Elementary Introduction to Graph Theory

(01EIG 2025/2026)

Lecture 2

Francesco Dolce

dolcefra@fit.cvut.cz

October 3, 2025

updated: October 6, 2025

PDF available at the address: dolcefra.pages.fit/ens/2526/EIG-lecture-02.pdf

Solution of Exercise in previous Lecture. Determine which of the following sequences are graphic, and in the positive case find a graph realising them: $(7, 6, 5, 4, 3, 3, 2)$, $(3, 3, 2, 2, 1, 1)$.

- $(7, 6, 5, 4, 3, 3, 2)$ is not a graphic sequence, since $7 > 6$.
- $(\mathbf{3}, \underline{3, 2, 2}, 1, 1) \rightarrow (\mathbf{2}, \underline{1, 1}, 1, 1) \rightarrow (0, 0, 1, 1) \sim (\mathbf{1}, \underline{1}, 0, 0) \rightarrow (0, 0, 0)$ is graphic (see Figure 1).

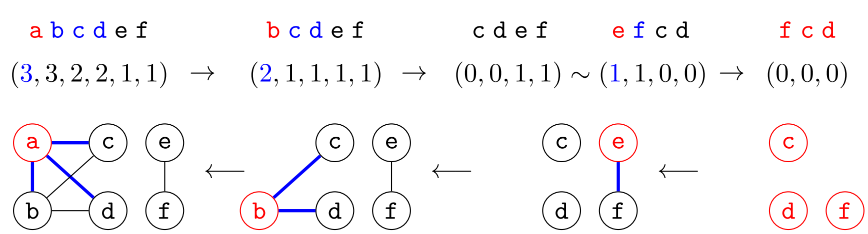


Figure 1: Realising a graph from a graphic sequence.

1 Subgraphs

Let $G = (V, E)$ be a graph. A graph $H = (U, F)$ is a *subgraph* of G, written $H \subseteq G$ (or, G is a *supergraph* of H, written $G \supseteq H$) if $U \subseteq V$ and $F \subseteq E$. We say that H is *contained in* G (or, G *contains* H).

Example 1 The graphs H_1, H_2, H_3 in Figure 2 are subgraphs of G. On the other hand, H_4 is not a subgraph of G.

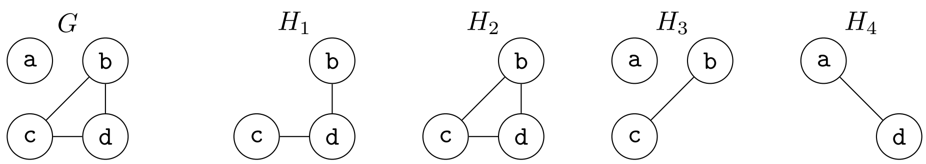


Figure 2: Five graphs with vertices (subsets of) $\{\mathtt{a}, \mathtt{b}, \mathtt{c}, \mathtt{d}\}$.

A subgraph $H = (U, F)$ of $G = (V, E)$ is called an *induced subgraph* of G by U, and we write $H = G[U]$, if for every two vertices $x, y \in U$ we have $\{x, y\} \in F \Leftrightarrow \{x, y\} \in E$. That is, the vertices in H are connected by exactly the same edges as in G. When H is simple we have $H = G[U] = \Big(U, E \cap \binom{U}{2}\Big)$.

Example 2 Let G, H_1, H_2 and H_3 as in Example 1. We have $H_2 = G[\{\mathtt{b}, \mathtt{c}, \mathtt{d}\}]$ and $H_3 = G[\{\mathtt{a}, \mathtt{b}, \mathtt{c}\}]$. Note that H_1 is not an induced subgraph of G, since $\{\mathtt{b}, \mathtt{c}\}$ is an edge of G but not of H_1.

If $U \subset V$ and $F \subset \binom{V}{2}$, we write $G - U = G[V \setminus U]$. That is, $G - U$ is obtained from G by deleting all vertices in U and their incident edges.

Given a graph $G = (V, E)$ and a set of edges F, the graph $G \setminus F = (V, E \setminus F)$ is obtained by deleting some edges, the ones in $E \cap F$, but keeping all vertices. We call $G \setminus F$ a *spanning subgraph* of G.

In a similar way, given a set of vertices U and a set of edges F, we define $G + U = (V \cup U, E)$ and $G + F = (V, E \cup F)$.

