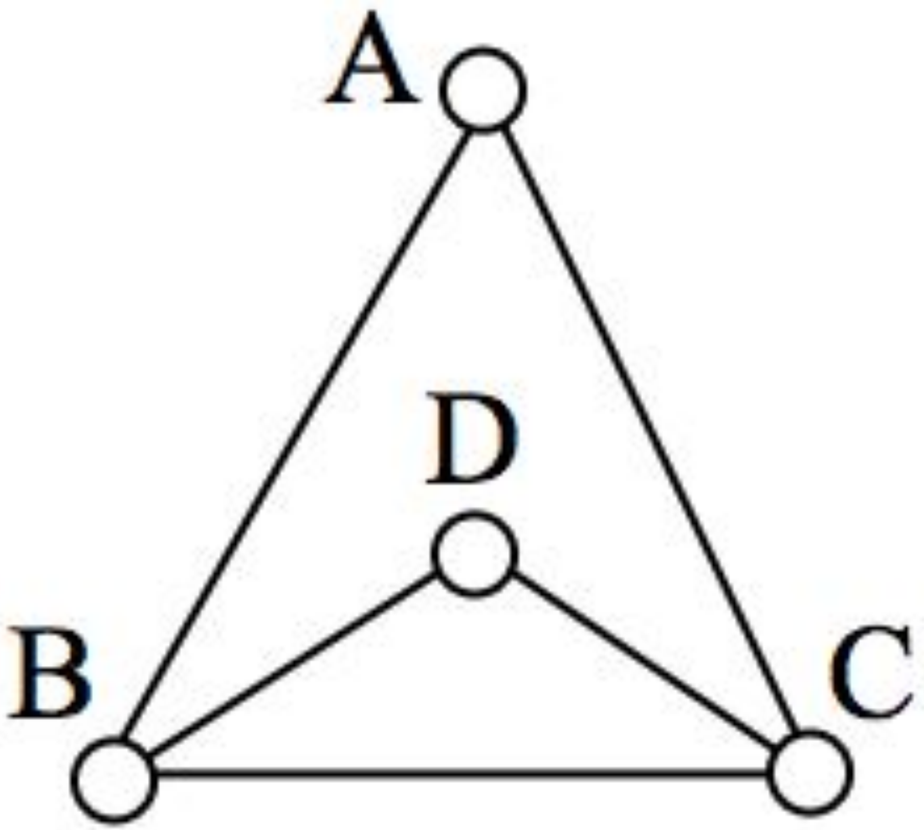
Euler path and Euler circuit(Fleury’s algorithm)

