Euler's Theorems & Fleury's Algorithm

Notes 24 – Sections 5.4 & 5.5

Essential Learnings

- Students will understand and be able to use Euler's Theorems to determine if a graph has an Euler Circuit or an Euler Path.

Euler’s Theorems

In this section we are going to develop the basic theory that will allow us to determine if a graph has an Euler circuit, an Euler path, or neither.

EULER'S CIRCUIT THEOREM

- If a graph is *connected* and *every vertex is even*, then it has an Euler circuit (at least one, usually more).
- If a graph has *any odd vertices*, then it does not have an Euler circuit.

How to Use the Theorem

Step 1 – Determine if the graph is connected. If it isn't, then an Euler circuit is impossible.

Step 2 – If the graph is connected, then we determine the degree of each vertex.

If a vertex has an odd degree, we know that an Euler circuit is no possible.

If there are no odd vertices, then we know that the answer is yes–the graph does have an Euler circuit!

Illustration using the Theorem

The graphs shown illustrate two of the three possible scenarios.

The graph cannot have an Euler circuit because it is disconnected.

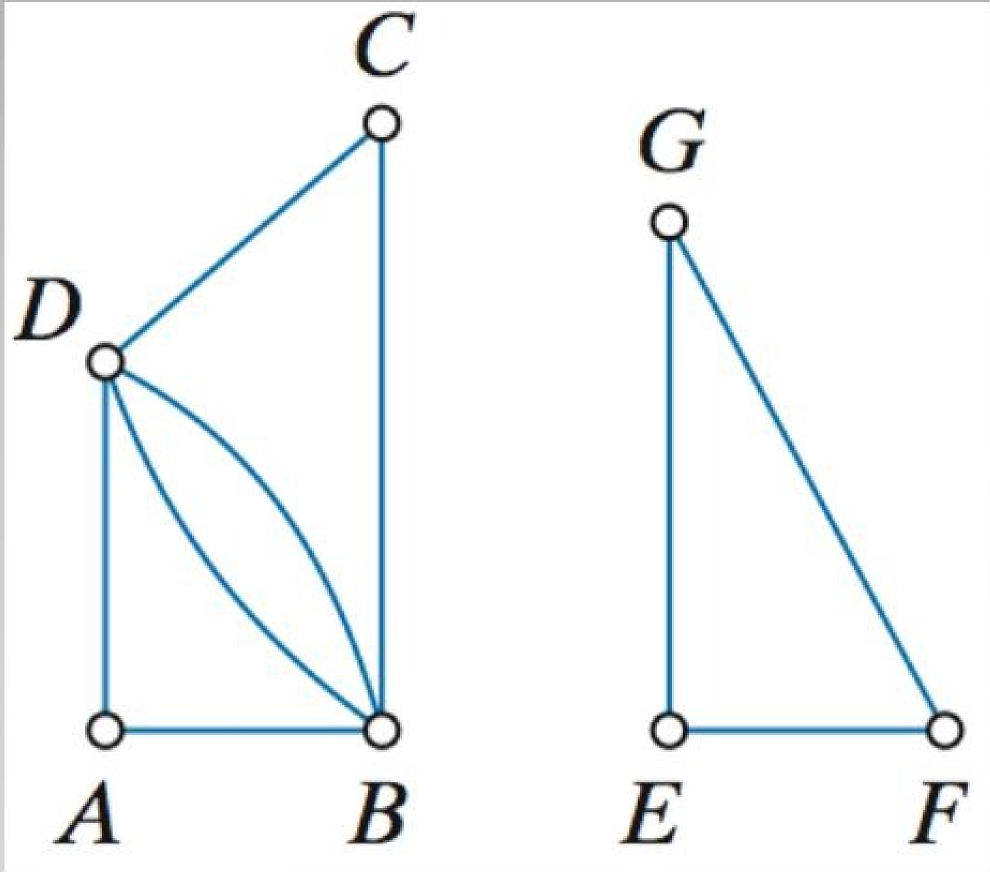


Illustration using the Theorem

This graph is connected, but we can quickly spot odd vertices (C is one of them; there are others).

This graph has no Euler circuits.

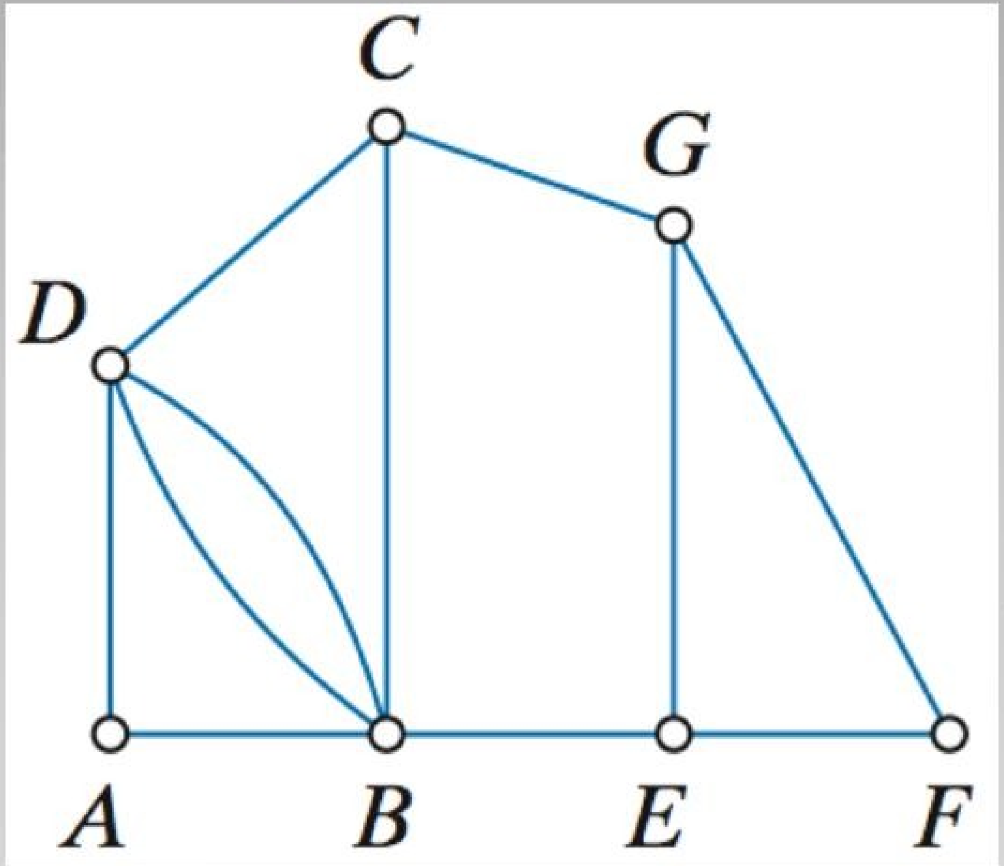
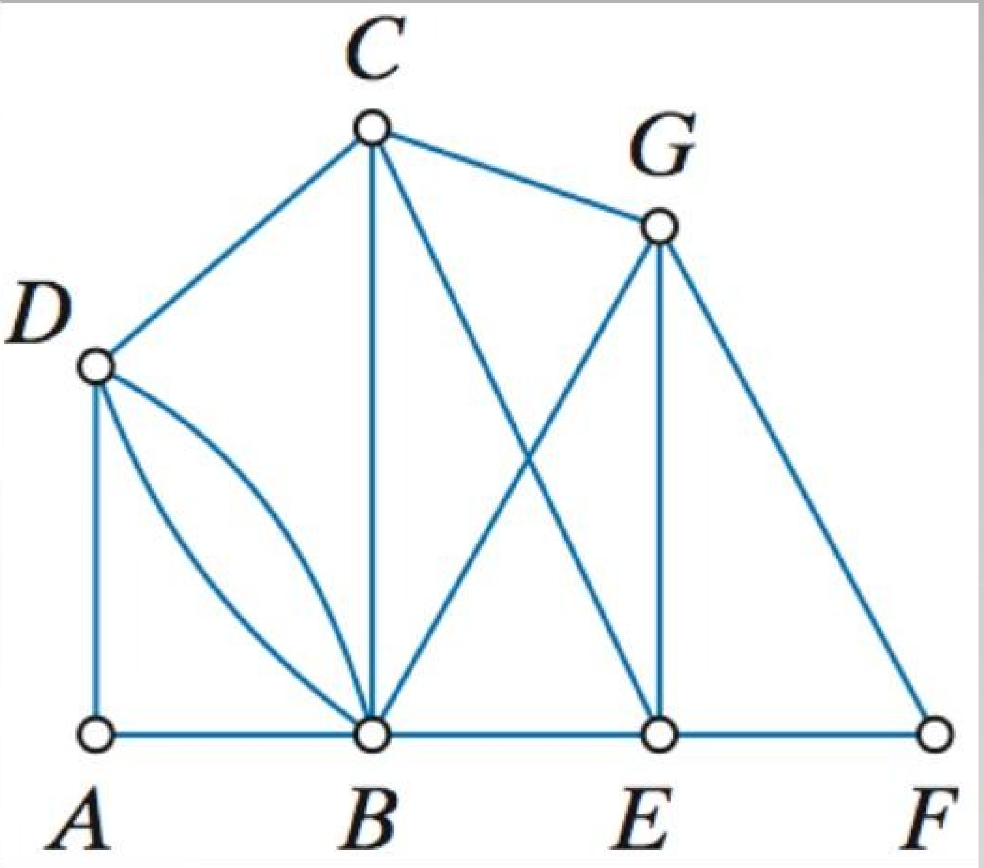


Illustration using the Theorem

This graph is connected and all the vertices are even.