When $U = \{u\}$ we just write $G + u$ instead of $G + \{u\}$ and $G - u$ instead of $G - \{u\}$. Similarly when $E = \{e\}$ we write $G + e$ and $G \setminus e$.

Example 3 Let $G, H_1 H_2, H_3$ and H_4 as in Example 1. Then $H_2 = G - \mathtt{a}$ and $H_3 = G - \mathtt{d}$. The graph H_1 is a spanning subgraph of H_2, indeed $H_1 = H_2 \setminus \{\mathtt{b}, \mathtt{c}\}$. Obviously we also have $H_2 = H_1 + \{\mathtt{b}, \mathtt{c}\}$. Moreover $G = H_2 + \mathtt{a}$.

2 Paths and cycles

A *path* is a non-empty graph $P = (V, E)$ with $V = \{v_0, v_1, \dots, v_k\}$ and $E = \{\{v_0, v_1\}, \{v_1, v_2\}, \dots, \{v_{k-1}, v_k\}\}$ The vertices v_0, v_k are *linked* by P and are called its *endpoints*. The vertices $v_1, v_2, \dots, v_{k-1}$ are the *inner vertices* of P.

If $k \geq 3$, we call *cycle* a graph of the form $C = (P - v_k) + \{v_{k-1}, v_0\}$.

The *length* of P (resp., of C) is k. A path (resp., cycle) is *odd* or *even* according to the parity of k. When a path (resp., cycle) appears as a subgraph G, we say that the path (resp., cycle) is *in* G.

We can represent a path (resp., a cycle) as a sequence of vertices

$$(v_0, v_1, \ldots, v_k)$$

or as a sequence of edges

$$(e_1, e_2, \ldots, e_k)$$

or, when interested in both vertices and edges, using the notation

$$(v_0 \xrightarrow{e_1} v_1 \xrightarrow{e_2} \cdots \xrightarrow{e_{k-1}} v_{k-1} \xrightarrow{e_k} v_k)$$

where $e_i = \{v_{i-1}, v_i\}$.

The *distance* between two vertices $u, v \in V$ in a graph $G = (V, E)$, denoted by $d_G(u, v)$, is the length of the shortest path in G with endpoints u and v. The *eccentricity* of v is defined as $\mathrm{ecc}_G(v) = \max\{d_G(v, u) \mid u \in V\}$.

The *center* of G is the set of its vertices with minimal eccentricity, i.e.,

$$\mathcal{C}(G) = \mathrm{argmin}_{v \in V} \mathrm{ecc}_G(v) = \left\{ v \in V \mid \mathrm{ecc}_G(v) = \min_{u \in V} \mathrm{ecc}_G(u) \right\}.$$

Example 4 The path P_4 and the cycle C_3 are represented on the left of Figure 3. On the right of the same figure you can see a path of length 5, $(\mathtt{b}, \mathtt{c}, \mathtt{d}, \mathtt{e}, \mathtt{f})$, and a cycle of length 4, $(\mathtt{b}, \mathtt{c}, \mathtt{d}, \mathtt{e})$, in a graph G.

Note that we have $d_G(\mathtt{b}, \mathtt{f}) = 2$ since a shortest path in G between the two vertices is $(\mathtt{b}, \mathtt{e}, \mathtt{f})$. One can check that, e.g., $\mathrm{ecc}_{P_4}(v_0) = 4$, $\mathrm{ecc}_{P_4}(v_3) = 3$; $\mathrm{ecc}_{C_3}(u_2) = 1$; $\mathrm{ecc}_G(\mathtt{a}) = 3$, and $\mathrm{ecc}_G(\mathtt{c}) = 2$. Moreover $\mathcal{C}(P_4) = \{v_2\}$, $\mathcal{C}(C_3) = \{u_0, u_1, u_2\}$, and $\mathcal{C}(G) = \{\mathtt{b}, \mathtt{c}, \mathtt{d}, \mathtt{e}\}$.

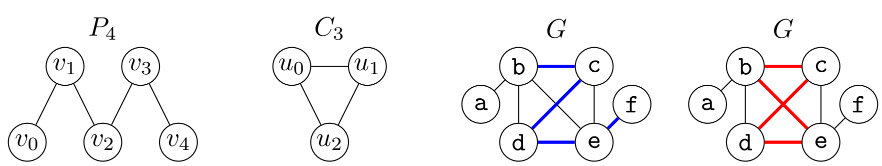


Figure 3: Paths and cycles.

