2 Graphs

2.1 Directed Graphs

Both in Computer Science and in the rest of Mathematics *graphs* are studied and used frequently. Graphs come in two flavors, *directed* graphs and *undirected* graphs.

There is no fundamental difference between directed graphs and relations: a directed graph just *is* an endorelation on a given set. If V −for Vertexes− is this set and if E −for Edges− is the relation we call the pair (V, E) a directed graph. Usually, set V will be finite, but this is not really necessary: infinite graphs are conceivable too. A directed graph (V, E) is finite if and only if V is finite. Unless stated otherwise, we confine our attention to finite graphs. Always set V will be *nonempty*.

Traditionally, the elements of set V are called "vertexes" or "nodes", whereas the elements of E, that is, the pairs (u, v) satisfying $u\,E\,v$, are called "directed edges" or "arrows". In this terminology we say that the graph contains "an arrow from u to v" if and only if $u\,E\,v$. Also, in this case, we say that u is a "predecessor" of v and that v is a "successor" of u.

Graphs can be represented by pictures, in the following way. Every vertex is drawn as a small circle with its name inside the circle, and every arrow from u to v is drawn as an arrow from u's circle to v's circle. Such a picture may be attractive because it enables us to comprehend, in a single glance, the whole structure of a graph, but, of course, drawing such pictures is only feasible if the set of vertexes is not too large. Figures 1 and 2 give simple examples.

41.

Figure 1: The smallest directed graphs: $V = \{ a \}$ with $E = \emptyset$ and $E = \{ (a, a) \}$

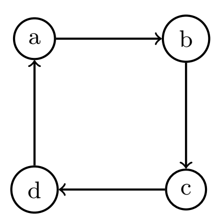

Figure 2: The graph of relation $\{ (a, b) , (b, c) , (c, d) , (d, a) \}$

If we are only interested in the pattern of the arrows we may omit the names of the vertexes and simply draw the vertexes as dots. The resulting picture is called an "unlabeled" (picture of the) graph.

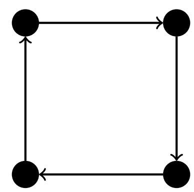

Figure 3: The same graph, unlabeled

Relation E may be such that $u\,E\,u$, for some u. In terms of graphs this means that a vertex may have an arrow from itself to itself. This is perfectly admissible, although in some applications such "auto-arrows" may be undesirable. Notice that the property "having no auto-arrows" is the directed-graph equivalent of the relational property "being irreflexive".

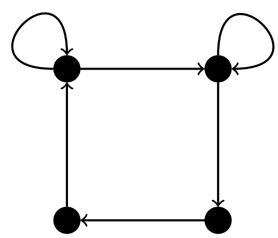

Figure 4: The unlabeled graph of relation $\{\ (a,a)\ ,\ (a,b)\ ,\ (b,b)\ ,\ (b,c)\ ,\ (c,d)\ ,\ (d,a)\ \}$

2.2 Undirected Graphs

Sometimes we are only interested in the (symmetric) concept of nodes being connected, independent of any notion of direction. An "undirected graph" is a symmetric (endo)relation E on a set V. As before, we call the elements of V "nodes" or "vertexes". The pairs (u,v) satisfying $u\,E\,v$ are now called "edges"; we also say that such u and v are "directly connected" or "neighbors".

Relation E being symmetric means that $u\,E\,v$ is equivalent to $v\,E\,u$; hence, being neighbors is a symmetric notion: edge (u,v) is the same as edge (v,u). In this view an undirected graph just is a special case of a directed graph, with this characteristic property: the graph contains an arrow from u to v if and only if the graph contains an arrow from v to u. So, arrows occur in pairs. See Figure 5, for a simple example. A more concise rendering of an undirected graph is obtained by combining every such pair of arrows into a single, undirected edge, as in Figure 6.

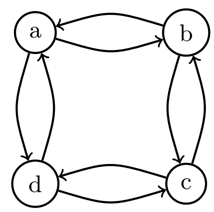

Figure 5: The graph of $\{\ (a, b)\ ,\ (b, a)\ ,\ (b, c)\ ,\ (c, b)\ ,\ (c, d)\ ,\ (d, c)\ ,\ (d, a)\ ,\ (a, d)\ \}$

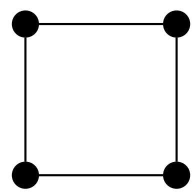

Figure 6: The same graph, with undirected edges and unlabeled

There is no fundamental reason why undirected graphs might not also contain edges connecting a node to itself. Such edges are called "auto loops". That is, if $u\,E\,u$ then u is directly connected to itself, so u is a neighbor to itself.[1] It so happens, however, that in undirected graphs auto-loops are more a nuisance than useful: many properties and theorems obtain a more pleasant form in the absence of auto-loops.

Therefore, we adopt the convention that undirected graphs contain no auto-loops. Formally, this means that an undirected graph is an *irreflexive* and symmetric relation.

In the case of finite graphs we sometimes wish to *count* the number of arrows or edges. We adopt the convention that, in an undirected graph, every pair of directly connected nodes counts as a single edge, even though this single edge corresponds to *two* arrows in the corresponding undirected graph. This reflects the fact that, in a symmetric relation, the pairs (u, v) and (v, u) are indistinguishable. For example, according to this convention, the undirected graph in Figure 6 has four edges.

