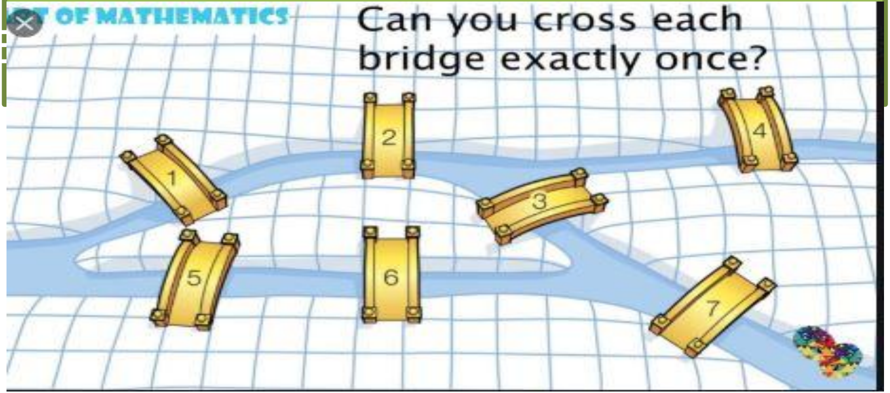
Euler and Semi-Euler Graphs

Teacher Incharge: Adil Mudasir



Euler Paths and Euler Circuits

An **Euler path** is a path that uses every edge of a graph exactly once.

An **Euler circuit/cycle** is a circuit that uses every edge of a graph exactly once.

- An Euler path starts and ends at different vertices.
- An Euler circuit starts and ends at the same vertex.

Note:

Graph containing Euler cycle is k/a Eular graph

Graph containing Euler path only is k/a Semi- eular graph

Euler Paths and Euler Circuits

An **Euler path** is a path that uses every edge of a graph exactly once.

An **Euler circuit/cycle** is a circuit that uses every edge of a graph exactly once.

- An Euler path starts and ends at different vertices.
- An Euler circuit starts and ends at the same vertex.

Note:

Graph containing Euler cycle is k/a Eular graph

Graph containing Euler path only is k/a Semi- eular graph

Euler Paths and Euler Circuits

An **Euler path** is a path that uses every edge of a graph exactly once.

An **Euler circuit/cycle** is a circuit that uses every edge of a graph exactly once.

- An Euler path starts and ends at different vertices.
- An Euler circuit starts and ends at the same vertex.

Note:

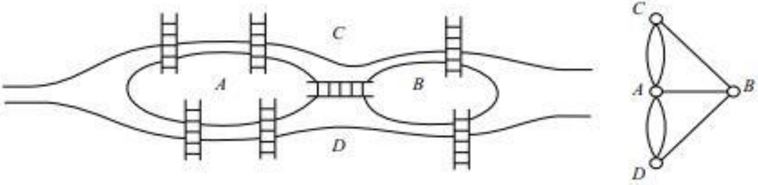
Graph containing Euler cycle is k/a Eular graph

Graph containing Euler path only is k/a Semi- eular graph

History: Koinsberg's Bridge Problem

Two islands A and B formed by the Pregal river (now Pregolya) in Konigsberg (then the capital of east Prussia, but now renamed Kaliningrad and in west Soviet Russia) were connected to each other and to the banks C and D with seven bridges. The problem is to start at any of the four land areas, A, B, C, or D, walk over each of the seven bridges exactly once and return to the starting point.

Euler modeled the problem representing the four land areas by four vertices, and the seven bridges by seven edges joining these vertices. This is illustrated in Figure 3.3.



Konigsberg Bridges

Underlying Graph

Fig. 3.3

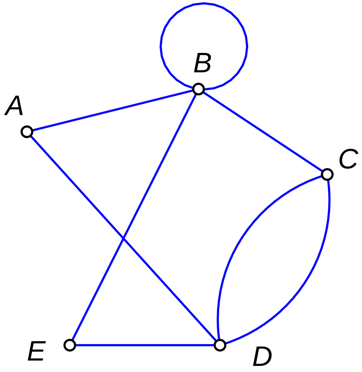
Solution of Koinsberg’s Bridge Problem

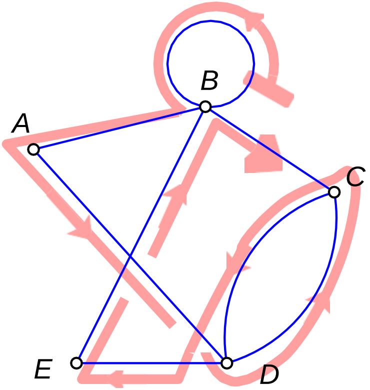
Theorem: An Eulerian trail exists in a connected graph if and only if there are either no odd vertices or two odd vertices.