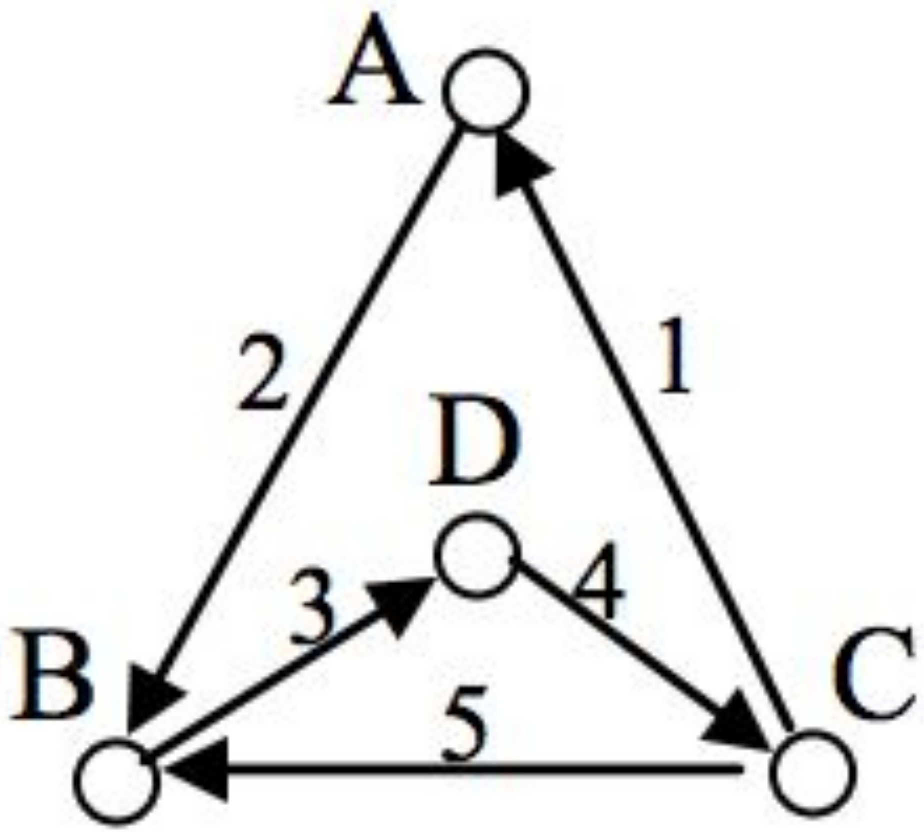
Rashad Mahmudov

EULER PATH

- An Euler path is a path that uses every edge in a graph with no repeats. Being a path, it does not have to return to the starting vertex.
- In the graph shown below, there are several Euler paths. One such path is CABDCB. The path is shown in arrows to the right, with the order of edges numbered.

