7.6 Strong Connectivity

7.6.1 Strongly Connected Graphs. We have defined connected directed graphs as directed graphs where any two vertices are joined by an undirected path. Now, we introduce the notion of strong connectivity where we require that any two vertices are joined by a **directed** path.

Definition. Given a directed graph G. We say that G is *strongly connected* if for any two vertices u, v there exists a directed path form u to v and a directed path from v to u. □

Remark. In the definition of a strongly connected graph we could, for any two vertices u and v, require only the existence of a directed path from u to v. Indeed, we require a directed path for **every** pair of vertices u, v, hence also for the pair v, u.

Notice that there always is a directed path from a vertex v to itself; indeed, it is the trivial path.

7.6.2 The following proposition gives a condition which guarantees that a connected directed graph is strongly connected.

Proposition. A connected directed graph is strongly connected if and only if every its edge is contained in a cycle. □

Justification: Assume that G is strongly connected and take an edge e, say from u to v. Since G is strongly connected, there exists a directed path P from v to u. If we add the edge e to P we obtain a cycle. So we have shown that e is contained in a cycle.

Assume that a directed graph G is connected and every its edge belongs to some cycle. We will show that for every $u, v \in V$ there is a directed path from u to v.

Take two vertices u and v. We know that G is a connected graph, hence there is an undirected path, say P, from u to v. If it is a directed path, we are done. Assume that in P there is an edge e for which P goes from the terminal vertex y of e to the initial vertex x of e. Denote by C a cycle in G containing e. Denote by P_e the cycle C without e. Then P_e is a directed path from x to y. Replace the edge e by P_e in P . We do the procedure above for every edge e of P such that the terminal vertex of e precedes the initial vertex of e. In such a way, we get a directed walk from u to v. Now, any directed walk from u to v contains a directed path from u to v. We have shown that for every pair u and v there is a directed path from u to v. Hence G is strongly connected. □

7.6.3 Components of Strong Connectivity. If a directed graph is not strongly connected we can ask about maximal subsets for which the induced subgraph is strongly connected. Such maximal sets of vertices are called strongly connected components.

Definition. Given a directed graph G. A set K of vertices is called a *strongly connected component*, also a *component of strong connectivity*, if the subgraph induced by K is strongly connected and it is maximal with this property (i.e. if we add to K any other vertex then the induced subgraph is not strongly connected). □

Remark. Every vertex of a directed graph G is contained in exactly one strongly connected component of G. The same does not hold for edges; if a graph is not strongly connected then it can contain edges that belong to no strongly connected component. Indeed, edges between two distinct components of strong connectivity.

7.6.4 Condensation of a Graph. The structure of strongly connected components of a directed graph is captured by so called condensation of a directed graph. For strongly connected graphs the condensation is a trivial graph which has one vertex and no edge.

Definition. Given a directed graph $G = (V, E)$. The *condensation of* G is a directed graph $\bar{G} = (\bar{V}, \bar{E})$ where $\bar{V}$ is the set of all strongly connected components of G, and there is an edge from a component K_1 to a component K_2 if and only if $K_1 \neq K_2$ and there exist vertices $u \in K_1$, $v \in K_2$ such that (u, v) is an edge of G. □

Remark. Note the the condensation of a directed graph is always an acyclic graph.

7.7 Euler graphs

In the section 7.1, we promised to give the theoretical background for the Euler's solution of the problem of seven bridges. Before that, let us recall that a trail in a graph is a walk where every edge is used at most once.

7.7.1 Euler Trails, Euler Graphs.

Definition. Given a directed graph G. A directed (an undirected) trail in G is called an *Euler trail* if it contains all edges.

Given an undirected graph G. An undirected trail in G is called an *Euler trail* if it contains all edges. □

Note that Euler trail can be open or closed. It is easy to notice that if a graph has an open Euler trail (a closed Euler trail) then it cannot have a closed (an open) one.

Definition. A directed (undirected) graph G is called an *Euler graph* if it contains a closed directed (undirected, respectively) Euler trail. □

7.7.2 Applications. Let us give two applications of Euler trails.

- **Drawing graphs with the smallest possible number of trails.** Given a connected undirected graph G. The task is to find the least possible number of edge disjunctive trails such that every edge of G is contained in some trail. It is evident that if G contains an Euler trail then every Euler trail is a solution. If in G there is no Euler trail then at least two trails are necessary to cover every edge of G.