For the case of no odd vertices, the path can begin at any vertex and will end there; for the case of two odd vertices, the path must begin at one odd vertex and end at the other. Any finite connected graph with two odd vertices is traversable. A traversable trail may begin at either odd vertex and will end at the other odd vertex.

Note: From this we can see that it is not possible to solve the bridges of Königsberg problem because there exists within the graph more than 2 vertices of odd degree.

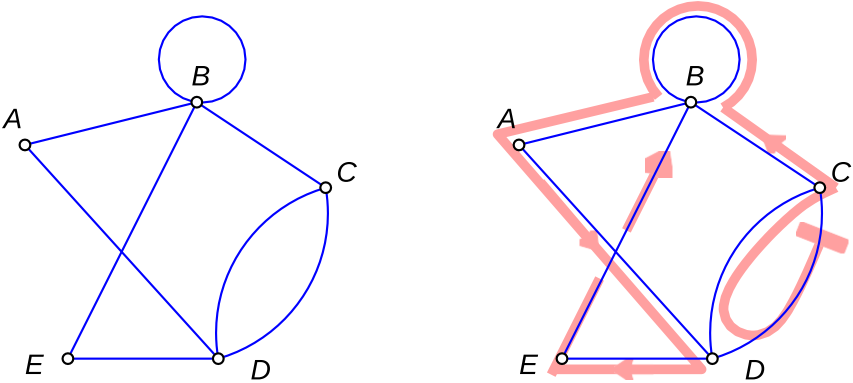
Euler Paths and Euler Circuits





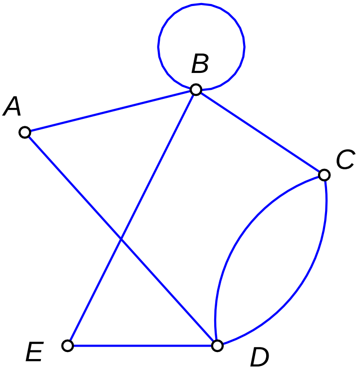
An Euler path: BBADCDEBC

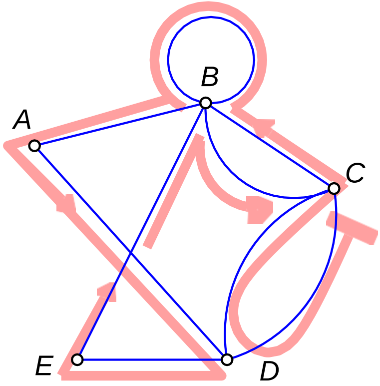
Euler Paths and Euler Circuits



Another Euler path: CDCBBADEB

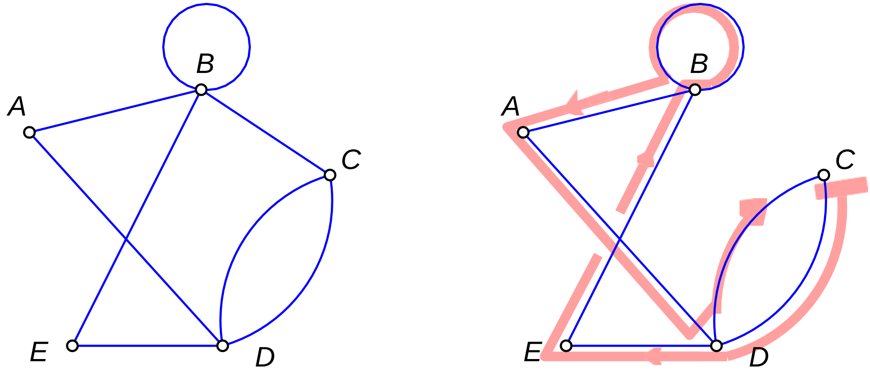
Euler Paths and Euler Circuits





An Euler circuit: CDCBBADEBC

Euler Paths and Euler Circuits



Another Euler circuit: CDEBBADC

Euler Paths and Euler Circuits

Is it possible to determine whether a graph has an Euler path or an Euler circuit, without necessarily having to find one explicitly?

The criterion/theorems for Euler Paths

Suppose that a graph has an Euler path P.

The criterion/theorems for Euler Paths

Suppose that a graph has an Euler path *P*.

For every vertex *v* other than the starting and ending vertices, the path *P* enters *v* the same number of times that it leaves *v* (say *s* times).

The criterion/theorems for Euler Paths

Suppose that a graph has an Euler path P.

For every vertex v other than the starting and ending vertices, the path P enters v the same number of times that it leaves v (say s times).

Therefore, there are $2s$ edges having v as an endpoint.