EULER PATH EXAMPLE

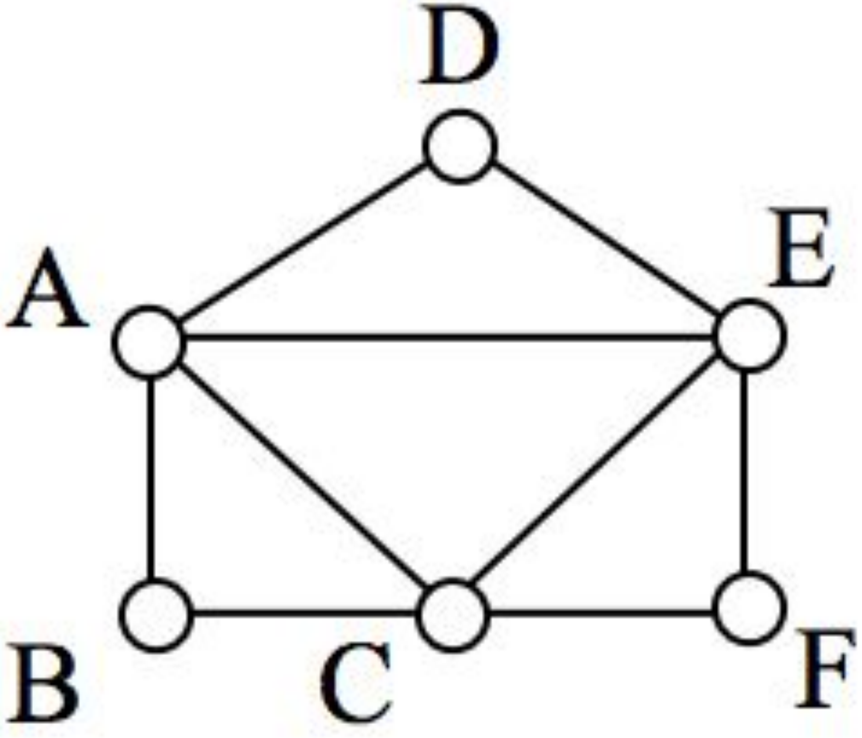


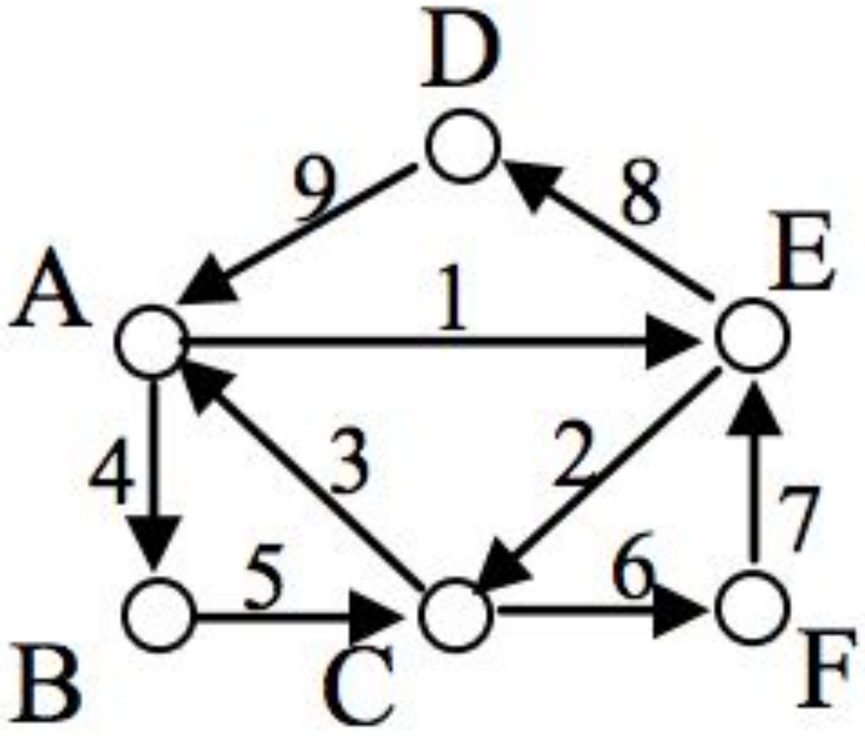


EULER CIRCUIT

- An Euler circuit is a closed path that uses every edge in a graph with no repeats. Being a closed path, it must start and end at the same vertex.
- The graph below has several possible Euler circuits. Here's a couple, starting and ending at vertex A: ADEACEFCBA and AECABCFEDA. The second is shown in arrows.

EULER CIRCUIT EXAMPLE



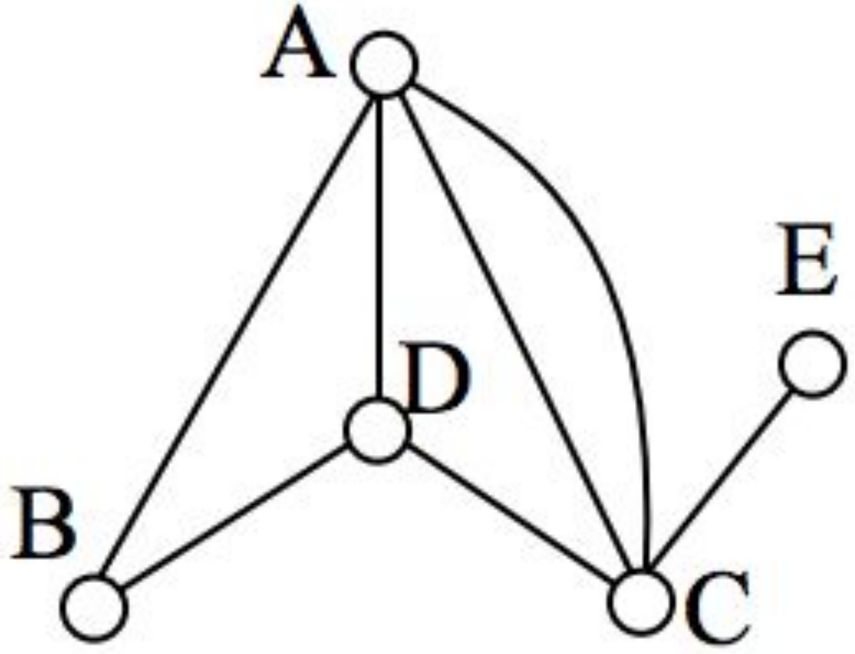


EULER'S PATH AND CIRCUIT THEOREMS

- A graph will contain an Euler path if and only if it contains at most two vertices of odd degree.
- A graph will contain an Euler circuit if and only if all vertices have even degree

EULER'S PATH AND CIRCUIT THEOREMS
EXAMPLE

- In the graph below, vertices A and C have degree 4, since there are 4 edges leading into each vertex. B is degree 2, D is degree 3, and E is degree 1. This graph contains two vertices with odd degree (D and E) and three vertices with even degree (A, B, and C), so Euler's theorems tell us this graph has an Euler path, but not an Euler circuit.

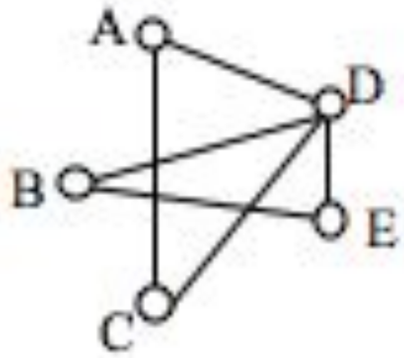


FLEURY’S ALGORITHM

- 1. Start at any vertex if finding an Euler circuit. If finding an Euler path, start at one of the two vertices with odd degree.
- 2. Choose any edge leaving your current vertex, provided deleting that edge will not separate the graph into two disconnected sets of edges.
- 3. Add that edge to your circuit, and delete it from the graph.
- 4. Continue until you’re done.
- Find an Euler Circuit on this graph using Fleury’s algorithm, starting at vertex A.