*　　*　　*

We have defined an undirected graph as an irreflexive and symmetric directed graph. Every directed graph can be transformed into an undirected one, just by "ignoring the directions of the arrows". In terms of relations this amounts to taking the symmetric closure of the relation and removal of the auto-arrows: in the undirected graph nodes u and v are neighbors if and only if, in the directed graph, there is an arrow from

[1]This shows that we should not let ourselves be confused by the connotations of the everyday-life word "neighbor": here the word is used in a strictly technical meaning.

u to v or from v to u (or both), provided $u \neq v$. For example, the directed graph in Figure 4 can thus be transformed into the undirected graph in Figure 6.

2.3 A more compact notation for undirected graphs

We have defined an undirected graph as an irreflexive − no edge between a node and itself − and symmetric relation. Although this is correct mathematically, it is not very practical. For example, the set of edges of the graph in Figure 5 now is $\{ (a,b), (b,a), (b,c), (c,b), (c,d), (d,c), (d,a), (a,d) \}$, in which every edge occurs *twice*: that nodes a and b, for instance, are connected is represented by the presence of both (a,b) and (b,a) in the set of edges. Yet, we do wish to consider this connection as a *single* undirected edge. It is awkward, then, to have to write down both (a,b) and (b,a) to represent this single edge. We would rather not be forced to distinguish these pairs.

We obtain a more convenient representation by using by two-element[2] sets $\{u,v\}$: as set $\{u,v\}$ equals set $\{v,u\}$ we only need to write this down once. So, in the sequel, an undirected graph will be a pair (V,E), where V is the set of nodes, as usual, and where E is a set of pairs $\{u,v\}$, with $u,v \in V$ and $u \neq v$, and such that:

$$\{u,v\} \in E \quad \Leftrightarrow \quad \text{“}u \text{ and } v \text{ are connected”} \quad .$$

For example, the set of edges of the graph in Figure 5 can now be written as: $\{ \{a,b\}, \{b,c\}, \{c,d\}, \{d,a\} \}$.

In the number of edges, written as $\#E$, we do not double count the edges in two directions, so in this example we have $\#E = 4$.

Although it is usual to write $(.,.)$ for ordered pairs, in which $(a,b) \neq (b,a)$, and in sets elements have no order by which $\{a,b\} = \{b,a\}$, in the literature one often sees (a,b) to denote an edge in an undirected graph.

2.4 Additional notions and some properties

Occasionally, we use infix operators for the relations in directed and undirected graphs. That is, sometimes we write $u\,E\,v$ as $u \rightarrow v$ and we speak of directed graph $(V, \rightarrow)$ instead of (V,E). Similarly, for symmetric relations we sometimes use $u \sim v$ instead of $u\,E\,v$ and we speak of undirected graph $(V, \sim)$. So, in this nomenclature, $u \rightarrow v$ means "the graph has an arrow from u to v" and $u \sim v$ means "in the graph u and v are neighbors".

In a directed graph $(V, \rightarrow)$, for every node u the *number of* nodes v satisfying $u \rightarrow v$ is called the "out-degree" of u, whereas the number of nodes u satisfying $u \rightarrow v$ is called the "in-degree" of v, provided these numbers are *finite*. Notice that if V is finite the in-degree and out-degree of every node are finite too. An auto-arrow adds 1, both to the in-degree and the out-degree of its node.

If relation $\rightarrow$ is symmetric, so $u \rightarrow v \Leftrightarrow v \rightarrow u$ for all u,v, then the in-degree of every node equals its out-degree.

[2]Undirected graphs contain no auto-edges, so the pair (u,u) is not an edge.

In an undirected graph $(V, \sim)$, the "degree" of a node u is its number of neighbors, that is, the number of nodes v with $u \sim v$. Thus, the degree of a node in an undirected graph equals the in-degree and the out-degree of that node in the underlying directed graph.

We write in, out, and deg for "in-degree", "out-degree", and "degree" respectively.

So for a directed graph (V, E) we have for $u, v \in V$:

$$\begin{aligned} in(v) &= \#\{ u \in V \mid (u, v) \in E \} \\ out(u) &= \#\{ v \in V \mid (u, v) \in E \}. \end{aligned}$$

By straightforward addition we obtain:

$$\textstyle\sum_{v \in V} in(v) = \#E = \sum_{u \in V} out(u).$$

For an undirected graph (V, E) we have

$$deg(u) = \#\{ v \mid \{u, v\} \in E \}.$$

2.1 Theorem. In an undirected graph (V, E) we have

$$\sum_{v \in V} deg(v) = 2 * \#E.$$

Proof. The number of ends of edges can be counted in two ways.

In the first one one observes that every edge has two ends, and since there are $\#E$ edges, this number is $2 * \#E$.

In the second one one observes that the degree of a node is the number of ends of edges to which it is attached. Adding all these yields $\sum_{v \in V} deg(v)$, so these numbers are equal. □

* * *

With N for the size of V, so N equals the number of vertexes in the graph, we have that the degree of every node is at most $N-1$. If the degree of a node equals $N-1$ then this node is a neighbor of *all other* nodes. If every node in an undirected graph has this property, the graph is called "complete". Similarly, a directed graph is complete if it contains an arrow from every node to every node. Thus, the complete directed graph corresponds to the complete relation $\top$, whereas the complete undirected graph corresponds to the relation $\top \backslash I$ (because of the omission of auto-loops). The complete undirected graph with N nodes is called the complete N-graph. Figure 7, for example, gives a picture of the complete 5-graph.