The criterion/theorems for Euler Paths

Suppose that a graph has an Euler path *P*.

For every vertex *v* other than the starting and ending vertices, the path *P* enters *v* the same number of times that it leaves *v* (say s times).

Therefore, there are $2s$ edges having *v* as an endpoint.

Therefore, all vertices other than the two endpoints of P must be even vertices.

The criterion/theorems for Euler Paths

Suppose the Euler path P starts at vertex x and ends at y.

The criterion/theorems for Euler Paths

Suppose the Euler path P starts at vertex x and ends at y.

Then P leaves x one more time than it enters, and leaves y one fewer time than it enters.

The criterion/theorems for Euler Paths

Suppose the Euler path P starts at vertex x and ends at y.

Then P leaves x one more time than it enters, and leaves y one fewer time than it enters.

Therefore, **the two endpoints of P must be odd vertices.**

The criterion/theorems for Euler Paths

The inescapable conclusion ("based on reason alone!"):

If a graph *G* has an Euler path, then it must have exactly two odd vertices.

Or, to put it another way,

If the number of odd vertices in *G* is anything other than 2, then *G* cannot have an Euler path.

The criterion/theorems for Euler Circuits

- Suppose that a graph G has an Euler circuit C.

The criterion/theorems for Euler Circuits

- Suppose that a graph *G* has an Euler circuit *C*.
- For every vertex *v* in *G*, each edge having *v* as an endpoint shows up *exactly once* in *C*.

The criterion/theorems for Euler Circuits

- Suppose that a graph G has an Euler circuit C.
- For every vertex v in G, each edge having v as an endpoint shows up *exactly once* in C.
- The circuit C enters v the same number of times that it leaves v (say s times), so v has degree $2s$.

The criterion/theorems for Euler Circuits

- Suppose that a graph *G* has an Euler circuit *C*.
- For every vertex *v* in *G*, each edge having *v* as an endpoint shows up *exactly once* in *C*.
- The circuit *C* enters *v* the same number of times that it leaves *v* (say *s* times), so *v* has degree 2*s*.
- That is, **v must be an even vertex.**

The criterion/theorems for Euler Circuits

The inescapable conclusion ("based on reason alone"):

If a graph *G* has an Euler circuit, then all of its vertices must be even vertices.

Or, to put it another way,

If the number of odd vertices in *G* is anything other than 0, then *G* cannot have an Euler circuit.

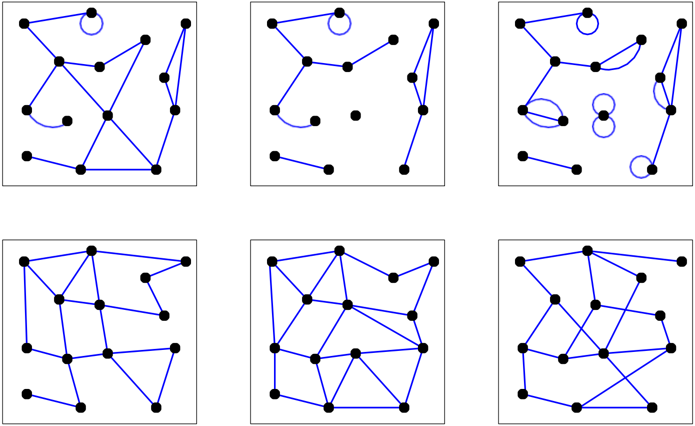
Things You Should Be Wondering

- Does **every** graph with **zero** odd vertices have an Euler circuit?

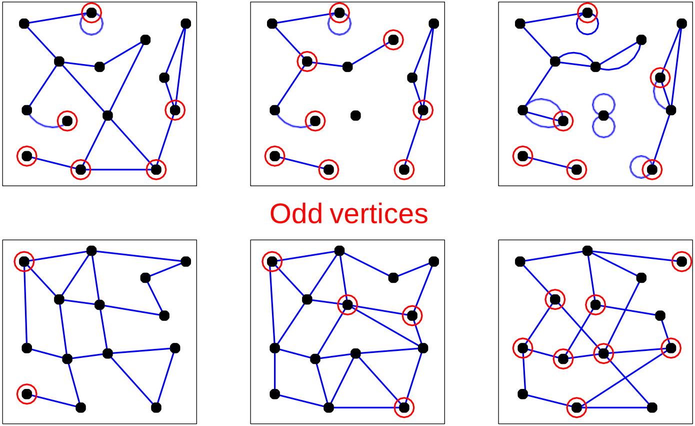
- Does **every** graph with **two** odd vertices have an Euler path?

- Is it possible for a graph have just **one** odd vertex?

