Euler Paths and Euler Circuits

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

An **Euler circuit** 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.
Euler Paths and Euler Circuits

An Euler path: BBADADCDEBC
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: CDEBBADDC
Is it possible to determine whether a graph has an Euler path or an Euler circuit, without necessarily having to find one explicitly?
Suppose that a graph has an Euler path $P$. 

The Criterion for Euler Paths

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

For every vertex $\nu$ other than the starting and ending vertices, the path $P$ enters $\nu$ the same number of times that it leaves $\nu$ (say $s$ times).
The Criterion 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.
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.
Suppose the Euler path $P$ starts at vertex $x$ and ends at $y$. 
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.
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 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.
Suppose that a graph $G$ has an Euler circuit $C$. 
The Criterion 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 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 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 $2s$.

- That is, $v$ must be an even vertex.
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.
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 to have just one odd vertex?
How Many Odd Vertices?
How Many Odd Vertices?
How Many Odd Vertices?

Number of odd vertices

6
8
6
2
4
8
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 edges in a graph is

\[
\frac{d_1 + d_2 + \cdots + d_n}{2}
\]

which must be an integer.

Therefore, the numbers \(d_1, d_2, \cdots, d_n\) must include an even number of odd numbers.

Therefore, every graph has an even number of odd vertices!
The number of edges in a graph is

\[ d_1 + d_2 + \cdots + d_n \]

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 \( d_1, d_2, \ldots, d_n \) must include an even number of odd numbers.
The number of edges in a graph is

\[
\frac{d_1 + d_2 + \cdots + d_n}{2}
\]

which must be an \textbf{integer}.

Therefore, \(d_1 + d_2 + \cdots + d_n\) must be an \textbf{even number}.

Therefore, the numbers \(d_1, d_2, \cdots, d_n\) must include an \textbf{even number of odd numbers}.

\textbf{Every graph has an even number of odd vertices!}
Here’s what we know so far:

<table>
<thead>
<tr>
<th># odd vertices</th>
<th>Euler path?</th>
<th>Euler circuit?</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>No</td>
<td>Maybe</td>
</tr>
<tr>
<td>2</td>
<td>Maybe</td>
<td>No</td>
</tr>
<tr>
<td>4, 6, 8, ...</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>1, 3, 5, ...</td>
<td>No such graphs exist!</td>
<td>No</td>
</tr>
</tbody>
</table>

Can we give a better answer than “maybe”? 
Here is the answer Euler gave:

<table>
<thead>
<tr>
<th># odd vertices</th>
<th>Euler path?</th>
<th>Euler circuit?</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>No</td>
<td>Yes*</td>
</tr>
<tr>
<td>2</td>
<td>Yes*</td>
<td>No</td>
</tr>
<tr>
<td>4, 6, 8, …</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>1, 3, 5,</td>
<td>No such graphs exist</td>
<td></td>
</tr>
</tbody>
</table>

* Provided the graph is connected.
Euler Paths and Circuits — The Last Word

Here is the answer Euler gave:

<table>
<thead>
<tr>
<th># odd vertices</th>
<th>Euler path?</th>
<th>Euler circuit?</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>No</td>
<td>Yes*</td>
</tr>
<tr>
<td>2</td>
<td>Yes*</td>
<td>No</td>
</tr>
<tr>
<td>4, 6, 8, ...</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>1, 3, 5,</td>
<td>No such graphs exist</td>
<td></td>
</tr>
</tbody>
</table>

* Provided the graph is connected.

Next question: If an Euler path or circuit exists, how do you find it?
Removing a single edge from a connected graph can make it disconnected. Such an edge is called a **bridge**.
Removing a single edge from a connected graph can make it disconnected. Such an edge is called a bridge.
Removing a single edge from a connected graph can make it disconnected. Such an edge is called a bridge.
Loops cannot be bridges, because removing a loop from a graph cannot make it disconnected.
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.
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.
“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

Diagram with vertices A, B, C, D, E, F and edges connecting them to form a graph.
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
Up until this point, the choices didn’t matter.
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.
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.
Path so far: FEACBD.  

Don’t cross the bridge!
Finding Euler Circuits and Paths

Path so far: FEACBDC
Path so far: FEACBDC  Now we have to cross the bridge CF.
Finding Euler Circuits and Paths

Path so far: FEACBDCF
Path so far: FEACBDCFD
Finding Euler Circuits and Paths

Path so far: FEACBDCFDB
Finding Euler Circuits and Paths

Euler Path: FEACBDCFDBA
Euler Path: FEACBDCFDBA
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

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.
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