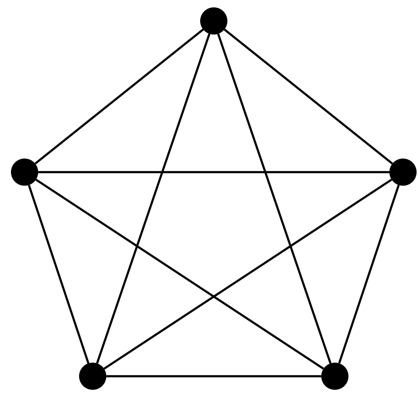

Figure 7: The complete 5-graph, unlabeled

2.5 Connectivity

2.5.1 Paths

We simultaneously consider a directed graph $(V, \rightarrow)$ and an undirected graph (with the same set of nodes) $(V, \sim)$. A *directed path from* node u to node v is a finite sequence $[\, s_0, \cdots, s_n\,]$ consisting of $n{+}1$, $0 \leq n$, nodes satisfying:

$$u = s_0 \ \wedge\ (\forall i: 0 \leq i < n: s_i \rightarrow s_{i+1}) \ \wedge\ s_n = v \quad .$$

Although this path contains $n{+}1$ nodes, it pertains to only n arrows, namely the n pairs (s_i, s_{i+1}), for all $i: 0 \leq i < n$. Therefore, we say that the *length* of this path equals n: the length of a path is the number of arrows in it. If $n = 0$ the path contains no arrows and we have $u = v$: the only paths of length 0 are the one-element sequences $[\, u\,]$ which are paths from u to u, for every node u. Paths of length 0 are called "empty" whereas paths of positive length are called "non-empty".

Similarly, in an undirected graph an *undirected path from* node u to node v is a finite sequence $[\, s_0, \cdots, s_n\,]$ consisting of $n{+}1$, $0 \leq n$, nodes satisfying:

$$u = s_0 \ \wedge\ (\forall i: 0 \leq i < n: s_i \sim s_{i+1}) \ \wedge\ s_n = v \quad .$$

Again, the length of this path is n, being the number of edges in it.

Whenever no confusion is possible, we simply use "path" instead of "directed path" or "undirected path". In any, directed or undirected graph, we call nodes u and v "connected" if the graph contains a path from u to v. Every node is connected to itself, because we have seen that for every node u a path, of length 0, exists from u to u.

In relational terms being connected means being related by the reflexive-transitive closure of the relation. In what follows, we denote the reflexive-transitive closures of relations $\rightarrow$ and $\sim$ by $\stackrel{*}{\rightarrow}$ and $\stackrel{*}{\sim}$, respectively, and we denote their transitive closures by $\stackrel{+}{\rightarrow}$ and $\stackrel{+}{\sim}$, respectively.

2.2 Lemma. In a directed graph the relation "is connected to" equals $\stackrel{*}{\rightarrow}$.

Proof. From the chapter on relations we recall the property $R^* = (\bigcup_{n:0\leq n} R^n)$; in terms of $\rightarrow$ this can be written as: $\stackrel{*}{\rightarrow} = (\bigcup_{n:0\leq n} \stackrel{n}{\rightarrow})$, where $\stackrel{n}{\rightarrow}$ denotes the equivalent of R^n . This means that $u \stackrel{*}{\rightarrow} v$ is equivalent to $(\exists n : 0\leq n : u \stackrel{n}{\rightarrow} v)$, whereas "u is connected to v" is equivalent to
$(\exists n : 0\leq n :$ "u is connected to v by a path of length n" $)$. We now prove the equivalence of these two characterizations term-wise; that is, for all natural n we prove that $u \stackrel{n}{\rightarrow} v$ is equivalent to "u is connected to v by a path of length n". We do so by Mathematical Induction on n :

$u \stackrel{0}{\rightarrow} v$
$\Leftrightarrow$ { definition of $\stackrel{0}{\rightarrow}$ }
$u\,I\,v$
$\Leftrightarrow$ { definition of I }
$u = v$
$\Leftrightarrow$ { definition of path }
"the path $[u]$, of length 0, connects u to v"
$\Leftrightarrow$ { definition of "connected", see below }
"u is connected to v by a path of length 0" .

As to the logical equivalence in the last step of this derivation: in the direction "$\Rightarrow$" this is just $\exists$-introduction; in the direction "$\Leftarrow$" we observe: for *every* path $[x]$, of length 0, we have that if $[x]$ connects u to v then $x = u$, hence $[x] = [u]$. (That is, the path of length 0 connecting u to v is *unique*.)

Furthermore, we derive, for $0\leq n$ and for nodes u, w :

$u \stackrel{n+1}{\rightarrow} w$
$\Leftrightarrow$ { definition of $\stackrel{n+1}{\rightarrow}$ }
$(\exists v :: u \stackrel{n}{\rightarrow} v \wedge v \rightarrow w)$
$\Leftrightarrow$ { Induction Hypothesis }
$(\exists v ::$ "u is connected to v by a path of length n" $\wedge\ v \rightarrow w)$
$\Leftrightarrow$ { definition of connected }
$(\exists v :: (\exists s :$ "s is a path of length n" $: u = s_0 \wedge s_n = v) \wedge v \rightarrow w)$
$\Leftrightarrow$ { $\wedge$ over $\exists$ }
$(\exists v :: (\exists s :$ "s is a path of length n" $: u = s_0 \wedge s_n = v \wedge v \rightarrow w))$
$\Leftrightarrow$ { dummy unnesting }
$(\exists s, v :$ "s is a path of length n" $: u = s_0 \wedge s_n = v \wedge v \rightarrow w)$
$\Leftrightarrow$ { if s is a path of length n then $s +\!\!+ [v, w]$ is a path of length $n{+}1$:

dummy transformation }

$(\exists t : \text{"}t \text{ is a path of length } n{+}1\text{"} : u = t_0 \wedge t_{n+1} = w\,)$

$\Leftrightarrow$ { definition of connected }

"u is connected to w by a path of length $n{+}1$" .