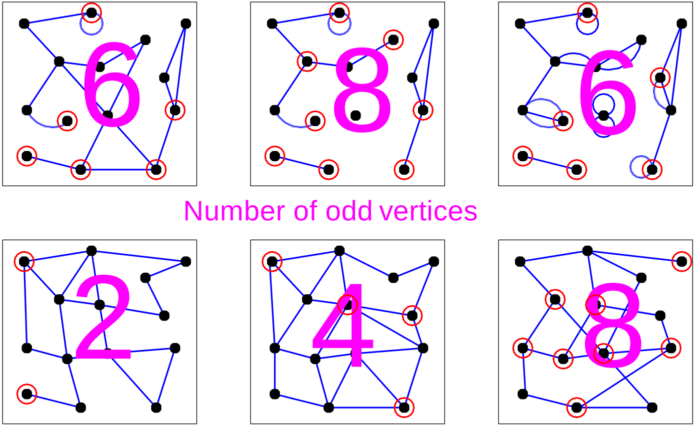
How Many Odd Vertices?



How Many Odd Vertices?



How Many Odd Vertices?



The Handshaking Theorem

The Handshaking Theorem says that

In every graph, the sum of the degrees of all vertices equals twice the number of edges.

If there are n vertices $V_1, \ldots, V_n$, with degrees $d_1, \ldots, d_n$, and there are e edges, then

$$d_1 + d_2 + \cdots + d_{n-1} + d_n = 2e$$

Or, equivalently,

$$e = \frac{d_1 + d_2 + \cdots + d_{n-1} + d_n}{2}$$

The Handshaking Theorem

Why "Handshaking"?

If n people shake hands, and the i^{th} person shakes hands d_i times, then the total number of handshakes that take place is

$$\frac{d_1 + d_2 + \cdots + d_{n-1} + d_n}{2}.$$

(How come? Each handshake involves two people, so the number $d_1 + d_2 + \cdots + d_{n-1} + d_n$ counts every handshake twice.)

The Number of Odd Vertices

- The number of edges in a graph is

$$\frac{d_1 + d_2 + \cdots + d_n}{2}$$

which must be an **integer**.

The Number of Odd Vertices

- The number of edges in a graph is

$$\frac{d_1 + d_2 + \cdots + d_n}{2}$$

which must be an **integer.**

- Therefore, $d_1 + d_2 + \cdots + d_n$ must be an **even number.**

The Number of Odd Vertices

- The number of edges in a graph is

$$\frac{d_1 + d_2 + \cdots + d_n}{2}$$

  which must be an **integer.**

- Therefore, $d_1 + d_2 + \cdots + d_n$ must be an **even number.**

- Therefore, the numbers $d_1, d_2, \cdots, d_n$ must include an **even number of odd numbers.**

The Number of Odd Vertices

- The number of edges in a graph is

$$\frac{d_1 + d_2 + \cdots + d_n}{2}$$

  which must be an **integer.**

- Therefore, $d_1 + d_2 + \cdots + d_n$ must be an **even number.**

- Therefore, the numbers $d_1, d_2, \cdots, d_n$ must include an **even number of odd numbers.**

- **Every graph has an even number of odd vertices!**

Back to Euler Paths and Circuits

Here's what we know so far:

# odd vertices	Euler path?	Euler circuit?
0	No	Maybe
2	Maybe	No
4, 6, 8, . . .	No	No
1, 3, 5, . . .	*No such graphs exist!*	

Can we give a better answer than "maybe"?

Euler Paths and Circuits — The Last Word

Here is the answer Euler gave:

# odd vertices	Euler path?	Euler circuit?
0 2 4, 6, 8, . . .	No **Yes*** No	**Yes*** No No
1, 3, 5,	No such graphs exist	

* *Provided the graph is connected.*

Euler Paths and Circuits — The Last Word

Here is the answer Euler gave:

# odd vertices	Euler path?	Euler circuit?
0	No	**Yes***
2	**Yes***	No
4, 6, 8, . . .	No	No
1, 3, 5,	No such graphs exist	

* *Provided the graph is connected.*

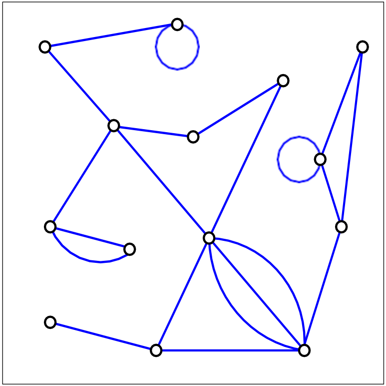
Next question: If an Euler path or circuit exists, how do you find it?

Bridges

Removing a single edge from a connected graph can make it disconnected. Such an edge is called a **bridge**.