3 Connected graphs

A graph is *connected* if it is non-empty and every two of its vertices are linked by a path in it. A maximal connected subgraph is a *component* of the graph. A graph with more than one component (i.e., that is not connected) is called *disconnected*.

Example 5 The graph G in Figure 4 has three components:

$$H_1 = G[\{\mathtt{a}, \mathtt{b}, \mathtt{c}, \mathtt{d}, \mathtt{e}\}], \quad H_2 = G[\{\mathtt{f}, \mathtt{g}, \mathtt{h}, \mathtt{i}\}] \quad \text{and} \quad H_3 = G[\{\mathtt{j}\}].$$

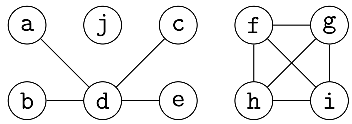
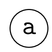


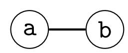
Figure 4: A graph with three components.

Let S_n be the number of different connected graphs on n vertices.

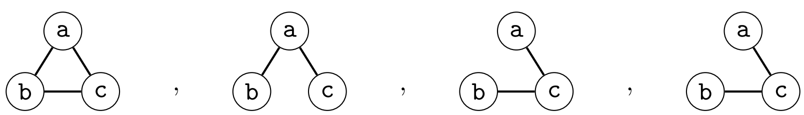
- If $V = \{\mathtt{a}\}$, there is only one possible graph, so $S_1 = 1$.



- If $V = \{\mathtt{a}, \mathtt{b}\}$, the two vertices are connected by an edge, so $S_2 = 1$.



- If $V = \{\mathtt{a}, \mathtt{b}, \mathtt{c}\}$, there are four possible connected graphs, so $S_3 = 4$.



Theorem 1 *For every $n \in \mathbb{N}$*

$$n \cdot 2^{\binom{n}{2}} = \sum_{k=1}^{n} \binom{n}{k} S_k \cdot k \cdot 2^{\binom{n-k}{2}}.$$

Proof. Let us count, in two different ways, all "rooted" graphs on n vertices, i.e., graphs with one particular vertex emphasised.

We know that the number of labelled graphs (connected or not) is $2^{\binom{n}{2}}$. So, the total number of rooted graphs is $n \cdot 2^{\binom{n}{2}}$.

On the other hand, each "root" will appear in a connected component of size k, with $1 \leq k \leq n$. For a fixed k, we have $\binom{n}{k}$ possibilites to select the k vertices, S_k ways of having the component connected, and k ways of selecting the "root" in the connected component; we do not know whether the remaining $n - k$ vertices are connected or not, so we have $2^{\binom{n-k}{2}}$ possibilites for them. ∎

Theorem 1 is quite unsatisfactory since it is a recursive formula. To compute, e.g., S_{20} one needs to successively compute $S_1, S_2, S_3, \ldots, S_{19}$.

Example 6 We can compute S_4 using Theorem 1, since

$$4 \cdot 2^{\binom{4}{2}} = \binom{4}{1} S_1 \cdot 1 \cdot 2^{\binom{3}{2}} + \binom{4}{2} S_2 \cdot 2 \cdot 2^{\binom{2}{2}} + \binom{4}{4} S_3 \cdot 3 \cdot 2^{\binom{1}{2}} + \binom{4}{4} S_4 \cdot 4 \cdot 2^{\binom{0}{2}}$$

we have, using the known values of S_1, S_2 and S_3,

$$4 \cdot 64 \ = \ 4 \cdot 1 \cdot 1 \cdot 8 \ + \ 6 \cdot 1 \cdot 2 \cdot 2 \ + \ 4 \cdot 4 \cdot 3 \cdot 1 \ + \ 1 \cdot S_4 \cdot 4 \cdot 1.$$

Hence $S_4 = 64 - 8 - 12 = 38$.

Exercise. Find all connected simple graphs of order 4.

4 Walks

Given a simple graph $G = (V, E)$, a sequence $(v_i)_{i=0}^k$ of vertices $v_i \in V$ is called a *walk* (or *k-walk*) in G if for every $1 \leq j \leq k$ we have $\{v_{j-1}, v_j\} \in E$.

A walk is *closed* if its initial and terminal vertices are the same. A walk where all its edges are distinct is called a *trail*.

The notions of *length*, *endpoints*, *inner vertices* in a walk are defined analogously as in a path. Nevertheless, note that, contrary to paths (resp., cycles), vertices in a walk (resp., closed walk) can be repeated.