□

In a very similar way we can prove that the relation "is connected by a non-empty path length" is equivalent to $\overset{+}{\rightarrow}$. Moreover, the proof of the above lemma does not depend on particular properties of the directed relation $\rightarrow$: the lemma and its proof also are valid for undirected graphs, provided, of course, we replace $\overset{*}{\rightarrow}$ and $\overset{+}{\rightarrow}$ by $\overset{*}{\sim}$ and $\overset{+}{\sim}$ respectively.

* * *

Note that being connected in an undirected graph is a symmetric relation: u is connected to v if and only if v is connected to u, because $[\,s_0, \cdots, s_n\,]$ is a path from u to v if and only if the *reverse* of s, that is, the sequence $[\,s_n, \cdots, s_0\,]$, is a path from v to u.

In directed graphs, being connected is not necessarily symmetric, of course: the existence of a *directed path* (usually) does not imply the existence of directed path in the reverse direction.

2.5.2 Path concatenation

Let s be a directed path of length m from node u to node v, and let t be a directed path of length n from node v to node w. So, the end point of s, which is v, equals the starting point of t, that is, we have $s_m = t_0$.

From s and t we can now construct a directed path, of length $m{+}n$, from node u to node w; this is called the "concatenation" of s and t, and we denote it by $s \mathbin{+\!\!+} t$. For s and t paths of length m and n, respectively, their concatenation $s \mathbin{+\!\!+} t$ is a path of length $m{+}n$, defined as follows:

$$\begin{aligned} (s \mathbin{+\!\!+} t)_i &= s_i \quad , \text{for } 0 \le i \le m \\ (s \mathbin{+\!\!+} t)_{m+i} &= t_i \quad , \text{for } 0 \le i \le n \end{aligned}$$

Keep in mind that $s \mathbin{+\!\!+} t$ is defined *only if* $s_m = t_0$, and this is implied by this definition: on the one hand $(s \mathbin{+\!\!+} t)_m = s_m$, on the other hand $(s \mathbin{+\!\!+} t)_m = t_0$. In this case, $s \mathbin{+\!\!+} t$ is a path from u to w indeed. This we prove as follows:

$(s \mathbin{+\!\!+} t)_0$

$=$ { definition of $\mathbin{+\!\!+}$ }

s_0

$=$ { s is a path from u to v }

u ,

as required; and, for $0 \le i < m$:

$(s +\!\!+ t)_i \rightarrow (s +\!\!+ t)_{i+1}$
$=$ { definition of $+\!\!+$ }
$s_i \rightarrow s_{i+1}$
$=$ { s is a path of length m }
true

as required; and, for $0 \le i < n$:

$(s +\!\!+ t)_{m+i} \rightarrow (s +\!\!+ t)_{m+i+1}$
$=$ { definition of $+\!\!+$ }
$t_i \rightarrow t_{i+1}$
$=$ { t is a path of length n }
true

as required; and, finally:

$(s +\!\!+ t)_{m+n}$
$=$ { definition of $+\!\!+$ }
t_n
$=$ { t is a path from v to w }
w ,

as required.

Concatenation of undirected paths is defined in exactly the same way: here concatenation is actually an operation on sequences of nodes, and the difference between $\rightarrow$ and $\sim$, that is, the difference between directed and undirected, only plays a role in the interpretation of such sequences as paths.

We now conclude that, both in directed and in undirected graphs, if a path s, say, exists from node u to node v and if a path t, say, exists from node v to node w, then also a path exists from node u to node v, namely $s +\!\!+ t$. Thus we have proved the following lemma.

2.3 Lemma. Both in directed and in undirected graphs, the relation "is connected to" is transitive.
□

2.5.3 The triangular inequality

Every path in a graph has a *length*, which is a natural number. Every non-empty set of natural numbers has a smallest element. Therefore, if node u is connected to node v we can speak of the *minimum* of the *lengths* of all paths from u to v. This we call the "distance" from u to v. Because, in undirected graphs, connectedness is symmetric, we have, in undirected graphs, that the distance from u to v is equal to the distance from v to u.

If u is *not* connected to v we define, for the sake of convenience, the distance from u to v to be ∞ ("infinity"), because ∞ can be considered, more or less, as the identity element of the minimum-operator. Note, however, that ∞ is not a natural number and that we must be very careful when attributing algebraic properties to it. For example, it is viable to define $\infty + n = \infty$, for every natural n, and even $\infty + \infty = \infty$, but $\infty - \infty$ cannot be defined in a meaningful way. An important property is:

(0) $n < \infty$, for all $n \in \mathsf{Nat}$;

(1) $n \leq \infty$, for all $n \in \mathsf{Nat} \cup \{\infty\}$.