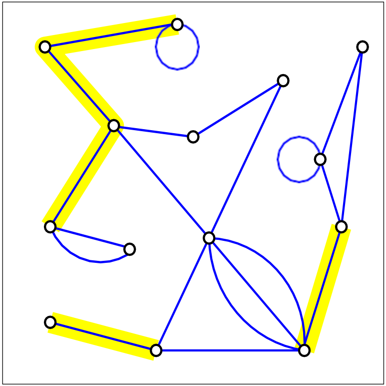
Bridges

Removing a single edge from a connected graph can make it disconnected. Such an edge is called a **bridge**.



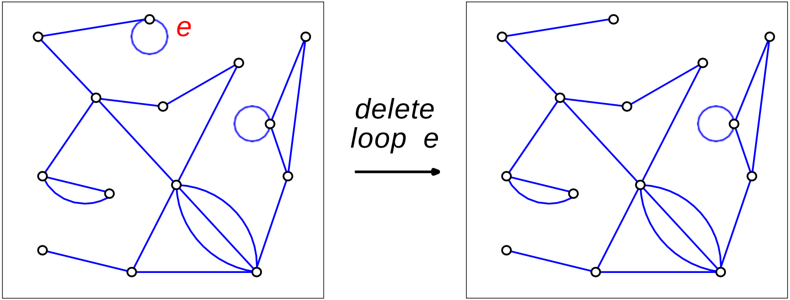
Bridges

Removing a single edge from a connected graph can make it disconnected. Such an edge is called a **bridge**.



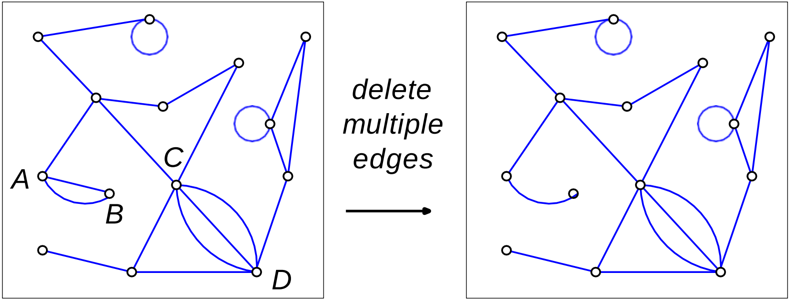
Bridges

Loops cannot be bridges, because removing a loop from a graph cannot make it disconnected.



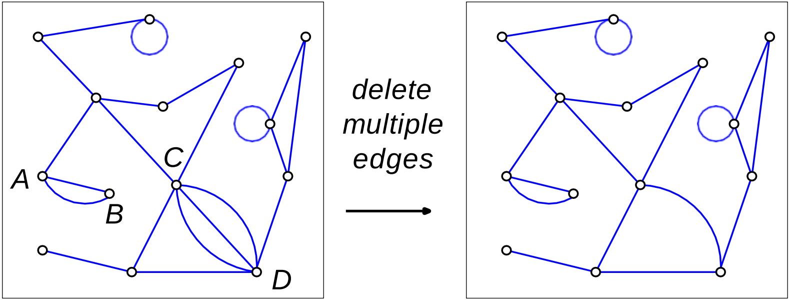
Bridges

If two or more edges share both endpoints, then removing any one of them cannot make the graph disconnected. Therefore, none of those edges is a bridge.



Bridges

If two or more edges share both endpoints, then removing any one of them cannot make the graph disconnected. Therefore, none of those edges is a bridge.

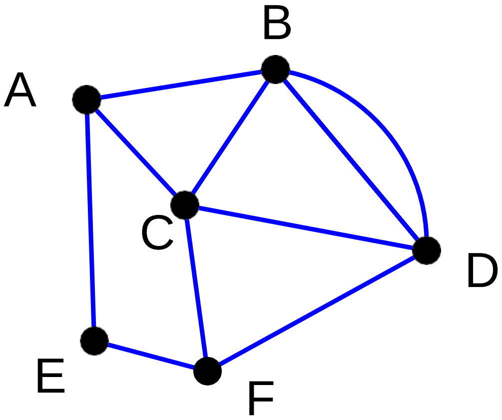


Finding Euler Circuits and Paths

“Don’t burn your bridges.”