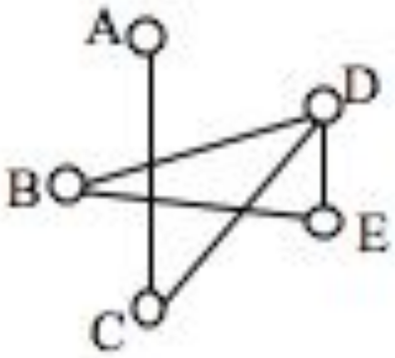
FLEURY'S ALGORITHM EXAMPLE

Original Graph.
Choosing edge AD.



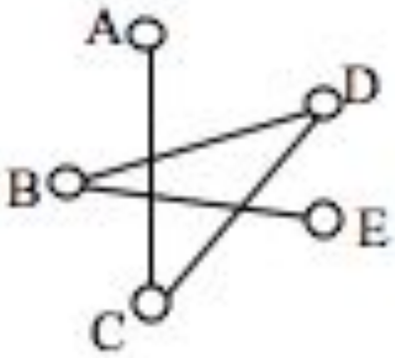
Circuit so far: AD

AD deleted. D is current.
Can't choose DC since that would disconnect graph.
Choosing DE



Circuit so far: ADE

E is current.
From here, there is only one option, so the rest of the circuit is determined.



Circuit: ADEBDCA

FLEURY'S ALGORITHM PSEUDOCODE

procedure *Fleury*(connected directed multigraph G with vertices $v_1, v_2, ...v_n$ and every vertex has an even degree)

H:=G

circuit:=v_1

initial:=v_1

while H has edges

  edge:=an edge with *initial* as initial vertex that is not a cut edge (unless we don't have any other choice).

  initial:=terminal vertex of *edge*

  circuit:=*circuit*, *initial* (concatenation)

  H:=H with *edge* and all isolates vertices removed.

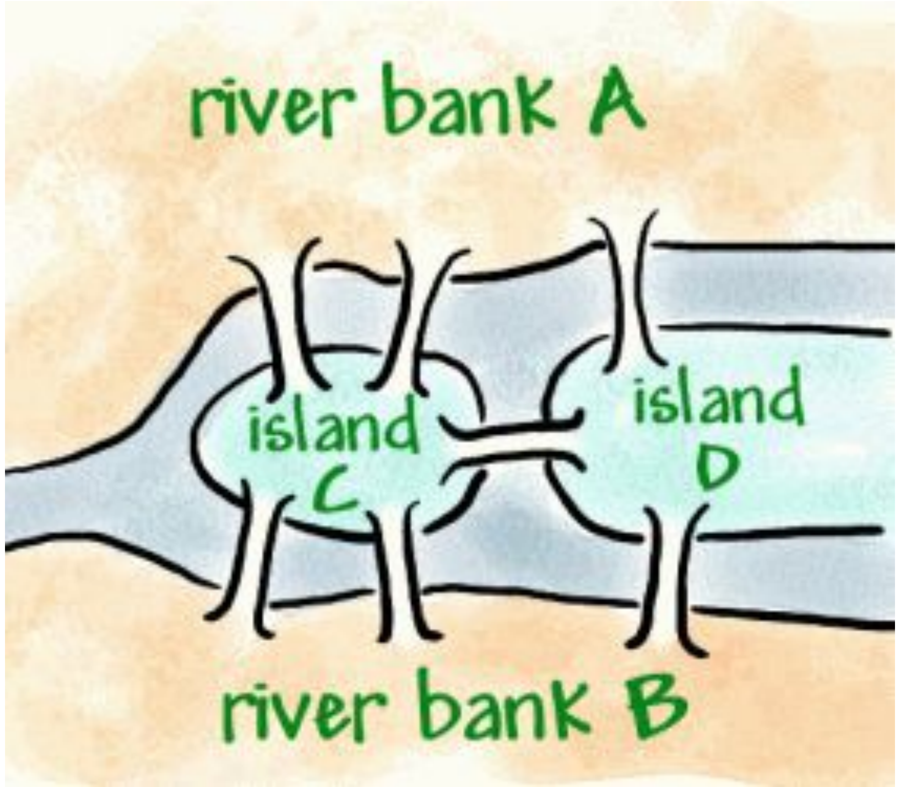
return *circuit*

Time Complexity

- Worst case time complexity: **Θ**((V+E)^2)
- Average case time complexity: **Θ**((V+E)^2)
- Best case time complexity: **Θ**((V+E)^2)
- Space complexity: **Θ**(V^2)

