

## WORKSHEET 14

# Eulerian Paths and Circuits

A *graph* is a collection of dots, or *vertices*, with curves, or *edges*, connecting pairs of dots. A typical example is shown in Figure 1.

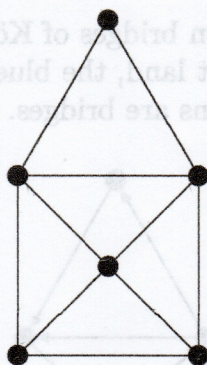


Figure 1. A graph.

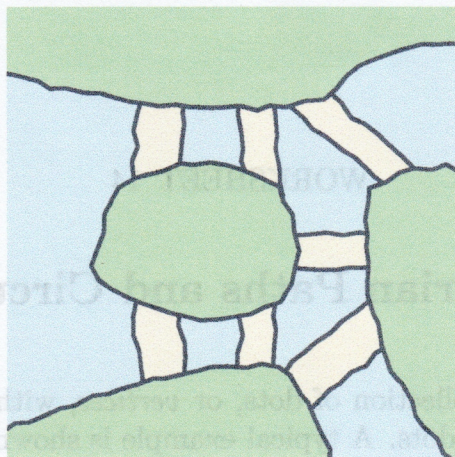
Given a graph, it is natural to ask if it is possible to start at one of the vertices and travel along each edge exactly once. Such a path is called an *Eulerian path*. If this path also ends up at the starting vertex, it is called an *Eulerian circuit*.

PROBLEM 14.1. Does the graph shown in Figure 1 have an Eulerian path? What about an Eulerian circuit?

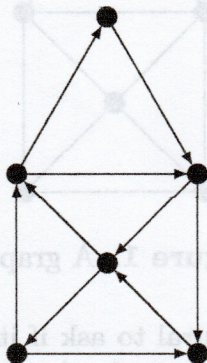
PROBLEM 14.2. Draw several other graphs with a small number of vertices, checking in each one whether it has an Eulerian path, an Eulerian circuit, or neither.

PROBLEM 14.3. Can you find a criterion for when a graph has an Eulerian path or an Eulerian circuit? Make sure to stress test your criterion by challenging it on examples where you think it is most likely to fail. This will help you to refine your criterion until it is exactly right.

In order to answer questions like this, it is helpful to be aware of the two types of criteria: *necessary* and *sufficient* conditions. Let  $P$  be a property that a graph might or might not have. (For instance,  $P$  could be the property that the graph has exactly 8 vertices, or that it has a vertex connected to more than half of the other vertices.) We



**Figure 2.** The seven bridges of Königsberg. The green areas represent land, the blue represents water, and the yellow regions are bridges.

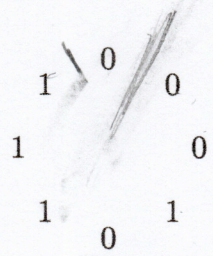


**Figure 3.** A directed graph.

say that  $P$  is a *necessary condition* for having an Eulerian path (say) if every graph with an Eulerian path has property  $P$ , and we say that  $P$  is a *sufficient condition* for having an Eulerian path if every graph with property  $P$  has an Eulerian path. The ideal situation is that we can find a condition that is both necessary and sufficient. In the case of both Eulerian paths and Eulerian circuits, such a necessary and sufficient condition exists.

**PROBLEM 14.4.** The seven bridges of Königsberg, shown in Figure 2 is a famous problem solved by Euler. Is it possible to travel over each bridge exactly once? Is it possible to do so while returning to the same region?

A *directed graph* is similar to a graph, except that each edge has an arrow on it, and it is only possible to travel in the direction of the arrow. See Figure 3 for an example.



**Figure 4.** A de Bruijn sequence with  $b = 2$  and  $n = 3$ .

**PROBLEM 14.5.** Find a necessary and sufficient condition for when a directed graph has an Eulerian path or an Eulerian circuit.

**PROBLEM 14.6.** By now, you should know when an Eulerian path or circuit exists in a (possibly directed) graph. But can you actually find it? Can you come up with an algorithm that will quickly find an Eulerian path or circuit when one exists? (There are several good answers to this problem.)

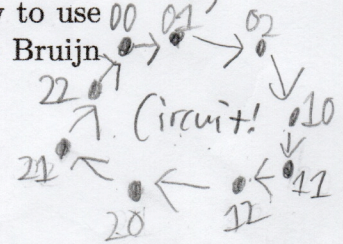
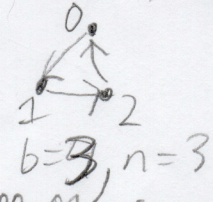
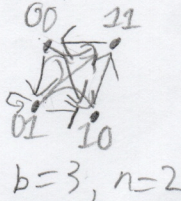
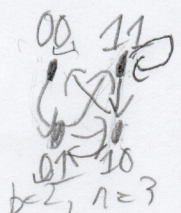
**PROBLEM 14.7.** If  $G$  is an (undirected) graph, let  $2 \cdot G$  be the graph with the same vertices as  $G$ , but with two edges for each edge of  $G$ , connecting the same pair of vertices. For which graphs does  $2 \cdot G$  have an Eulerian path or circuit?

**PROBLEM 14.8.** Suppose a graph has an Eulerian path that isn't an Eulerian circuit. Can it also have an Eulerian circuit? Could it have another Eulerian path with different starting and ending vertices?

**PROBLEM 14.9.** Let  $b$  and  $n$  be positive integers. The *de Bruijn graph*  $DB(b, n)$  has  $b^{n-1}$  vertices, which are indexed by sequences of length  $n - 1$ , each of which is an integer from 0 to  $b - 1$ , and we place a directed edge from a vertex  $(s_1, s_2, \dots, s_{n-1})$  to  $(s_2, s_3, \dots, s_{n-1}, d)$  for each  $d$  with  $0 \leq d \leq b - 1$ . Show that the de Bruijn graph has an Eulerian circuit.