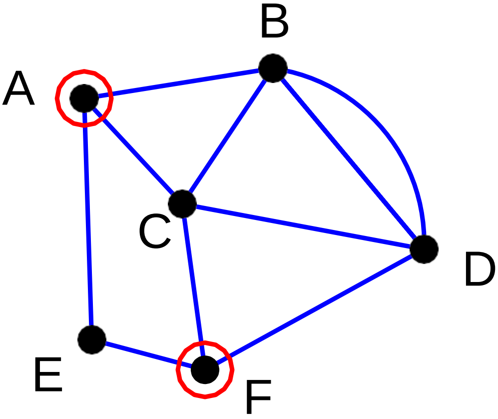
Finding Euler Circuits and Paths

Problem: Find an Euler circuit in the graph below.



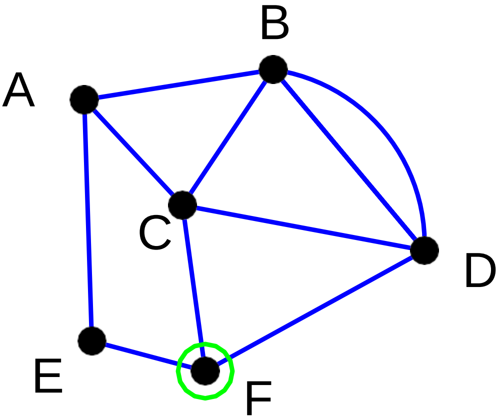
Finding Euler Circuits and Paths

There are two odd vertices, A and F. Let's start at F.



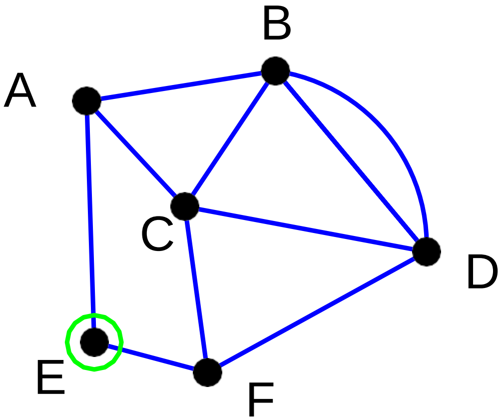
Finding Euler Circuits and Paths

Start walking at F. When you use an edge, delete it.



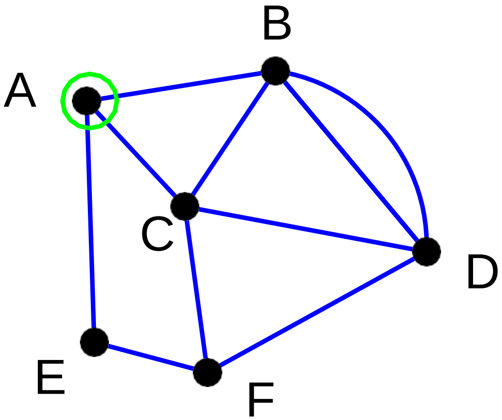
Finding Euler Circuits and Paths

Path so far: FE



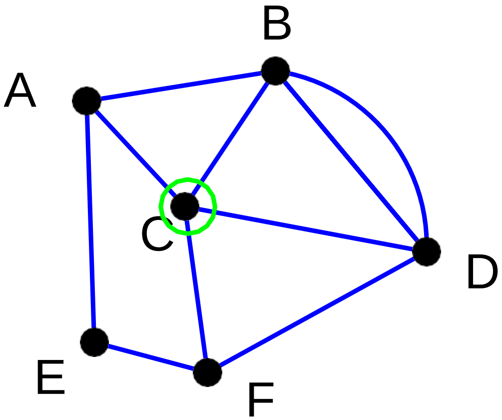
Finding Euler Circuits and Paths

Path so far: FEA



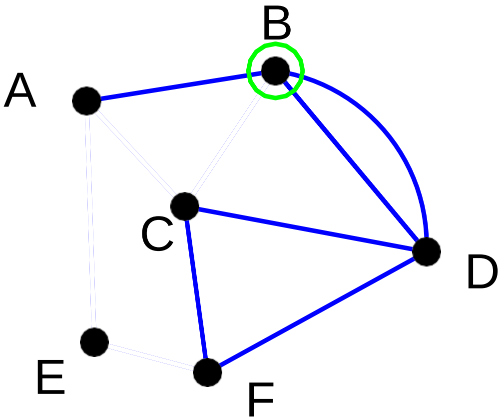
Finding Euler Circuits and Paths

Path so far: FEAC



Finding Euler Circuits and Paths

Path so far: FEACB



Finding Euler Circuits and Paths

Up until this point, the choices didn’t matter.

Finding Euler Circuits and Paths

Up until this point, the choices didn't matter.

But now, crossing the edge BA would be a mistake, because we would be stuck there.