We denote the distance from u to v by $dist(u,v)$. Then, function $dist$ is defined as follows, for all nodes u, v:

$$dist(u,v) = \infty \text{ , if } u \text{ is not connected to } v\text{;}$$
$$dist(u,v) = (\mathsf{min}\, n : 0 \leq n \wedge \text{“a path of length n exists from } u \text{ to } v\text{”} : n) \text{ , if } u \text{ is connected to } v\text{.}$$

Function $dist$ now satisfies what is known in Mathematics as the "triangular inequality". This lemma holds for both directed and undirected graphs.

2.4 Lemma. All nodes u, v, w satisfy: $dist(u,w) \leq dist(u,v) + dist(v,w)$.

Proof. By (unavoidable) case analysis. If $dist(u,v) = \infty$ or $dist(v,w) = \infty$ then also $dist(u,v) + dist(v,w) = \infty$; now, by property (1), we have $dist(u,w) \leq \infty$, so we conclude, for this case: $dist(u,w) \leq dist(u,v) + dist(v,w)$, as required.

Remains the case $dist(u,v) < \infty$ and $dist(v,w) < \infty$. In this case, paths exist from u to v and from v to w. Let s be a path, of length m, from u to v and let t be a path, of length n, from v to w. Then, as we have seen in the previous subsection, $s +\!\!+ t$ is a path, of length $m+n$, from u to w. By the definition of $dist$, we conclude: $dist(u,w) \leq m+n$. As this inequality is true for *all* such paths s and t, it is true for paths of minimal length as well. Hence, also for this case we have: $dist(u,w) \leq dist(u,v) + dist(v,w)$, as required.

□

2.5.4 Connected components

A directed graph $(V, \rightarrow)$ is *strongly connected* if every node is connected to every node, that is, if there is a directed path from every node u to every node v. In relational terms, this means that $\overset{*}{\rightarrow} = \top$. The adverb "strongly" stresses the fact that, in directed graphs, strong connectedness is a symmetric notion: for every two nodes u, v there is a path from u to v *and* there is path from v to u.

An undirected graph is *connected* if every pair of nodes is connected by a path. Relationally, a graph is connected if and only if $\overset{*}{\sim} = \top$. As we have seen, in undirected graphs connectedness is symmetric. It even is an equivalence relation. A *connected component* is a *maximal subset* of the nodes of the graph that is connected: the connected components of an undirected graph are the equivalence classes of $\overset{*}{\sim}$.

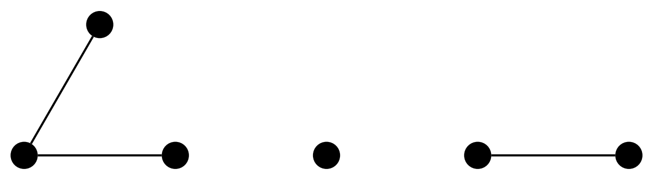

Figure 8: An undirected graph with 3 connected components

2.6 Cycles

A *cycle* in a graph is a non-trivial path from a node to itself. For undirected graphs a proper definition of 'non-trivial' needs some care. Generally, a graph may contain few cycles, many cycles, or no cycles at all. In the latter case the graph is called *acyclic*.

2.6.1 Directed cycles

In a directed graph a *cycle* is a (directed) path from a node to itself. For example, if $a \rightarrow b$ and $b \rightarrow a$ then the path $[\, a\,,\, b\,,\, a\,]$ is a cycle, and so is the path $[\, b\,,\, a\,,\, b\,]$. Although these are different paths they constitute, in a way, the same cycle. The simplest possible case of a directed cycle is $[\, a\,,\, a\,]$, namely if $a \rightarrow a$.

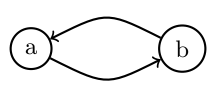

Figure 9: A simple directed cycle

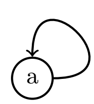

Figure 10: An even simpler cycle

2.6.2 Undirected cycles

In undirected graphs the notion of cycles is somewhat more complicated. For example, if, in undirected graph $(V, \sim)$, we have $a \sim b$ and, hence, also $b \sim a$, then $[\, a\,,\, b\,,\, a\,]$ is a path from node a to itself. Yet, we do not wish to consider this a cycle. More generally, we do not wish the pattern $[\ \cdots,\, a\,,\, b\,,\, a\,,\, \cdots\ ]$ to occur anywhere in a cycle: in a cycle, every next edge should be different from its predecessor. As a consequence, in an undirected graph the smallest possible cycle involves at least *three* nodes and *three* edges. Some texts require even stronger conditions for a path to be a cycle, for instance, that no node occurs more than once. We choose for the weakest version only excluding directly going back.

These considerations give rise to the following definition. An undirected cycle is a path $[\, s_0, \cdots, s_n\,]$, of length n, with the following additional properties:

$$3 \leq n$$

$$s_0 = s_n$$

$$(\forall i : 0 \leq i \leq n-2 : s_i \neq s_{i+2}\,) \ \wedge\ s_{n-1} \neq s_1$$

The first of these conditions expresses that a cycle comprises at least 3 nodes, the second condition expresses that the path's last node equals its first node – thus "closing the cycle" –, and the last condition precludes that every two successive edges in the cycle are different. The conjunct $s_{n-1} \neq s_1$ really is needed here: the "last" edge, $\{\, s_{n-1}, s_n\,\}$, which is the same as $\{\, s_{n-1}, s_0\,\}$, and the "first" edge, $\{\, s_0, s_1\,\}$, are successive too, which must be different as well.