The graph does have Euler circuits.



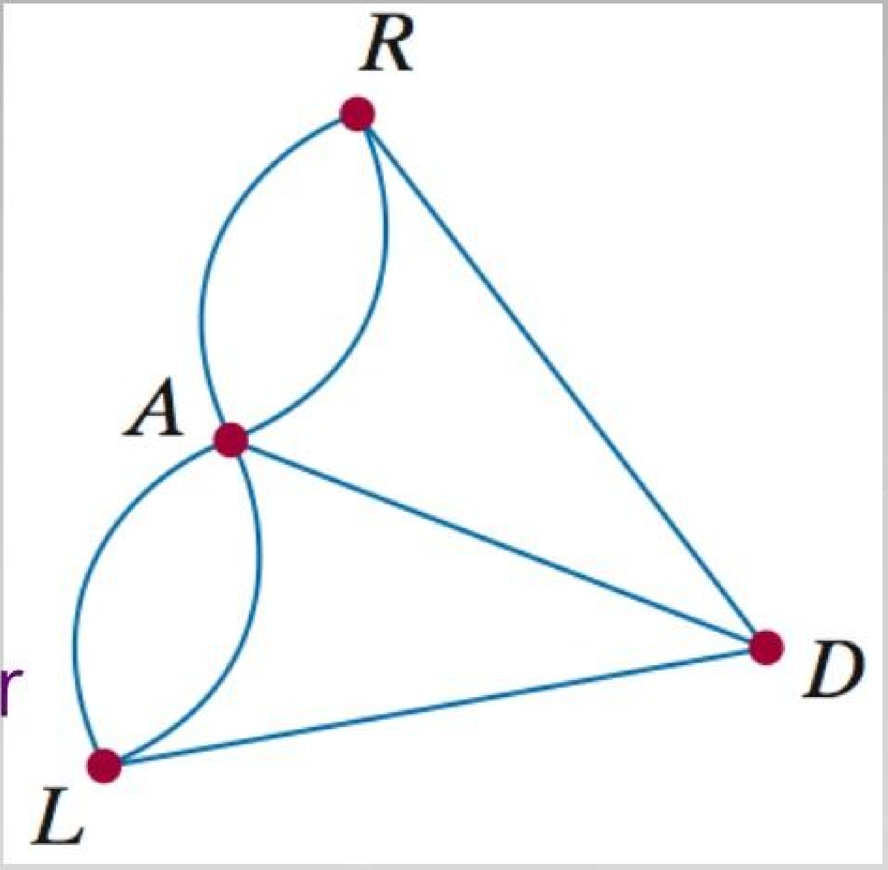
EULER'S PATH THEOREM

- If a graph is *connected* and has *exactly two odd vertices*, then it has an Euler path (at least one, usually more). Any such path must start at one of the odd vertices and end at the other one.
- If a graph has *more than two* odd vertices, then it cannot have an Euler path.

Example – The Seven Bridges of Königsberg: Part 3

In the Königsberg bridges problem, we saw that the layout of the bridges in the old city can be modeled by the graph.

This graph has four odd vertices; thus, it does not have an Euler circuit or an Euler path.



Example – The Seven Bridges of Königsberg: Part 3

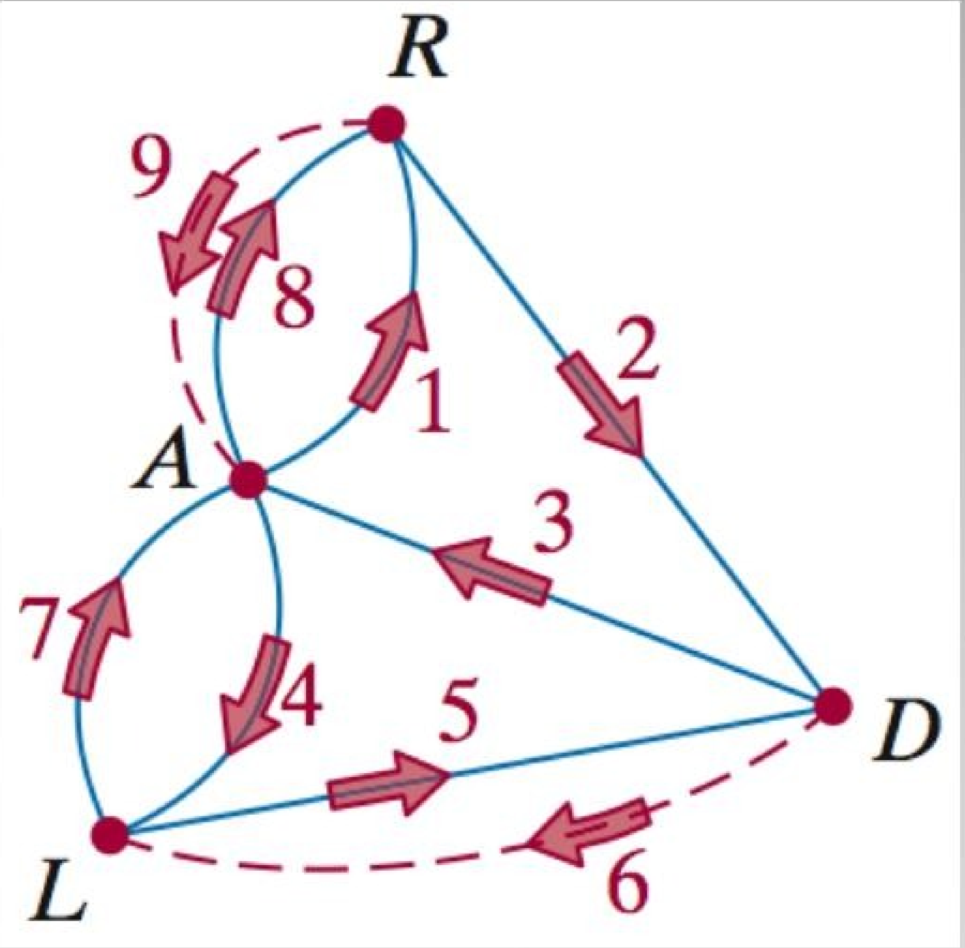
There is no possible way anyone can walk across all the bridges without having to re-cross some of them!

How many bridges will need to be re-crossed?

It depends. If we want to start and end in the same place, we must re-cross at least two of the bridges.

Example – The Seven Bridges of Königsberg: Part 3

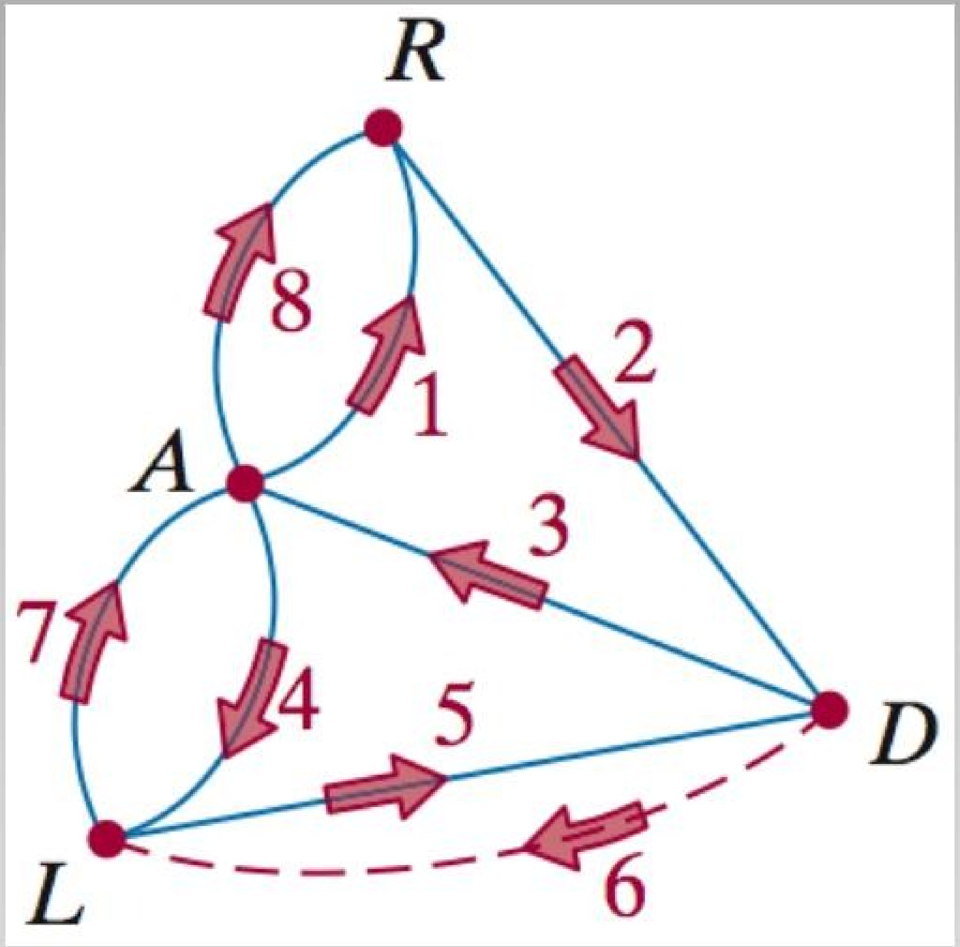
In this route the bridge connecting *L* and *D* is crossed twice, and so is one of the two bridges connecting *A* and *R*.