Finding Euler Circuits and Paths

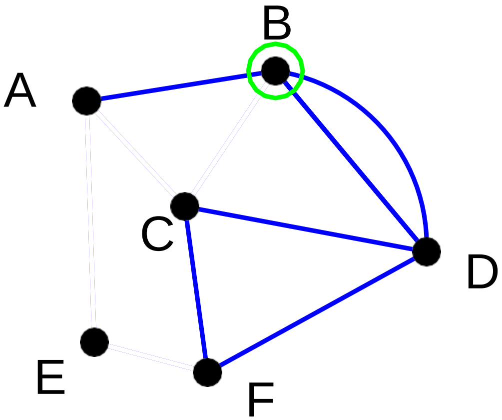
Up until this point, the choices didn't matter.

But now, crossing the edge BA would be a mistake, because we would be stuck there.

The reason is that BA is a **bridge**. We don't want to cross ("burn"?) a bridge unless it is the only edge available.

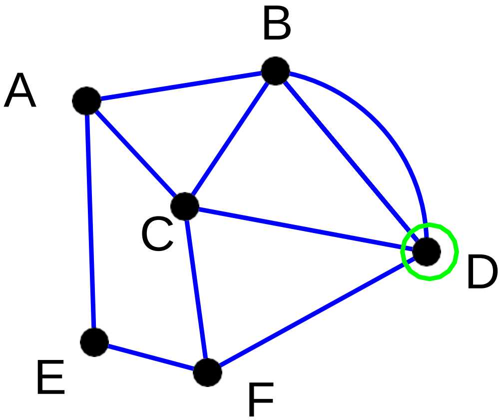
Finding Euler Circuits and Paths

Path so far: FEACB



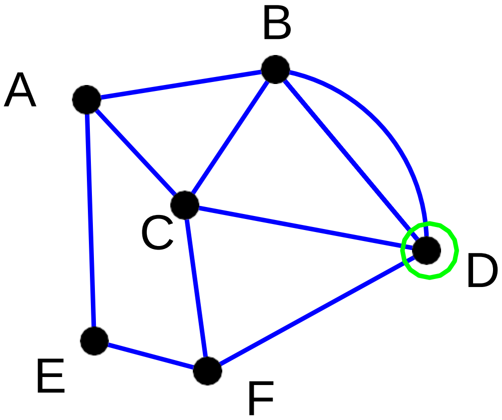
Finding Euler Circuits and Paths

Path so far: FEACBD.



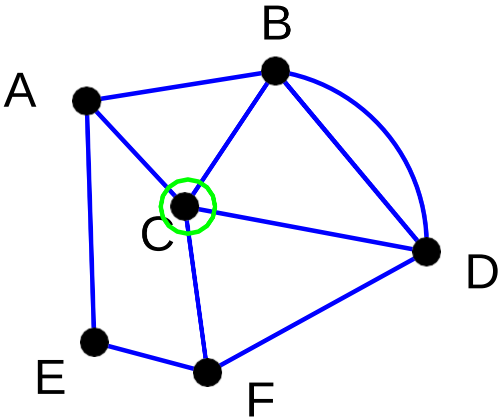
Finding Euler Circuits and Paths

Path so far: FEACBD. Don't cross the bridge!



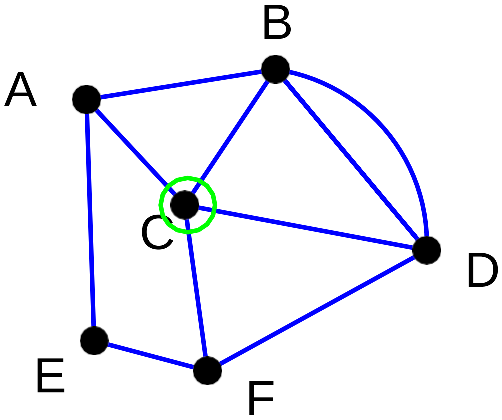
Finding Euler Circuits and Paths

Path so far: FEACBDC



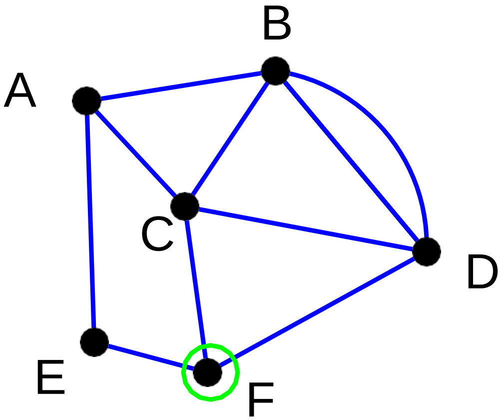
Finding Euler Circuits and Paths

Path so far: FEACBDC Now we have to cross the bridge CF.