**PROBLEM 14.10.** A *de Bruijn sequence* of window length  $n$  is a way of arranging  $b^n$  numbers around a circle, each of which is an integer from 0 to  $b - 1$  (each used  $b^{n-1}$  times), in such a way that every window of  $n$  consecutive numbers is different from every other window. (See Figure 4 for an example when  $b = 2$  and  $n = 3$ .) Explain how to use an Eulerian circuit on the de Bruijn graph to construct a de Bruijn sequence. *n=3 Try it*

~~00101110~~  
~~0010110~~

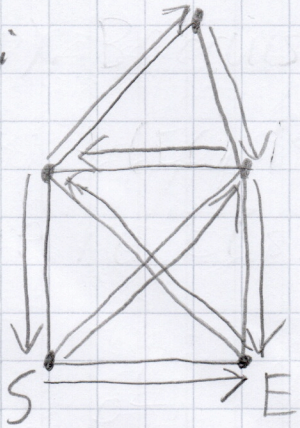


00010111  
00010110  
0010110

\* an odd number of + for an EC, = 0 odd points.  
Eulerian Paths and Circuits  
 Alexander Friesen 2/3-8/26

14.1: Here is an example of a Eulerian Path (EP):

However, there is no Eulerian Circuit (EC)



14.2: that fits this shape. There are more EPs.

14.2: 1 point: •, Always possible.

2 points: ↗ or ••, Possible w/line.

3 points: △, Possible.

4 points: □, ▢, ▣, P, P, I<sup>(impossible)</sup>.

5 points: ⬠, ⬡, ⬢, ⬣, ..., P, I, P, I...

6 points: ⬤, ⬥, ⬦, ..., P, P, P...

14.3: I noticed (in 14.2) that each graph is only possible when:

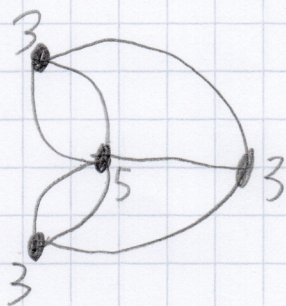
- 0 points have odd\* connections, and the path is an EC
- 2 points are "odd", and the path is not an EC.

Necessary: Graph has 0, 2 odd points ≠

Sufficient: Graph has 0, or 2 odd points

\*For each node, of course.

14.4: Königsberg also looks like:



In this graph, every point is "odd," so an EP is impossible.

14.5: In a directed graph (DF), the "in" and "out" vertices must be measured separate from each other.\*

So, an EP will always enter a vertice the same number of times it exits a vertice. This means that for an EP to be successful, the "in" edges and the number of "out" edges must be equal except for on the start and end points. On the start/end, the difference between in/out is allowed to be  $+1$  or  $-1$  to either side to account for EP. However, in an EC every node has "in" = "out".

\*Ending there doesn't count.

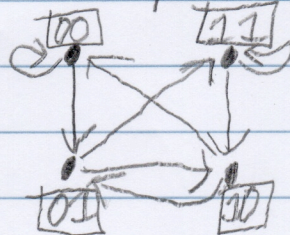
14.6: If you know that an EP exists, try to find a point with an odd number of connections, and start there. If you can't, then the graph is an EC, in which you draw the biggest loop that goes through every point once and the start/end twice. If you can, then find the second odd node, the end. Create the largest loop possible from the start that satisfies the Euler path conditions, making sure to only pass through the end point once.\* If you can't form one loop from start to end, then pick an even point in your loop and start a loop from there, making sure to pass through every remaining edge. Repeat this until the loop created finishes at the odd "end" point.

14.7: In this case, because the connections of each node are doubled, every node always has an even parity. Therefore, an EC will always be present in any  $\mathbb{F}_2$  graph.

14.8: If a graph has an  $EP \neq EC$ , then there must be 2 nodes that have odd connections, otherwise the start and end would be the same node. Therefore, if a graph only has an EP, it cannot ever have an EC also because of the odd nodes. However, if a graph has an EC then it also has an EP. If a graph has an  $EP \neq EC$ , then the only other EP is a reversed direction of travel, not from different start/end nodes - otherwise more than 2 odd nodes would exist.

14.9: In a de Bruijn graph, the set of possible characters for each node is of length  $b$ . This means that each node has  $b^{n-2}$  possible ending strings, and likewise for starting strings (where the directed edge that matches goes to.) This creates  $b^{n-2}$  "out" and  $b^{n-2}$  "in" directed edges for every node on the DB graph. Because  $\text{in} = \text{out} = b^{n-2}$  edges for each node, every DB graph has a directed EC.

14.10: Instead of a circle, let's write out the DB sequence: 00010111... (0001...) The corresponding  $b=2/n=3$  graph looks like:



By "walking" the EC, starting from 00, the DB sequence forms:  $\boxed{00}$  -----,  
 0 loops to itself:  $000$  -----,  
 $00 \rightarrow 01$ :  $000\boxed{01}$  -----,  $01 \rightarrow 10$ :  $000\boxed{10}$  ---->  
 $10 \rightarrow 01$ :  $000\boxed{101}$  ----,  $01 \rightarrow 11$ :  $000\boxed{1011}$  ----,  
 $11$  loops to itself:  $000\boxed{10111}$ ,  $11 \rightarrow 10$  and  
 $10 \rightarrow 00$  finish the sequence with  $000\boxed{101110}$ .  
 Because the sequence is cyclic, the final 0 is the same as the first 0:  $00010111...$