In Figure 14, for example, we have that $[\, a, b, d, b, c, a\,]$ is not a cycle, because it contains edge $\{b, d\}$ twice in succession. Without the conjunct $s_{n-1} \neq s_1$, however, the path $[\, d, b, c, a, b, d\,]$ would be a cycle, which is undesirable: whether or not a certain collection of nodes constitutes a cycle should not depend on which node is the first node of the path representing that cycle.

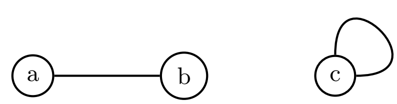

Figure 11: No cycles at all

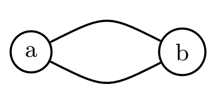

Figure 12: Not even an (undirected) graph

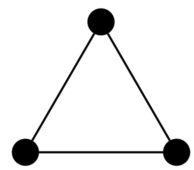

Figure 13: The smallest undirected cycle

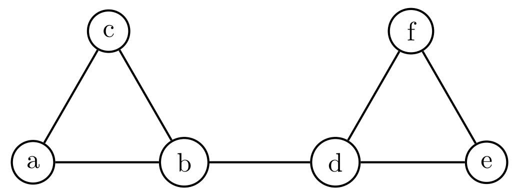

Figure 14: $[\,a,b,d,e,f,d,b,c,a\,]$ is a cycle, $[\,a,b,d,b,c,a\,]$ is not

Thus we obtain the following lemma, which expresses that cycles are "invariant under rotation". This lemma is useful because it allows us to let any node in a cycle be the starting node of the path representing that cycle.

2.5 Lemma. [Rotation Lemma] For every natural n, $3\leq n$, a path $[\,s_0, s_1, \cdots, s_{n-1}, s_0\,]$ is a cycle if and only if the path $[\,s_1, \cdots, s_{n-1}, s_0, s_1\,]$ is a cycle.
□

2.7 Euler and Hamilton cycles

2.7.1 Euler cycles

In a undirected graph a cycle with the property that it contains *every edge* of the graph exactly once is called an *Euler cycle*.

2.6 Theorem. For every connected graph $(V, \sim)$:

$$\text{"}(V,\sim) \text{ contains an Euler cycle"} \;\Leftrightarrow\; (\forall v : v\in V : \text{"}deg(v) \text{ is even"}) \quad .$$

Proof. By mutual implication.

"$\Rightarrow$": We consider an Euler cycle in graph $(V, \sim)$. Let v be a node. Wherever v occurs in the Euler cycle v has a predecessor u, say, in the cycle and a successor w, say, in the cycle. This means that u, v, w are all different and $u\sim v$ and $v\sim w$. Thus, all edges associated with v occurring in the Euler cycle occur in pairs; hence, the total number of edges associated with v occurring in the Euler cycle is even. Because the cycle is an Euler cycle *all* of v's edges occur in the Euler cycle; hence, $deg(v)$ is even.

"$\Leftarrow$": Assuming $(\forall v : v \in V : \text{"}deg(v) \text{ is even"})$ we prove the existence of an Euler cycle by sketching an algorithm for the construction of an Euler cycle. This algorithm consists of two phases. In the first phase a collection of (one or more) cycles is formed such that *every* edge of the graph occurs exactly once in exactly one of these cycles. In the second phase, the cycles in this collection are combined into larger cycles, thus reducing the number of cycles in the collection while retaining the property that *every* edge of the graph occurs exactly once in exactly one of the cycles in the collection. As soon as this collection contains only one cycle, this one cycle is a Euler cycle.

first phase: Initially all edges are white. The property $(\forall v : v \in V : \text{"}deg(v) \text{ is even"})$ will remain valid for the subgraph formed by V and the white edges only: it is an invariant of this phase. Another invariant is that all red edges form a collection of cycles with the property that *every red edge* of the graph occurs exactly once in exactly one of these cycles. Initially this is true because there are no red edges: initially the collection of red cycles is empty. If, on the other hand, all edges are red the collection of red cycles comprises all edges of the graph, and the first phase terminates. As long as the graph contains at least one white edge, the following step is executed.

Select a white edge, $\{s_0, s_1\}$, say. Because $deg(s_1)$ is even, node s_1 has a neighbor s_2, say, that differs from s_0 and such that edge $\{s_1, s_2\}$ is white as well. Repeating this indefinitely yields an infinite sequence $s_{i:0 \leq i}$ of nodes, pairwise connected by white edges. As the graph is finite, this sequence contains a sub-path $[s_p, \cdots, s_q]$, for some p, q with $0 \leq p < q$, that is a cycle, comprising white edges only. Now all white edges in this cycle are turned red. Because, for every node in this cycle, its associated edges occur in pairs, the number of white edges associated with any node in this cycle is even and, as a result, the degree of all nodes remains even under reddening of the white edges in this cycle. This process is repeated as long as white edges exist. Because in a undirected graph every cycle contains at least 3 edges the number of white edges thus decreases (by at least 3), this first phase will not go on forever, and will end in a situation where no white edges exist any more.

second phase: The second phase terminates if the collection of red cycles contains only one cycle. As long as this collection contains at least two cycles it takes two cycles that have a node in common. This is always possible due to the assumption that the graph is connected: if for a cycle all nodes are on none of the other cycles, this cycle is isolated and does not admit a path to any node on the other cycles, contradicting connectedness. Now these two cycles with a node v in common can be joined to a single cycle: if the one cycle is a path s from v to v and the other cycle is a path t from v to v, then the concatenation of s and t is a cycle covering both original cycles. It satisfies the cycle condition since s and t are disjoint. In this way the total number of cycles decreases, while still every edge of the original graph occurs exactly once as an edge of one of the cycles. This process is repeated until only one cycle remains and all cycles have been glued together; by construction this cycle is an Euler cycle.
□

2.7.2 Hamilton cycles

In a (directed or undirected) graph a cycle with the property that it contains *every node* of the graph exactly once is called a *Hamilton cycle*.