Example – The Seven Bridges of Königsberg: Part 3

If we are allowed to start and end in different places, we can do it by re-crossing just one of the bridges.

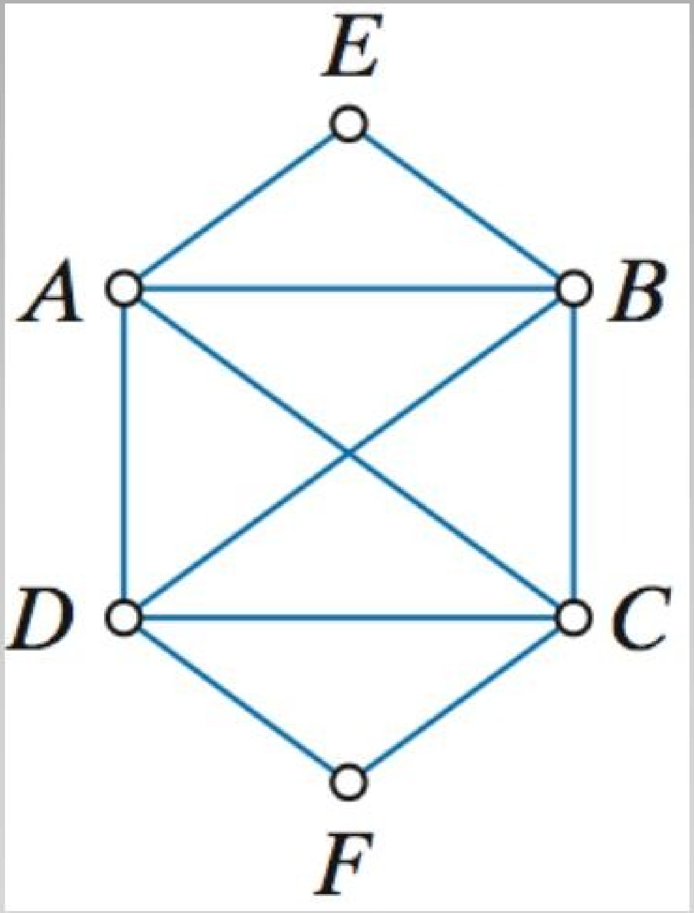
One possible route starting at *A*, crossing bridge *LD* twice, and ending at *R* is shown.



Example – Child's Play: Part 2

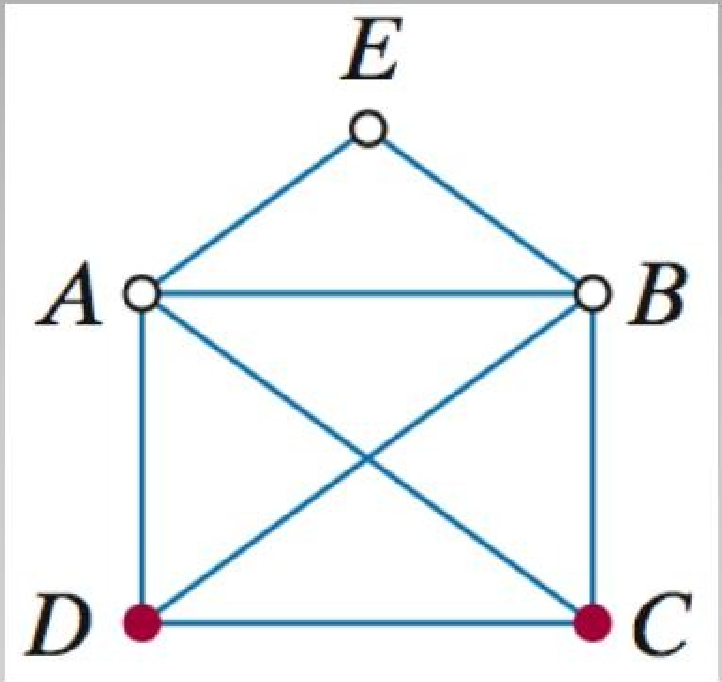
This graph is connected and the vertices are all even.

By Euler's circuit theorem we know that the graph has an Euler circuit, which implies that the original line drawing has a closed unicursal tracing.



Example – Child's Play: Part 2

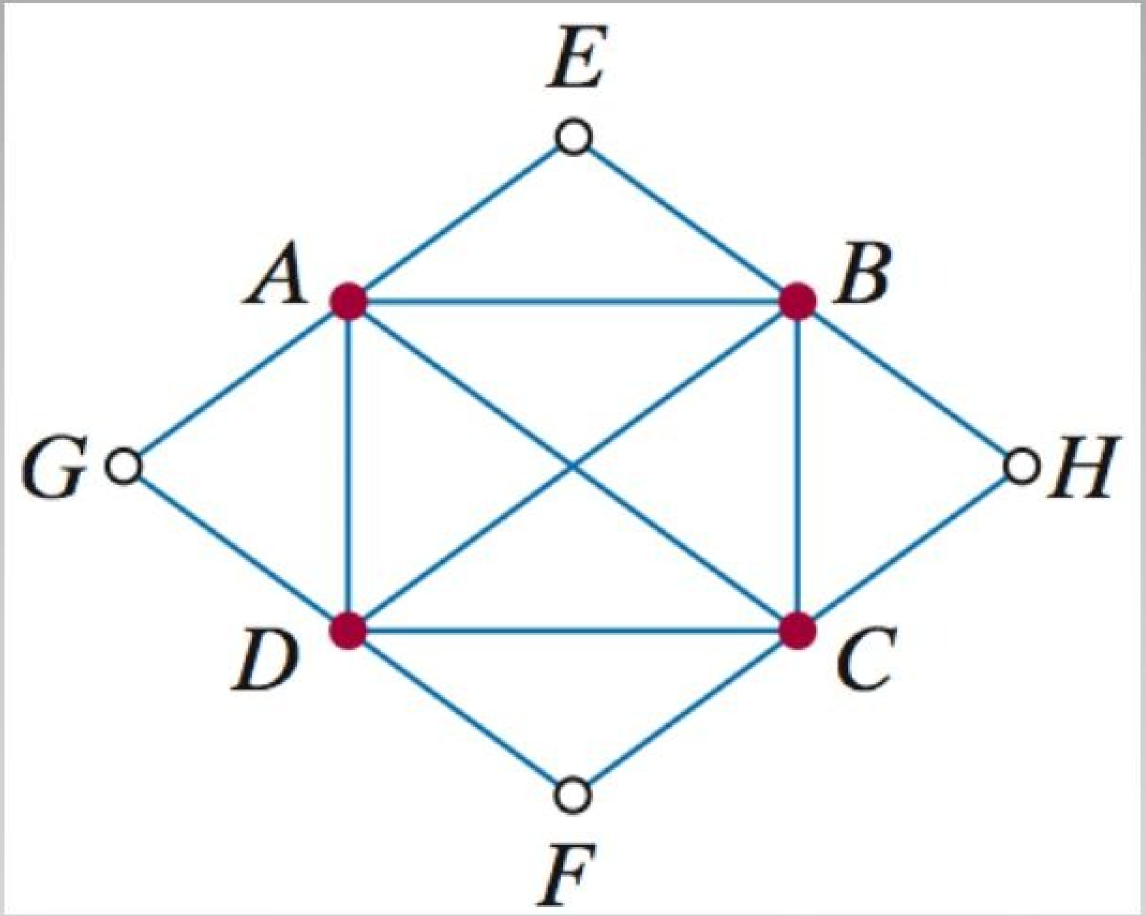
This graph is connected and has exactly two odd vertices (C and D). By Euler's path theorem, the graph has an Euler path (open unicursal tracing).



Moreover, we now know that the path has to start at C and end at D, or vice versa.