  Solutions of this problem could be used for example if we draw a graph using a computer and we want to minimize the number of "shifts of the drawing pen".

- **Chinese Postman Problem.** A postman has to go along every street. How to do it if he wants to minimize the number of kilometers he has to walk?

  Let us construct a graph $G = (V, E)$ where V is the set of all crossings, and E is the set of streets which a postman must walk along. If in G, there exists an Euler closed trail then it is an optimal solution; indeed, in this case the postman will go along each street exactly once.

  If there is no closed Euler trail, the postman will have to go twice along some streets. To find a solution which will minimize the number of kilometers the postman must walk, we need the information how long each street is. So for every edge e a positive number $c(e)$ is given – its Length. We add parallel edges to the graph G in such a way that the resulting graph will contain a closed Euler trail and the sum of lengths of added edges will be the smallest possible.

7.7.3 Proposition. Given a connected directed graph G with at least two vertices. Then G contains a closed directed Euler trail if and only if for every vertex v of G the following holds

$$d^-(v) = d^+(v).$$

(In other words, the number of edges that terminate in v is the same as the number of edges that start in v.)

In a connected graph G there is a closed undirected Euler trail if and only if each vertex has an even degree. □

Justification. We give the proof only for directed graphs; for undirected graphs the proof is similar and easier.

It is evident that if a graph G has a closed directed Euler trail then it should satisfy the condition that $d^+(v) = d^-(v)$ for any vertex v. Indeed, if we "go" along such a closed trail we can match the edge along which we enter v with the edge along which we leave v. Hence there must be the same number of edges "going into v" as is the number of edged that "go out of v".

The other implication follows from an algorithm which constructs a closed directed Euler trail for a connected directed graph that satisfy the condition above. □

7.7.4 A Procedure for Finding a Closed Directed Euler Trail. Let G be a graph satisfying the condition from 7.7.3 with at least 2 vertices. We choose an arbitrary vertex v of G. Since G is connected and satisfies the condition above, for every vertex v there is at least one edge that starts in v and at least one edge that terminates in v.

We randomly form a directed trail starting in v as follows: We put $v_1 := v$, we go along an edge e_1 with the initial vertex v_1. Denote by v_2 the terminal vertex of e_1. Since $d^-(v_2) = d^+(v_2)$, there is an edge e_2 with the initial vertex v_2 that has not been used yet, we go along e_2 into its terminal vertex v_3. We continue in this manner till it is possible. The only situation when the trail cannot be extended is when we return to $v = v_1$ and every edge with the initial vertex v has already been used. Hence, we have randomly constructed a closed directed trail T.

If the trail T contains all edges of G we are ready; it is a closed directed Euler trail.

If the closed directed trail T does not contain all edges there exists an edge e which is not contained in T. Denote by x the initial vertex of e. Since G is connected, there exists an undirected path from v to x. This path must start with some edge incident to v and so contained in T. Hence there must be a vertex w where the undirected path from v to x leaves the trail T. Then it means that there is an edge incident to w that is not included in T. Since $d^-(w) = d^+(w)$, there is also an edge e' with the initial vertex w and not contained in T. Let us randomly form a maximal closed directed trail T_1 starting in w formed by edges that are not contained in T. Due to the condition in the proposition, T_1 must end in w. We disconnect T in w and insert T_1. So we get a new closed directed trail, say T'. If T' contains all edges, T' is a closed directed Euler trail. If not, there must be a vertex w' on T' and some edge incident to w' which is not contained in T' and we repeat the procedure.

Since G has only a finite number of edges, we must end up with a closed directed Euler trail. □

7.7.5 Open Euler Trials. Now, we state a characterization of connected graphs (directed and undirected) that contain an open Euler trial.

Proposition. Given a connected directed graph G. Then G contains an open directed Euler trail if and only if there exist two vertices u_1, u_2 such that

$$d^-(u_1) = d^+(u_1) - 1, \;\; d^-(u_2) = d^+(u_2) + 1,$$

and for every other vertex v of G it holds that $d^-(v) = d^+(v)$.