A naive algorithm to compute whether a given graph contains a Hamilton cycle is conceptually simple: enumerate all cycles and check whether any of them is a Hamilton cycle. This naive algorithm is quite inefficient, of course, but really efficient algorithms are not (yet) known: the problem to decide whether a graph contains a Hamilton cycle is *NP-hard*, which in practice means that all algorithms will require an amount of computation time that grows exponentially with the size of the graph.

Notice the contrast in complexity between the notion of Euler and Hamilton cycles. On the one hand, Theorem 2.6 provides a simple algorithm to evaluate the existence of an Euler cycle −just calculate the degrees of the nodes−, and its proof contains a relatively straightforward algorithm for the construction of an Euler cycle. On the other hand, calculating the existence of an Hamilton cycle, let alone construction of one, is NP-hard.

Thus, two seemingly similar notions −Euler cycles and Hamilton cycles− happen to have essentially different properties.

2.7.3 A theorem on Hamilton cycles

We consider finite, undirected graphs, with at least 4 nodes. We present a theorem giving a sufficient condition for the existence of Hamilton cycles, namely if the graph contains "sufficiently many" edges. In our case the notion of "sufficiently many" and the theorem take the following shape.

2.7 Theorem. We consider an undirected graph with n nodes, $4 \leq n$. If, for every two unconnected nodes, the sum of their degrees is at least n, then the graph contains a Hamilton cycle.
□

To formalize this, let V be a (fixed) set of nodes, with $n = \#V$, $4 \leq n$. In what follows variables u, v, p, q range over V, with $p \neq q$. The set E of edges is variable; that is, as a function of E we define predicates P and H, as follows:

$$P(E) \;=\; (\forall u, v : u \neq v : \; \{u, v\} \notin E \;\Rightarrow\; deg(u) + deg(v) \geq n \;) \quad \text{, and:}$$

$$H(E) \;=\; \text{"graph } (V, E) \text{ contains a Hamilton cycle"} \quad .$$

Predicate P formalizes our particular version of "sufficiently many": P expresses that, for every two unconnected nodes, the sum of their degrees is at least n.

Both P and H are *monotonic*, as follows:

monotonicity: For all E and for any two nodes p, q:

$$P(E) \;\Rightarrow\; P(E \cup \{\{p, q\}\}) \quad \text{, and:}$$

$$H(E) \;\Rightarrow\; H(E \cup \{\{p, q\}\}) \quad .$$

□

In addition, for the extreme cases, the empty graph $\bot$ and the complete graph $\top$, we have:

$$\neg(P(\bot)) \ \wedge \ \neg(H(\bot)) \ \wedge \ P(\top) \ \wedge \ H(\top) \ .$$

The theorem now states that every graph satisfying predicate P contains at least one Hamilton cycle.

2.8 Theorem. $(\forall E :: P(E) \Rightarrow H(E))$.

□

We present two proofs for this theorem. These proofs are essentially the same, but they differ in their formulation. The crucial part in both proofs is the following:

Core Property: For set E of edges and for any two nodes p, q with $\neg(\{p,q\} \in E)$:

$$P(E) \wedge H(E \cup \{\{p,q\}\}) \ \Rightarrow \ H(E) \ .$$

□

Notice that the Core Property also holds if $\{p,q\} \in E$, but in a trivial way only: then $H(E \cup \{\{p,q\}\}) = H(E)$, so in this case the property is void.

We will present a proof for the Core Property later, but first we will show how it is used in the proofs of the Theorem.

2.7.4 A proof by contradiction

The first proof runs as follows, by contradiction. That is, we suppose that the Theorem is false. Then, there exists a set F of edges such that $P(F)$ and $\neg(H(F))$. Because $\neg(H(F))$ and $H(\top)$, and because $F \subseteq \top$, there also exists a "turning point", that is, a set E of edges and a pair p, q of nodes such that:

$$F \subseteq E \ \wedge \ \neg(H(E)) \ \wedge \ H(E \cup \{\{p,q\}\}) \ .$$

Notice that, because of $\neg(H(E)) \wedge H(E \cup \{\{p,q\}\})$, we have −Leibniz!− that $E \neq E \cup \{\{p,q\}\}$, hence $\{p,q\} \notin E$.

Because of the monotonicity of P, and because $P(F)$ and $F \subseteq E$, set E satisfies $P(E)$ too. Now, from $P(E)$ and $H(E \cup \{\{p,q\}\})$ we conclude, using the Core Property, $H(E)$. In conjunction with the assumed $\neg(H(E))$ we obtain the desired contradiction.

2.7.5 A more explicit proof

The reasoning in the previous proof is somewhat strange: the assumption $\neg(H(E))$ is not really used in the proof proper: it is only used to conclude a contradiction. Therefore, we should be able to construct a more direct proof. In addition what does "there exists a 'turning point'" really mean, mathematically speaking?