Example 7 Let G be the graph in Figure 5. The sequence $(\mathtt{d}, \mathtt{a}, \mathtt{c}, \mathtt{a}, \mathtt{b})$ is a walk in G but not a path. The sequence $(\mathtt{a}, \mathtt{c}, \mathtt{b}, \mathtt{c}, \mathtt{a})$ is a closed walk in G but not a cycle. The sequence $(\mathtt{d}, \mathtt{a}, \mathtt{b}, \mathtt{c})$ is both a walk and a path. The sequence $(\mathtt{d}, \mathtt{a}, \mathtt{b}, \mathtt{c}, \mathtt{a})$ is a trail (but not a path nor a cycle).

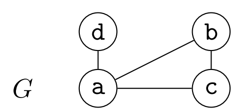


Figure 5: A graph G.

Connectedeness of a pair of vertices in a graph through a walk is an equivalence relation. Indeed:

- every vertex is connected by a 0-walk to itself;
- if there is a walk $W = (v_i)_{i=0}^k$ from a to b, then the reverse walk $\widetilde{W} = (v_{k-i})_{i=0}^k$ connects b to a;
- if a, b are connected by a walk $W_1 = (v_i)_{i=0}^k$ and v, w are connected by a walk $W_2 = (u_i)_{i=0}^h$, then there is a walk from u to w obtained $W_1 W_2 = (w_i)_{i=0}^{k+h}$, where $w_i = v_i$ for $0 \leq i \leq k$ and $w_i = u_{i+k}$ for $k+1 \leq k \leq k+h$.

The equivalence class of a vertex v determined by the connectedness relation gives the component of the graph containing the vertex v.

We can also consider walks in non simple graphs.

Example 8 Let G be the graph in Figure 6. A walk in G is given by

$$(h, j, j, i, e, f, f) = (\mathtt{b} \xrightarrow{h} \mathtt{c} \xrightarrow{j} \mathtt{c} \xrightarrow{j} \mathtt{c} \xrightarrow{i} \mathtt{b} \xrightarrow{e} \mathtt{a} \xrightarrow{f} \mathtt{d} \xrightarrow{f} \mathtt{a}).$$

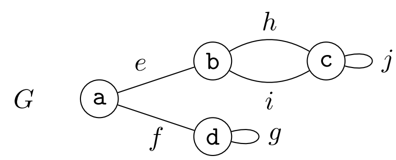


Figure 6: A non simple graph G.

5 Trees

A graph that does not have any cycle in it is called *acyclic*. An acyclic graph is also called a *forest*. A connected forest, i.e., a graph that is both connected and acyclic, is called a *tree*. Vertices of degree 1 in a tree are called *leaves*.

Example 9 The graph G on the left of Figure 7 is not acyclic. The graph H on the centre is a forest. The graph $K_{1,5}$, on the right, is a tree.

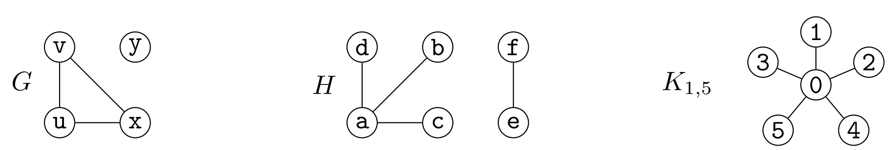


Figure 7: A graph with a cycle (left) a forest (centre) and a tree (right).

The trivial graph has exactly one leaf. Every tree having more than one vertex has at least 2 leaves. Indeed, it is enough to consider the endpoints of a longest path.

Theorem 2 *Let $G = (V, E)$ be a graph s.t. $d_G(v) \geq 2$ for every $v \in V$. Then G contains a cycle.*

Proof. The statement is clear if G has a loop or two parallel edges. Thus, WLOG, let us suppose that G is simple.

Let $P = (v_0, v_1, \dots, v_{k-1}, v_k)$ be a longest path in G. Since $d_G(v_k) \geq 2$, then there exists $v \in \mathrm{N}_G(v_k)$ with $v \neq v_{k-1}$. If $v \notin P$, then $P + \{v\}$ would be a path longer than P, which is a contradiction. Thus, $v = v_i \in P$, which implies that $(v_i, \ldots, v_{k-1}, v_k)$ is a cycle in G. ∎

A graph $G = (V, E)$ is *minimally connected* if it is connected but $G \setminus e$ is disconnected for every $e \in E$. It is *maximally acyclic* if it is acyclic but for any two vertices $u, v \in V$ s.t. $\{u, v\} \notin E$ we have that $G + \{u, v\}$ contains a cycle.