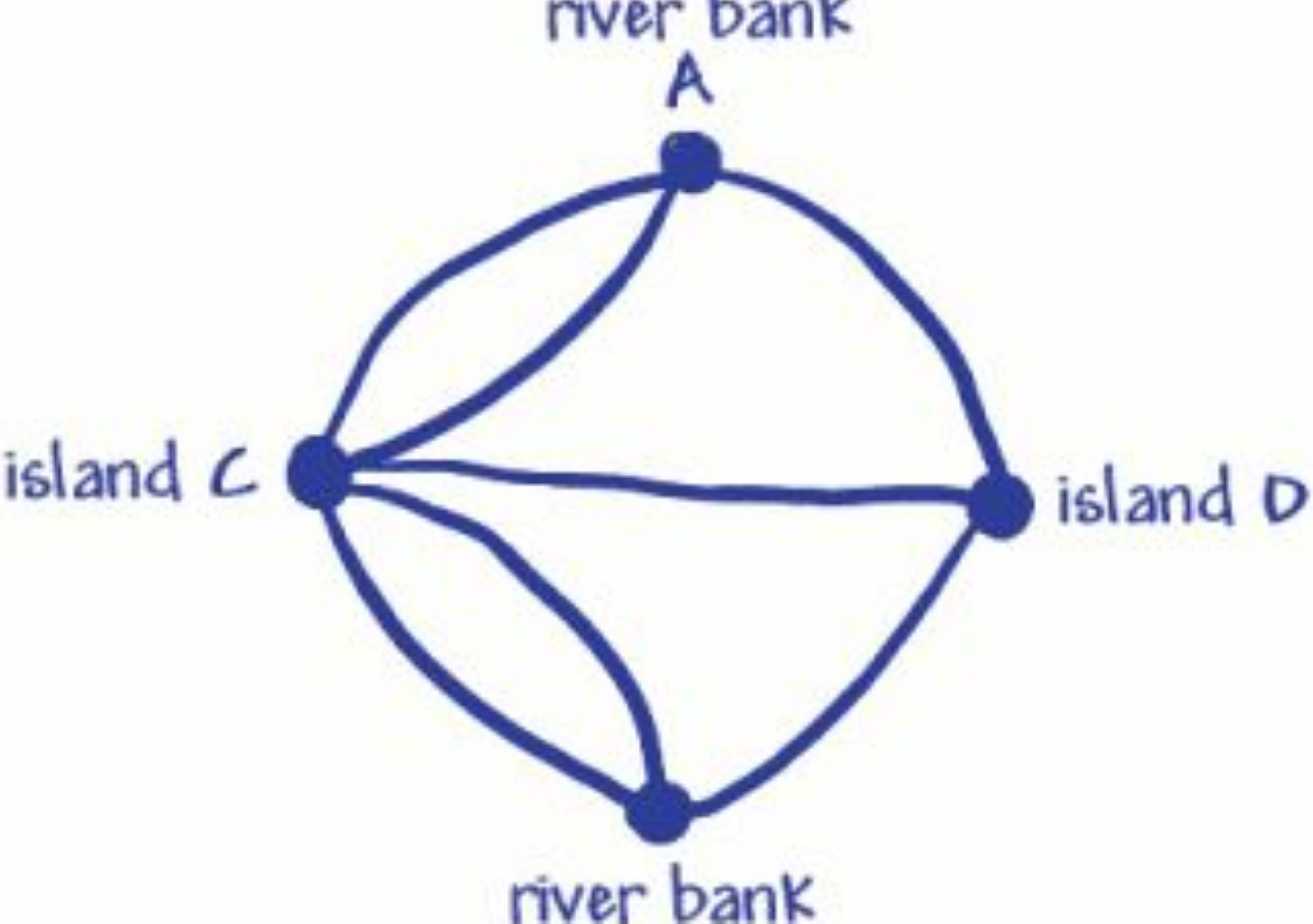
The Königsberg Bridge Problem

- Konigsberg is a town on the Preger River, which in the 18th century was a German town, but now is Russian. Within the town are two river islands that are connected to the banks with seven bridges.