Example – Child's Play: Part 2

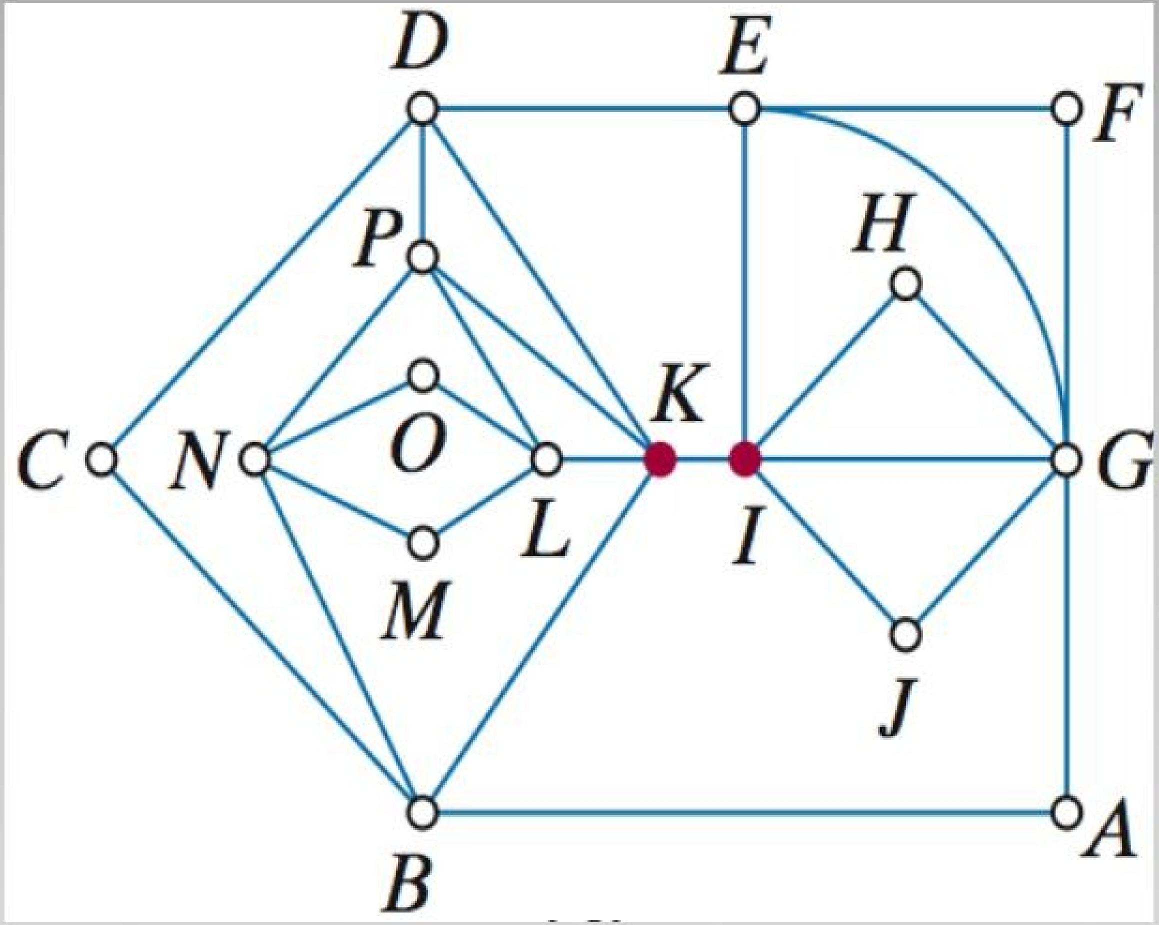
The graph has four odd vertices, so it does Not have an Euler path or an Euler circuit.



Example – Child's Play: Part 2

The full power of Euler's theorems is best appreciated when the graphs get bigger.

This graph is not extremely big, but we can no longer "see" an Euler circuit or path.



EULER'S SUM OF DEGREES THEOREM

- The sum of the degrees of all the vertices of a graph equals twice the number of edges (and therefore is an even number).
- A graph always has an even number of *odd* vertices.

Euler's Sum of Degrees Theorem

TABLE 5-1 Euler's Theorems (Summary)

Number of odd vertices	Conclusion
0	G has Euler circuit
2	G has Euler path
$4, 6, 8, \ldots$	G has neither
$1, 3, 5, \ldots$	Better go back and double check! This is impossible!

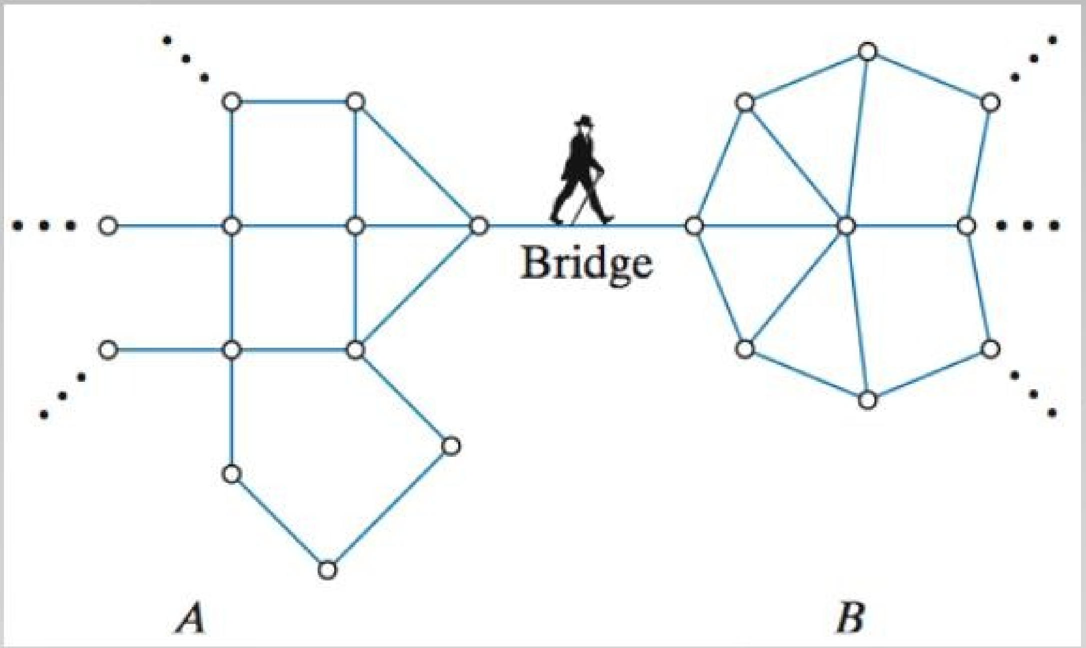
Fleury’s Algorithm

Fleury's Algorithm

The idea behind Fleury's algorithm can be paraphrased by that old piece of folk wisdom: *Don't burn your bridges behind you*.

Fleury's Algorithm

In graph theory the word bridge has a very specific meaning – it is the only edge connecting two separate sections (call them *A* and *B*) of a graph, as illustrated.



Fleury's Algorithm

Thus, Fleury's algorithm is based on a simple principle:

To find an Euler circuit or an Euler path, ***bridges are the last edges you want to cross.*** Our concerns lie only on how we are going to get around the *yet-to-be-traveled* part of the graph. Thus, when we talk about bridges that we want to leave as a last resort, we are really referring to *bridges of the to-be-traveled part of the graph*.

FLEURY'S ALGORITHM FOR FINDING AN EULER CIRCUIT (OR PATH)

Preliminaries.

* Make sure that the graph is connected and either has no odd vertices (circuit) or has just two odd vertices (path).

Start - Choose a starting vertex.

* In a circuit, this can be any vertex
* In a path, it must be one of the two odd vertices.

FLEURY'S ALGORITHM FOR FINDING AN EULER CIRCUIT (PATH)

Intermediate Steps.
At each step you have a choice, don't choose a bridge unless you have to.

End - When you can't travel any more, the circuit (path) is complete.