In a connected graph G there exists an open undirected Euler trail if and only if there are two vertices with odd degree and all other vertices have even degree. □

Justification. Again, we prove the proposition for directed Euler trials only. The proof for undirected Euler trials is analogous.

It is easy to see that if there is an open Euler trail in G then G satisfies the condition from the proposition; indeed, u_1 is the vertex where the Euler trail starts, and u_2 where it terminates.

To construct an open Euler trail we proceed similarly as in 7.7.4. We start in u_1 and randomly construct a maximal trail T. Because of the conditions, T must terminate in u_2. If T contains all edges, it is an open Euler trail. If not, there must be a vertex w, $w \neq u_1$, $w \neq u_2$, where no all edges incident with w were used. We insert into T a maximal trail from w to w containing edges not from T. After a finite number of inserting closed trails, we get an open Euler trail. □

7.8 Hamiltonian graphs

Recall that a path is a trail where vertices are not repeated, only in a closed path (a circuit if undirected, or a cycle if directed) initial and terminal vertices are the same.

7.8.1 Hamiltonian paths, circuits, and cycles. Another type of important walks in a graph are paths (closed or open) that contain all vertices. Such paths/circuits/cycles will be called Hamiltonian paths/circuits/cycles. More precisely:

Definition. Given a graph G. An open path is called a *Hamiltonian path* if it contains all vertices of G (and hence every vertex precisely once).

Similarly, a *Hamiltonian circuit* is a circuit that contains every vertex of the graph; a *Hamiltonian cycle* is a cycle that contains every vertex of the graph. □

7.8.2 Problems concerning Hamiltonian paths can be divided into two groups — existential and optimization problems. In existential problems we want to find out whether a given graph contains a Hamiltonian path (circuit, cycle, respectively). In optimization problems every edges of a graph has its *weight* (or *length*) which is an integer. In this case, the problem is to find a Hamiltonian path (circuit, cycle, respectively) with optimal sum of weights (lengths) of its edges.

Unlike problems where we have to find Euler trails, the problems concerning Hamiltonian paths, circuits, and/or cycles are rather difficult. A fast algorithm is not known that would solve either existential or optimization problems concerning Hamiltonian paths. Despite of this fact (and maybe due to this fact), the problems of Hamiltonian paths have lot of applications. In the next paragraph, we state some of them.

7.8.3 Applications. Let us give two applications of the problem of Hamiltonian paths.

- **Traveling Salesman Problem.** The original problem is the following: Given n towns, we denote them $1, 2, \dots, n$. For every pair of distinct towns i, j their distance $d(i, j)$ is given. The task is to find a succession of towns in which a salesman can visit the towns so that the total distance traveled is the smallest possible one.

  The problem can be reformulated as a graph problem: Given a complete graph G with the set of vertices $V = \{1, 2, \dots, n\}$, i.e. for every two distinct vertices $i \neq j$ there is an edge $\{i, j\}$ in G. For every $\{i, j\}$, we define its length to be $d(i, j)$. The goal is to find a Hamiltonian circuit C for which
  $$\sum_{e \in C} d(e)$$
  is the smallest possible.

- **Planning of processes.** Assume the following situation: we have a device on which processes $p_1, p_2, \dots, p_n$ are carried out. Moreover, there are pairs of processes p_i, p_j such that if p_i should follow p_j it is necessary to clean, convert, etc. the device. So in this case, it is necessary to pay some price in order to realize p_j just after finishing p_i.

  The task is to find a sequence of processes such that the sum of prices for its realization is zero. If such sequence does not exist, the problem is to find a sequence where the sum of prices is as small as possible.

This second case leads to the Traveling Salesman Problem.

7.8.4 There are easy necessary conditions for a graph to have a Hamiltonian path (circuit, cycle, respectively). Let us state some of them.

- If in a graph G there is a Hamiltonian path then G must be connected. (Indeed, a disconnected graph cannot have a Hamiltonian path.)
- If in a graph G there is a Hamiltonian circuit then every vertex of G must have degree at least 2.
- If in a graph G there is a Hamiltonian cycle then G must be strongly connected.

A nontrivial necessary and sufficient condition for finding whether a given graph has a Hamiltonian path (circuit, circle, respectively) is not known.