The Königsberg Bridge Problem

- It became a tradition to try to walk around the town in a way that only crossed each bridge once, but it proved to be a difficult problem. Leonhard Euler, a Swiss mathematician in the service of the Russian empress Catherine the Great, heard about the problem. In 1736 Euler proved that the walk was not possible to do. He proved this by inventing a kind of diagram called a network, that is made up of vertices (dots where lines meet) and arcs (lines).


river bank
A
island C
island D
river bank

The Königsberg Bridge Problem

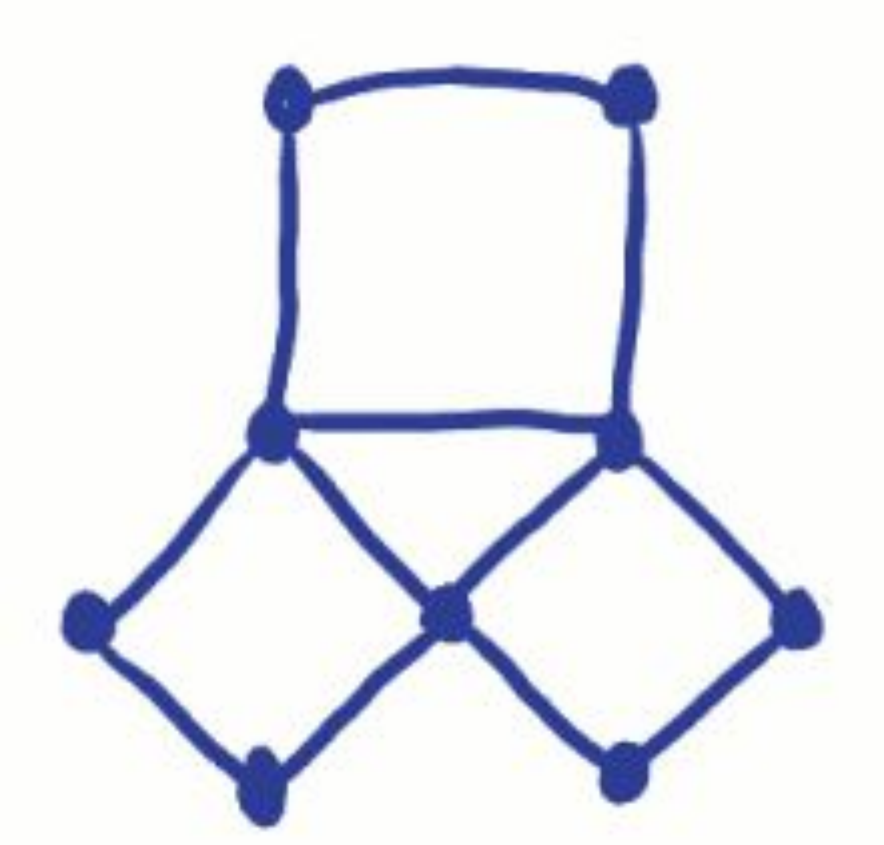
- He used four dots (vertices) for the two riverbanks and the two islands. These have been marked A, B and C, D. The seven lines (arcs) are the seven bridges. You can see that 3 bridges (arcs) join to riverbank A, and 3 join to riverbank B. 5 bridges (arcs) join to island C, and 3 join to island D. This means that all the vertices have an odd number of arcs, so they are called odd vertices. (An even vertex would have to have an even number of arcs joining to it).

The Königsberg Bridge Problem

- Remember that the problem was to travel around town crossing each bridge only once. On Euler's network this meant tracing over each arc only once, visiting all the vertices. Euler proved it couldn't be done because he worked out that to have an odd vertex you would have to begin or end the trip at that vertex. (Think about it). Since there can only be one beginning and one end, there can only be two odd vertices if you're going to be able to trace over each arc only once. Since the bridge problem has 4 odd vertices, it just isn't possible to do! What happens if there are no odd vertices at all? Can this network be traced?

The Königsberg Bridge Problem

- The invention of networks began a whole new type of geometry called Topology. Topology is now used in many ways, including for planning and mapping railway networks.
- https://www.youtube.com/watch?v=nZwSo4vfw6c



Thank you for your attention