* In a circuit, you end back at the starting vertex

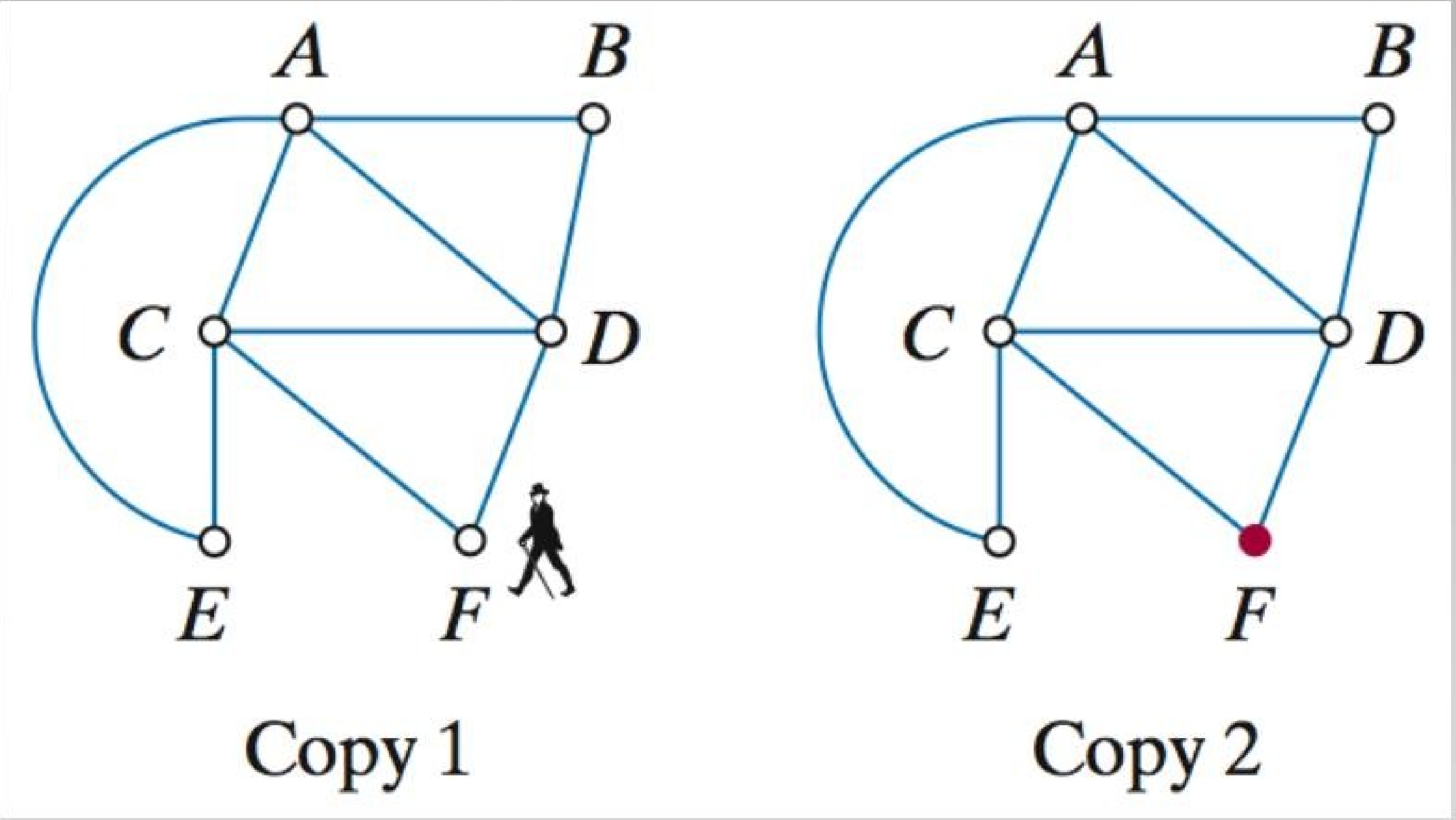
* Ina path, you end at the other odd vertex.

Fleury’s Algorithm Bookkeeping

- Every time you travel along an edge, erase the edge from copy 1, but mark it (say in red) and label it with the appropriate number on copy 2.
- As you move forward, copy 1 gets smaller and copy 2 gets redder.
- At the end, copy 1 has disappeared; copy 2 shows the actual Euler circuit or path.

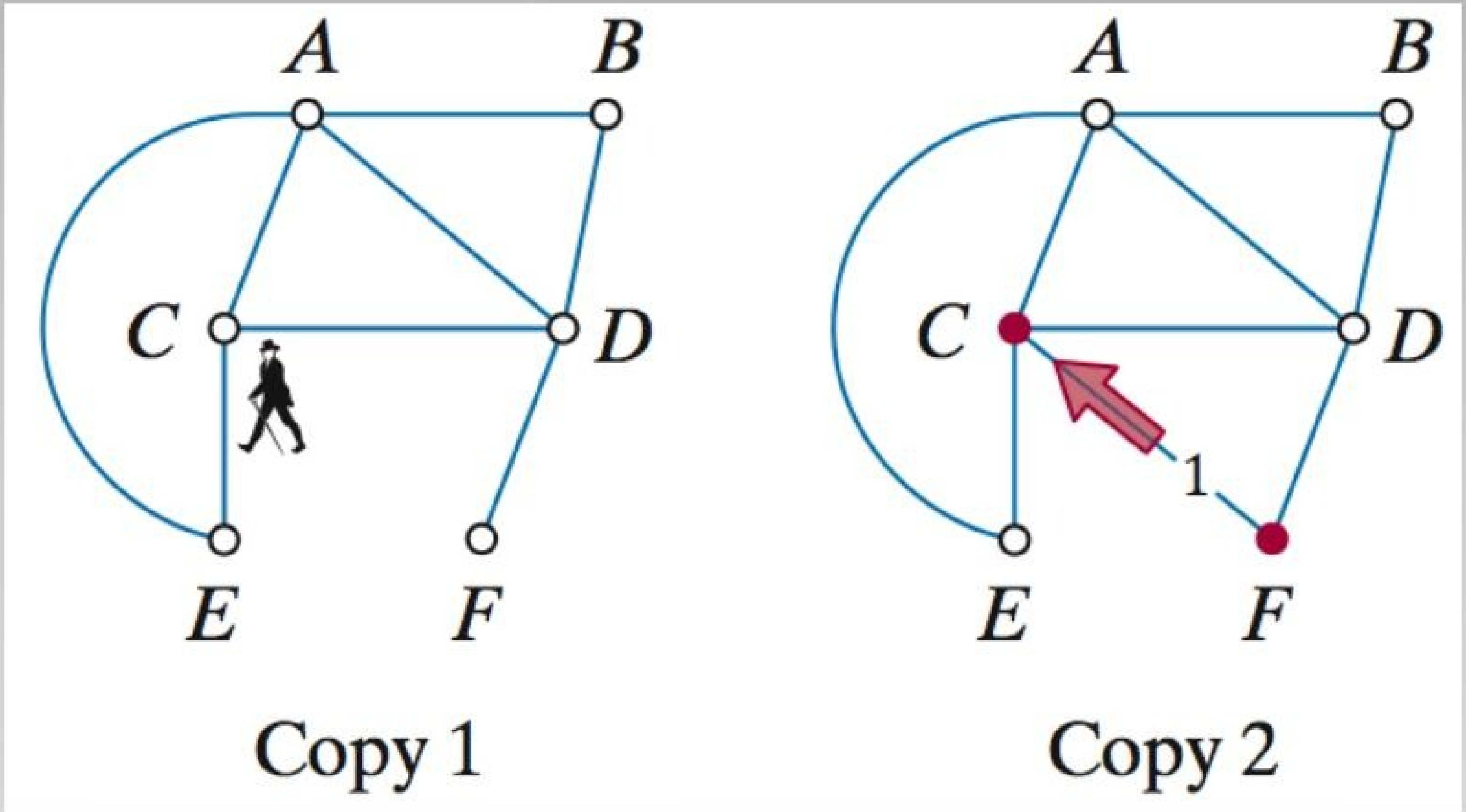
Example – Implementing Fleury's Algorithm

Start: We can pick any starting point we want. Let's say we start at *F*.



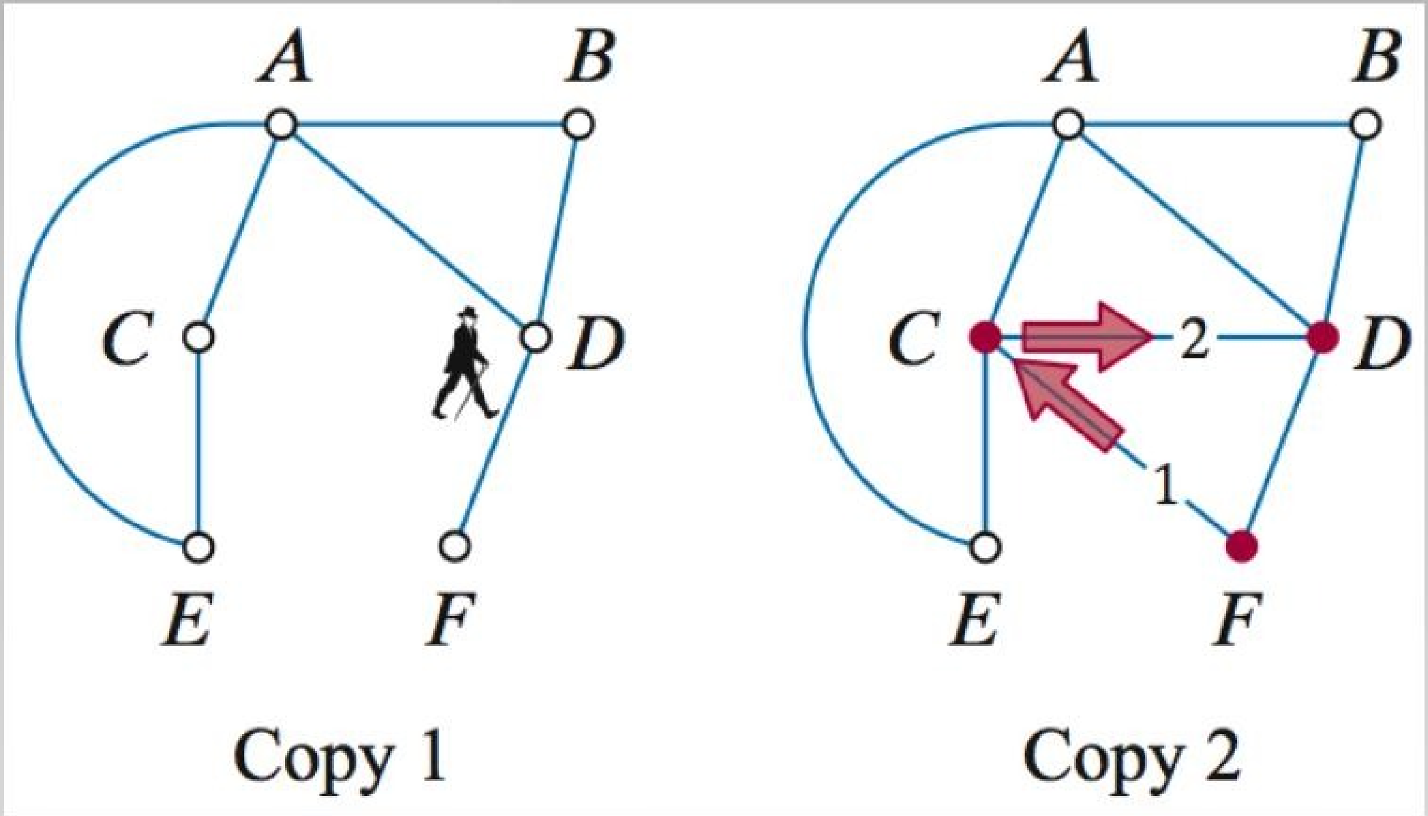
Example – Implementing Fleury's Algorithm