Theorem 3 *The following are equivalent for a graph $G = (V, E)$.*

1. *G is a tree.*

2. *Any two vertices in V are linked by a unique path in G.*

3. *G is minimally connected.*

4. *G is maximally acyclic.*

Proof. Exercise. ∎

Corollary 1 (Euler's formula for trees) *A connected graph of order n is a tree if and only if it has size $n - 1$.*

The hypothesis of connectedness is necessary.

Example 10 Let us consider the graphs G, H and $K_{1,5}$ in Example 9. The graph H, a forest but not a tree, has order 6 and size 4. The tree $K_{1,5}$ has order 5 and size 4. The graph G, which is not a forest, has order 4 and size 3.

Theorem 4 (Cayley's formula) *Let $\#V = n$. The number of trees with set of vertices V is n^{n-2}.*

Example 11 The only tree with one vertex is the trivial one. The only tree with two vertices $\mathtt{a}, \mathtt{b}$ is the one with one edge $\{\mathtt{a}, \mathtt{b}\}$. If the set of vertices is $\{\mathtt{a}, \mathtt{b}, \mathtt{c}\}$ we have three possible trees (see Figure 8). According to Theorem 4 there are 16 labelled trees on $V = \{\mathtt{a}, \mathtt{b}, \mathtt{c}, \mathtt{d}\}$ (which ones?).

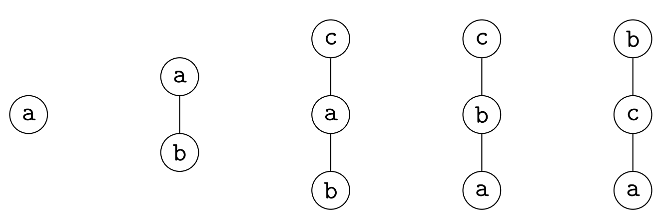


Figure 8: Possible trees on 1, 2 and 3 vertices.

A *rooted tree* $T(r)$ is a tree $T = (V, E)$ with a specified vertex $r \in V$ called the *root* of T. Given a vertex v in a rooted tree $T(r)$ different than r, we say that a vertex in $\mathrm{N}_T(v)$ on the unique path between r and v is a *parent* of v; all vertices in this path are called *ancestors* of v. Similarly, a vertex is a *child* of its parent, and u is a *descendant* of v if v is an ancestor of u.

We call *level* $\ell_T(v)$ of a vertex v in a rooted tree $T(r)$ the distance $d_T(r, v)$ (i.e., the length of the path from r to v in the tree). The root is the only vertex with level 0. The *height* of a rooted tree is the maximal level.

Note that when dealing with rooted trees we call *leaf* a vertex of degree 1 that has no children. Hence, except for the trivial tree with one vertex, the root is (in this context) never a leaf.

Example 12 The graph in Figure 9 is a rooted tree with root `r`. The vertices `a,b,c` are children of `r`. The vertex `h` is a child of `e` and a descendant of `a` (hence `e` is a parent of `h` and `a` an ancestor of `h`). The vertices `d,e,f,g` have level 2, while the height of the tree is 3 (`h` is a vertex of maximal level).

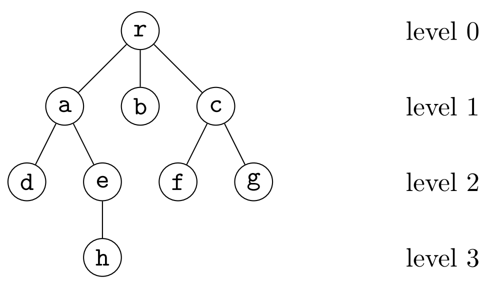


Figure 9: A rooted tree.

An *ordered tree* $T(r, \preceq)$ is a rooted tree $T(r) = (V, E)$ with root r and a partial order $\preceq$ on V giving the order of children of each vertex. We usually represent it by drawing the children of a node from left to right starting from the smallest.

Example 13 Let T be the rooted tree in Example 12 and represented in Figure 9. Then, $T(\mathtt{r}, \preceq)$, with the partial order $\preceq$ defined by

$$\mathtt{a} \prec \mathtt{b} \prec \mathtt{c}, \qquad \mathtt{d} \prec \mathtt{e} \qquad \text{and} \qquad \mathtt{f} \prec \mathtt{g}$$

is a ordered tree.