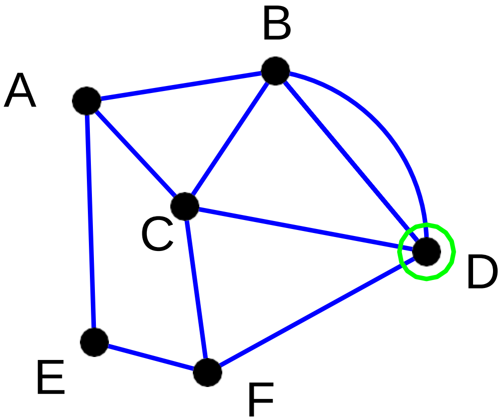
Finding Euler Circuits and Paths

Path so far: FEACBDCF



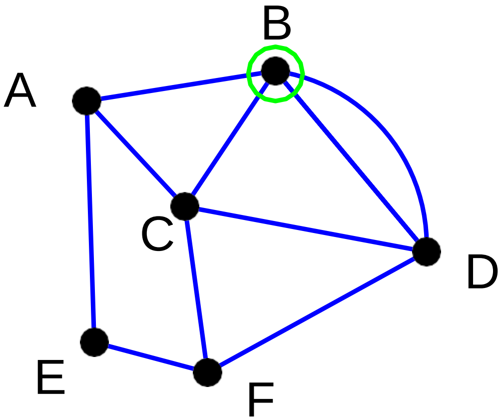
Finding Euler Circuits and Paths

Path so far: FEACBDCFD



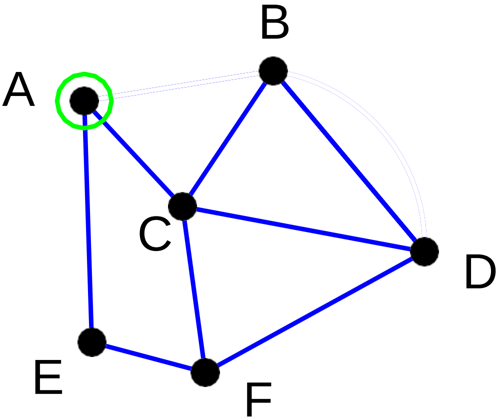
Finding Euler Circuits and Paths

Path so far: FEACBDCFDB



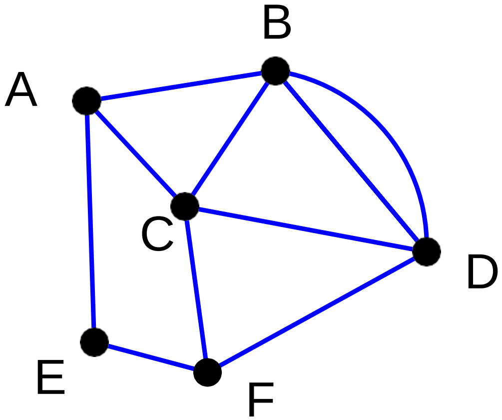
Finding Euler Circuits and Paths

Euler Path: FEACBDCFDBA



Finding Euler Circuits and Paths

Euler Path: FEACBDCFDBA



Fleury's Algorithm

To find an Euler path or an Euler circuit:

Fleury's Algorithm

To find an Euler path or an Euler circuit:

1. Make sure the graph has either 0 or 2 odd vertices.

Fleury's Algorithm

To find an Euler path or an Euler circuit:

1. Make sure the graph has either 0 or 2 odd vertices.
2. If there are 0 odd vertices, start anywhere. If there are 2 odd vertices, start at one of them.

Fleury’s Algorithm

To find an Euler path or an Euler circuit:

1. Make sure the graph has either 0 or 2 odd vertices.

2. If there are 0 odd vertices, start anywhere. If there are 2 odd vertices, start at one of them.

3. Follow edges one at a time. If you have a choice between a bridge and a non-bridge, always choose the non-bridge.

Fleury's Algorithm

To find an Euler path or an Euler circuit:

1. Make sure the graph has either 0 or 2 odd vertices.
2. If there are 0 odd vertices, start anywhere. If there are 2 odd vertices, start at one of them.
3. Follow edges one at a time. If you have a choice between a bridge and a non-bridge, always choose the non-bridge.
4. Stop when you run out of edges.

Fleury's Algorithm

To find an Euler path or an Euler circuit:

1. Make sure the graph has either 0 or 2 odd vertices.

2. If there are 0 odd vertices, start anywhere. If there are 2 odd vertices, start at one of them.

3. Follow edges one at a time. If you have a choice between a bridge and a non-bridge, always choose the non-bridge.

4. Stop when you run out of edges.

This is called **Fleury's algorithm**, and it always works!

Fleury's Algorithm: Another Example