Step 1: Travel from *F* to *C*. (Could have also gone from *F* to *D*.)



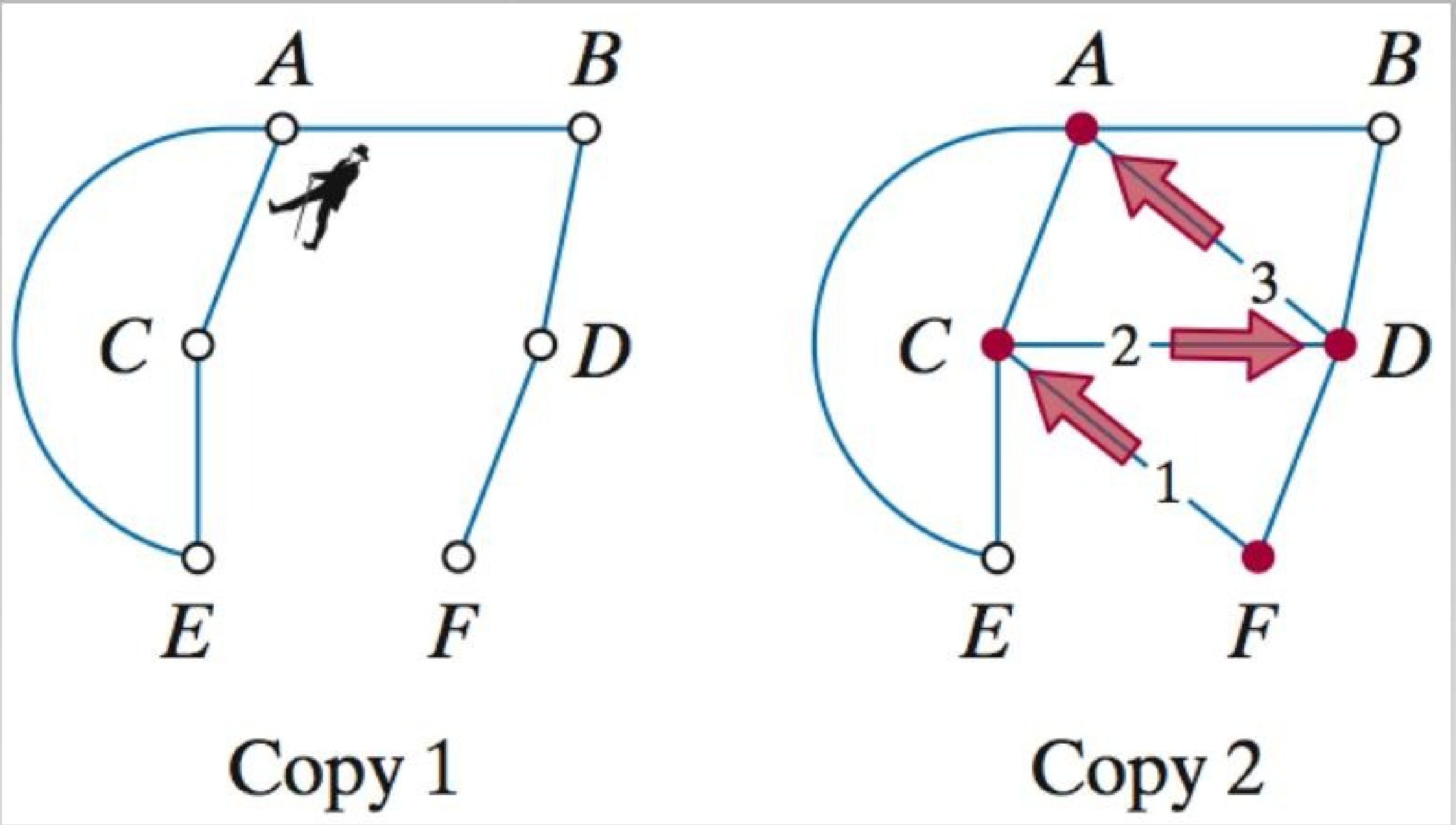
Example – Implementing Fleury's Algorithm

Step 2: Travel from C to *D*. (Could have also gone to *A* or to *E*.)



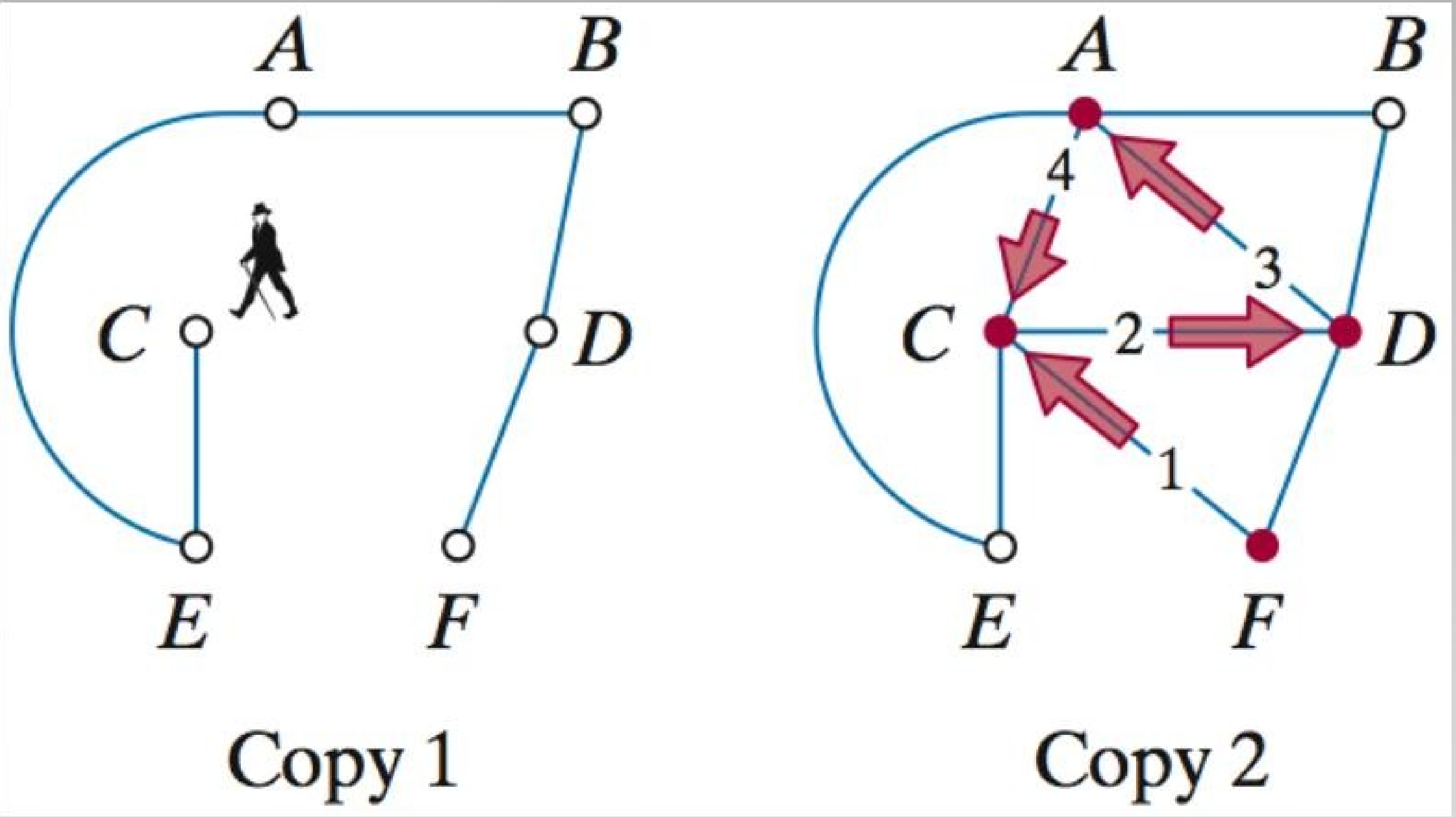
Example – Implementing Fleury's Algorithm

Step 3: Travel from *D* to *A*. (Could have also gone to *B* but not to *F* – *DF* is a bridge!)



