit, meaning this vertex has an odd degree, which is a contradiction. So we know that it is v, meaning we have found a closed trail, which we will called C_1.

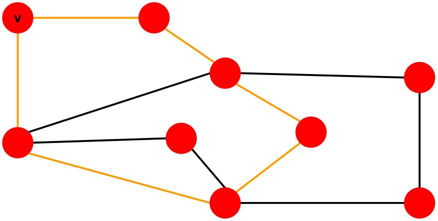
Proof (cont).

Starting at an arbitrary vertex v, begin constructing a trail. Continue until we reach a vertex we cannot leave. If this vertex is not v, we entered on one edge and do not have a corresponding exit, meaning this vertex has an odd degree, which is a contradiction. So we know that it is v, meaning we have found a closed trail, which we will called C_1.



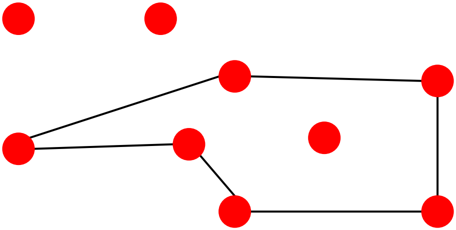
Proof (cont).

If C_1 contains all the edges of G, we have found an Eulerian trail and are done, if not remove all the edges of C_1 from G, let's call the resulting graph G'. G' is not necessarily connected.



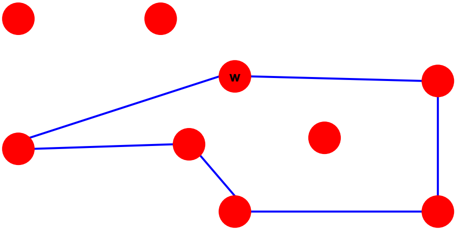
Proof (cont).

If C_1 contains all the edges of G, we have found an Eulerian trail and are done, if not remove all the edges of C_1 from G, let's call the resulting graph G'. G' is not necessarily connected.



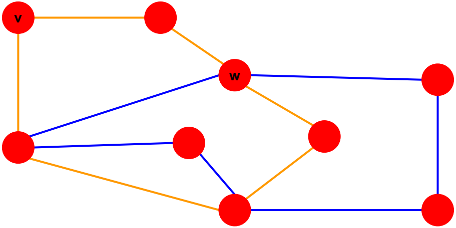
Proof (cont).

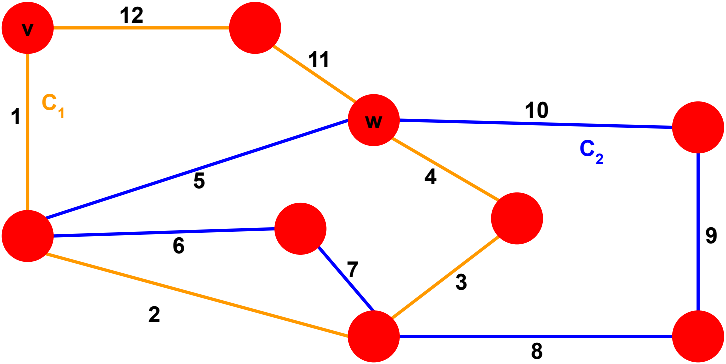
Select a vertex w that is in both C_1 and G. Then begin tracing a trail in G' Starting at w until we arrive back at w. Name the resulting closed trail C_2.



Proof (cont).

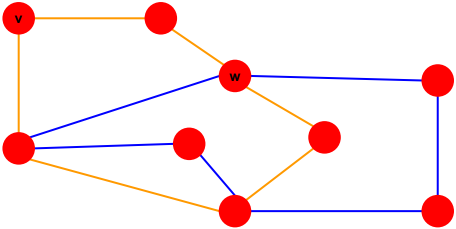
Combine C_1 and C_2 to form another closed trail denoted as C'. If C' contains all the edges of G, we have found a closed Eulerian trail. If not, remove all the edges of C' from G to get G''. Choose a vertex in both C' and G'' and repeat the steps above until we have used all the edges of G. $\square$




v
12
11
w
10
C_1
1
C_2
5
4
9
6
7
3
2
8

Proof cont.

Combine C_1 and C_2 to form another closed trail denoted as C'. If C' contains all the edges of G, we have found a closed Eulerian trail. If not, remove all the edges of C' from G to get G''. Choose a vertex in both C' and G'' and repeat the steps above until we have used all the edges of G. $\square$



Applications

- Computer Science

Applications

- Computer Science
- Networks

Applications

- Computer Science
- Networks
- Routing

References

[1] Belcastro, *Discrete Mathematics with Ducks*, CRC Press, 2012.

[2] Epp, *Discrete Mathematics with Applications*, Brooks Cole, 1996.