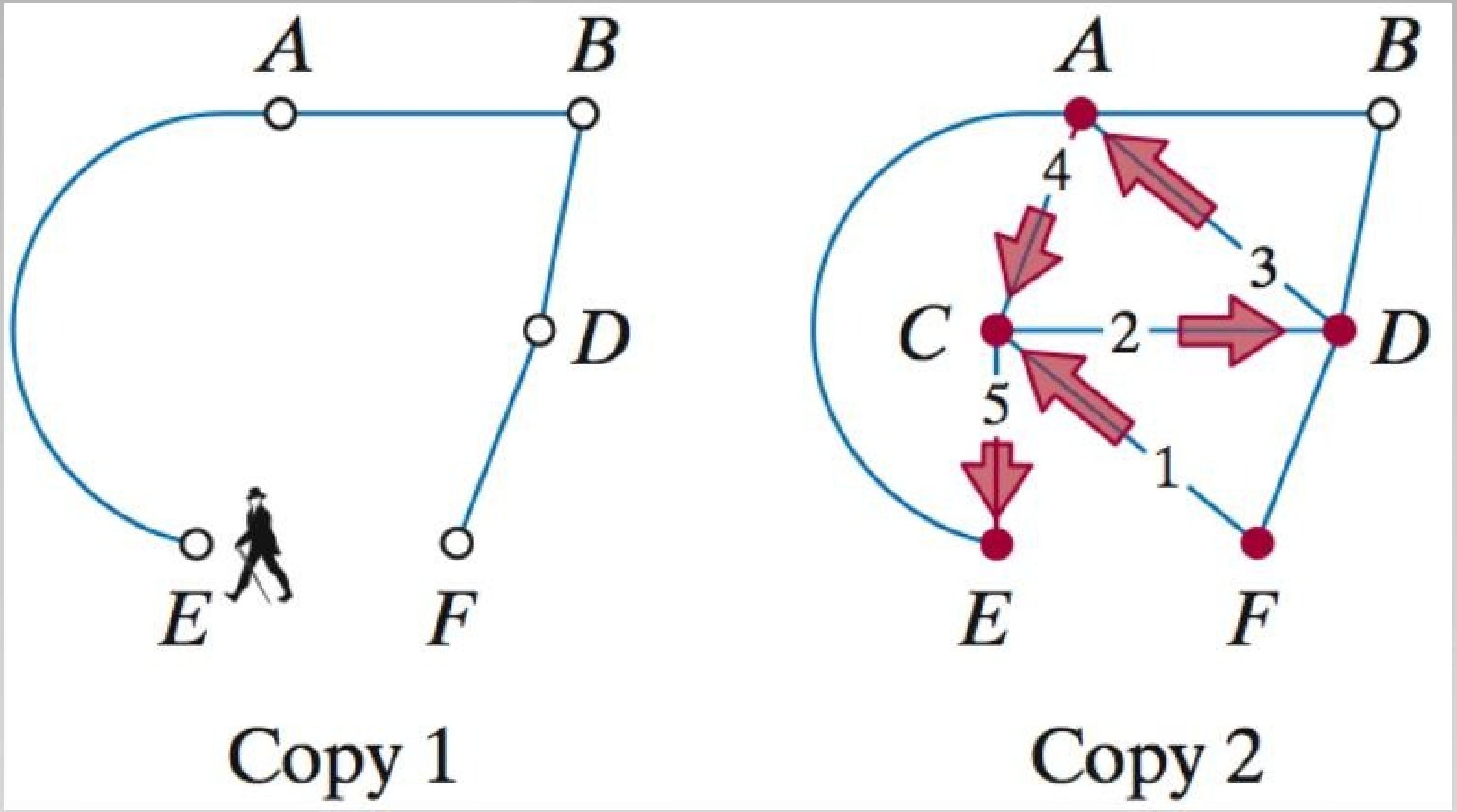
Example – Implementing Fleury's Algorithm

Step 4: Travel from *A* to *C*. (Could have also gone to *E* but not to *B* – *AB* is a bridge!)



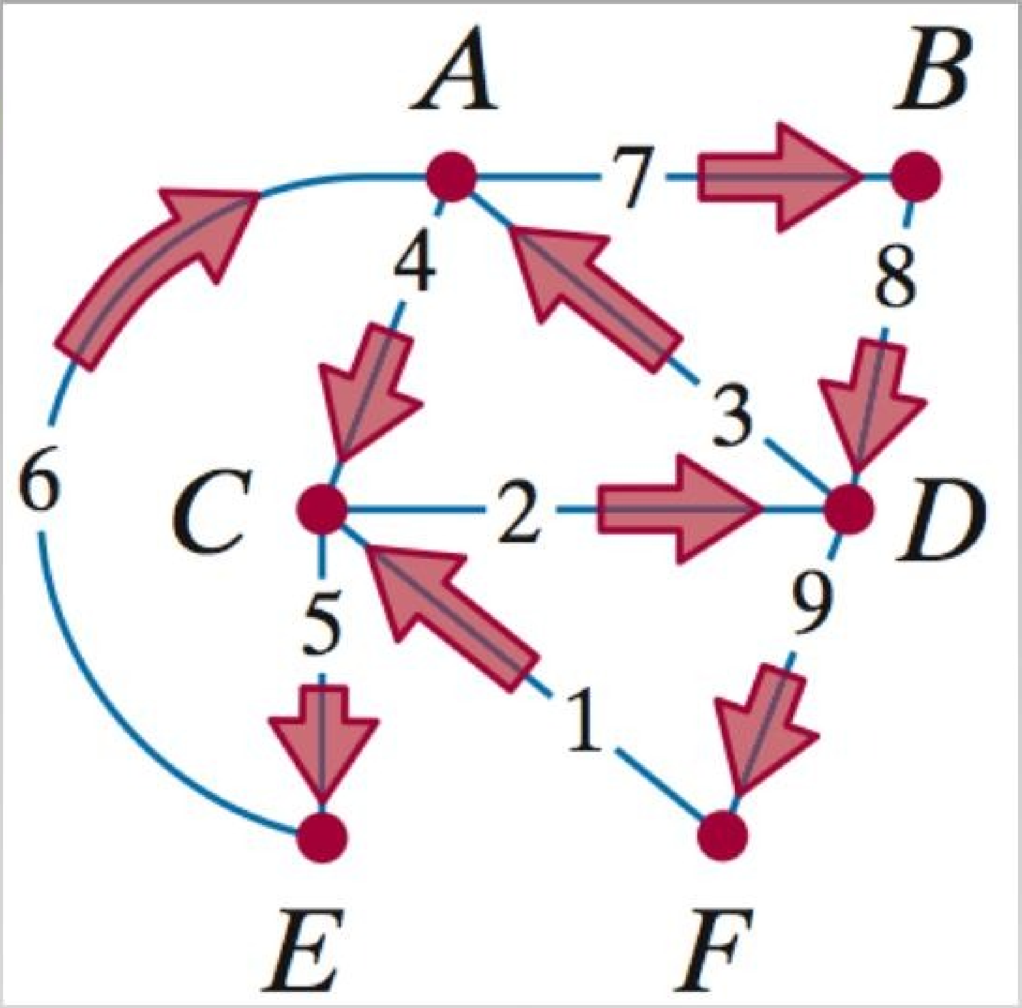
Example – Implementing Fleury's Algorithm

Step 5: Travel from C to *E*. (There is no choice!)



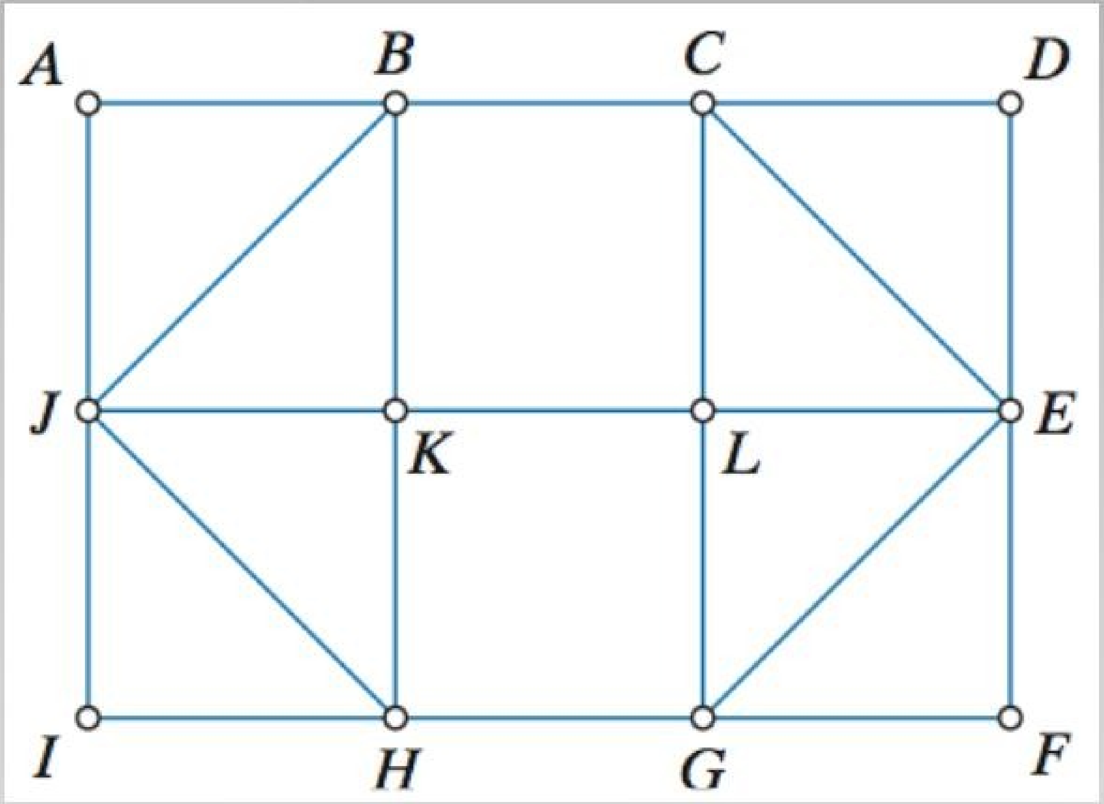
Example – Implementing Fleury's Algorithm

Steps 6, 7, 8, and 9:
Only one way to go at each step.



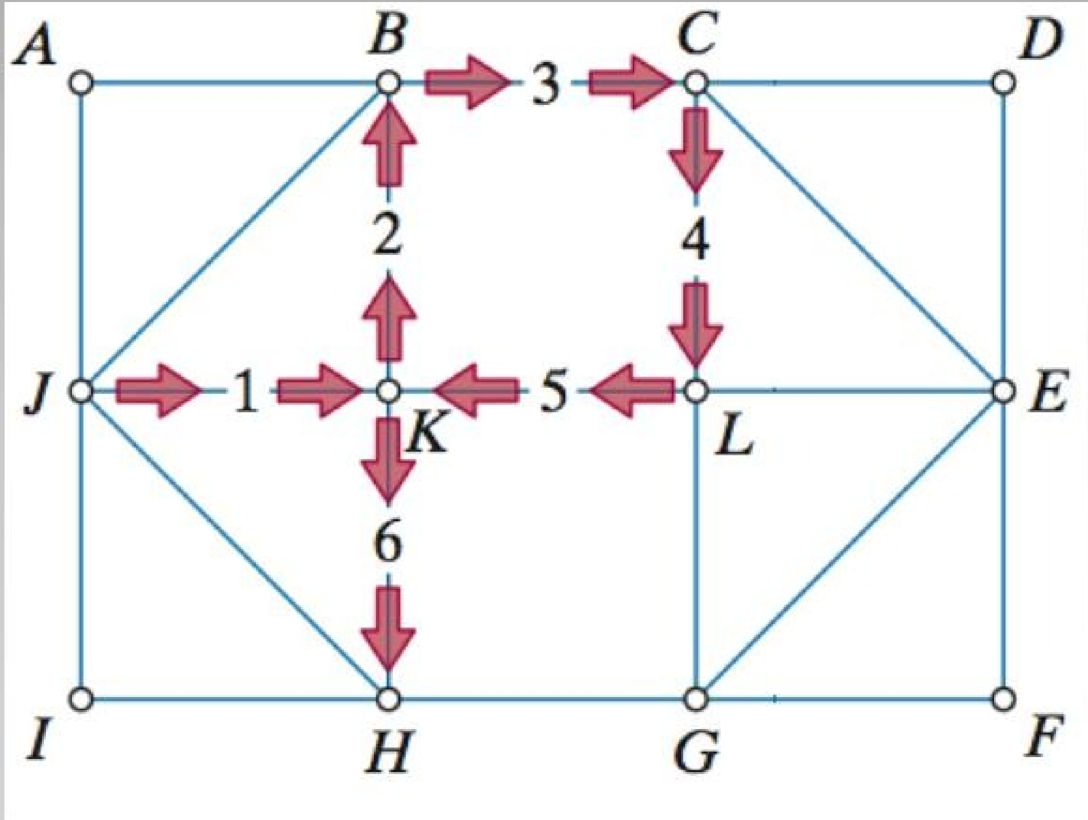
Example – Fleury's Algorithm for Euler Paths

We will apply Fleury's algorithm to the graph in Fig. 5-20.



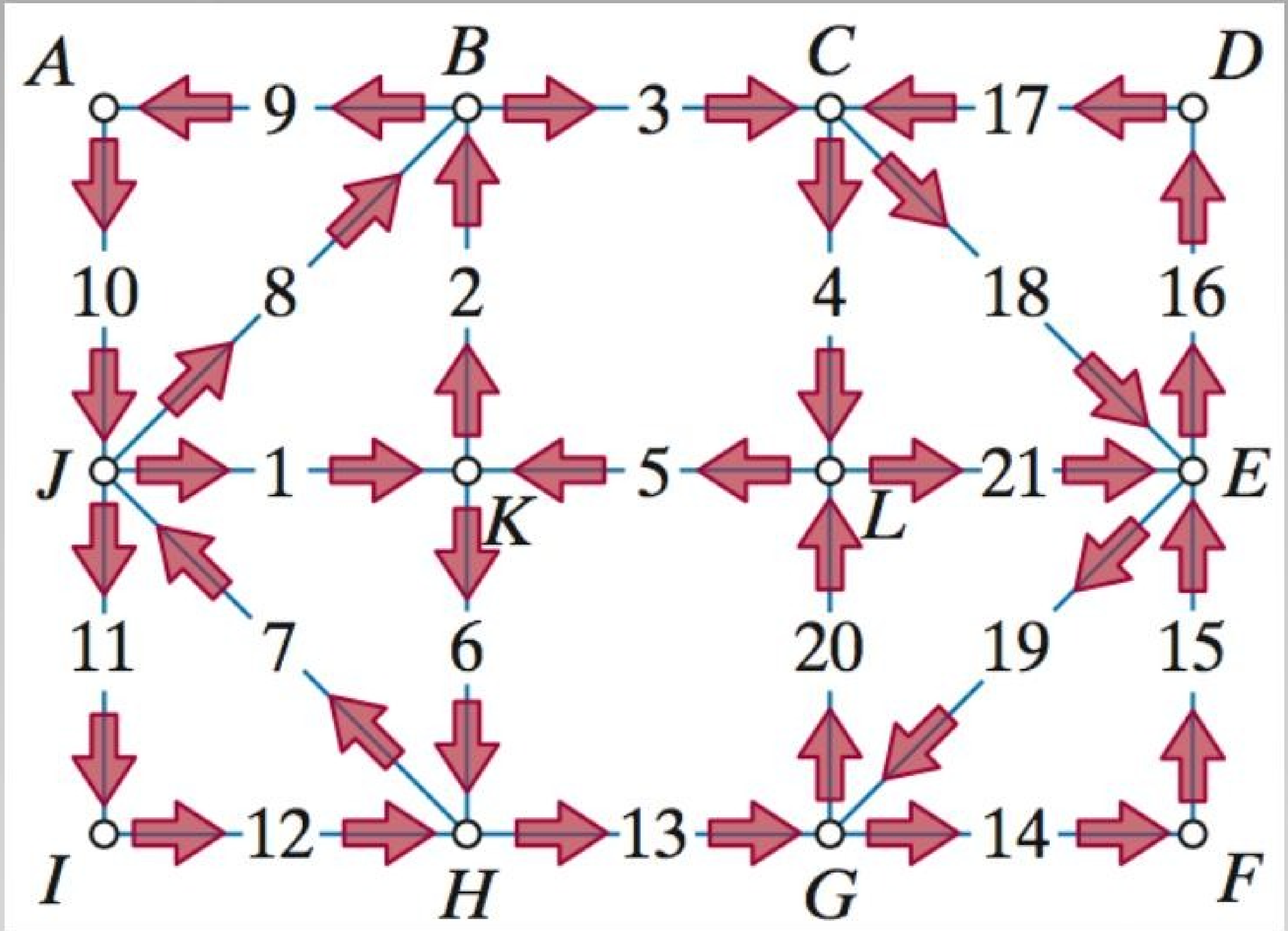
Example – Fleury's Algorithm for Euler Paths

Start at one of the odd vertices.



Copy 2 at Step 7

Example – Fleury's Algorithm for Euler Paths



Assignment

p. 193: 23, 24, 25, 29, 30, 31, 32, 35