Because of the monotonicity of P we have $P(E) \Rightarrow P(E \cup \{\{p,q\}\})$; therefore, by means of elementary propositional calculus, the Core Property can be rewritten thus:

$$(2) \qquad (P(E \cup \{\{p,q\}\}) \Rightarrow H(E \cup \{\{p,q\}\})) \;\Rightarrow\; (P(E) \Rightarrow H(E)) \quad ,$$

and this smells very strongly of a proof by Mathematical Induction. As a matter of fact, this *is* Mathematical Induction, albeit in a somewhat unusual direction, namely from larger towards smaller.

Firstly, we have $H(\top)$ −the complete graph contains a Hamilton cycle, very many even−, so we also have $P(\top) \Rightarrow H(\top)$. This is the basis of the induction.

Secondly, property (2) now represents the induction step. Because every set E of edges can be obtained from a larger set $E \cup \{\{p,q\}\}$, with $\{p,q\} \notin E$, we are done.

Notice that the fact that the collection of all possibles sets of edges is *finite*[3] is of no consequence: although usually applied to infinite sets the principle of Mathematical Induction is perfectly valid in a finite setting.

2.7.6 Proof of the Core Property

We repeat the Core Property, which is the essential part of both proofs of the Theorem.

Core Property: For set E of edges and for any two nodes p, q with $\{p,q\} \notin E$:

$$P(E) \wedge H(E \cup \{\{p,q\}\}) \;\Rightarrow\; H(E) \quad .$$

□

To prove this we assume that E is a set of edges and p, q are different nodes, such that $\{p,q\} \notin E$, satisfying $P(E)$ and $H(E \cup \{\{p,q\}\})$. The latter means that the graph $(V\,,\, E \cup \{\{p,q\}\})$ contains a Hamilton cycle. If such a Hamilton cycle does *not* contain edge $\{p,q\}$, then it also is a Hamilton cycle in the graph (V, E) ; hence, $P(E)$ and in this case we are done.

So, remains the case that $(V\,,\, E \cup \{\{p,q\}\})$ contains a Hamilton cycle that does contain edge $\{p,q\}$. Now we have to prove $P(E)$, that is, we must prove that (V, E) contains a Hamilton cycle as well, that is, *without* edge $\{p,q\}$.

For this purpose, let $[\, s_0, s_1, \cdots, s_n \,]$ be a Hamilton cycle in $(V\,,\, E \cup \{\{p,q\}\})$. This means that $\{ s_i \mid 0 \le i < n \} = V$ −recall that $n = \#V$−, that $s_n = s_0$, and that $(\forall i : 0 \le i < n : \{ s_i, s_{i+1} \} \in E \cup \{\{p,q\}\})$. We assume that this cycle contains edge $\{p,q\}$ and, without loss of generality, we assume that $s_0 = p$ and $s_1 = q$.
In this setting we prove that (V, E) contains a Hamilton cycle. To construct a Hamilton cycle *not* containing edge $\{p,q\}$ we take the Hamilton cycle introduced above, containing edge $\{p,q\}$, as a starting point. Removal of edge $\{p,q\}$ destroys the cycle, and what remains is a path connecting s_1, that is q, to s_0, that is p, that still contains all nodes of the graph and all edges of which are in E.

[3]for our *fixed*, finite set of nodes

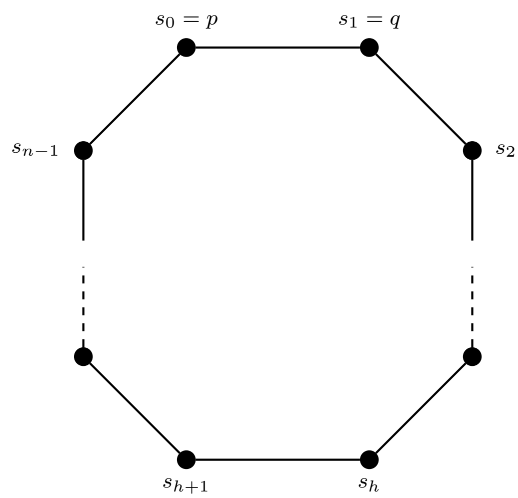

Figure 15: a Hamilton cycle, with edge $\{p, q\}$

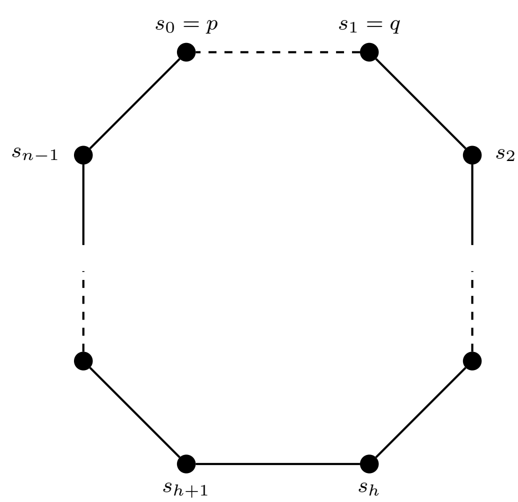

Figure 16: The remains of the cycle, after removal of edge $\{p, q\}$

Now we must restore the cycle by somehow reconnecting p and q, using edges in E only. We do so by selecting an index h in the interval $[2\,..\,n{-}1)$ such that both $\{p, s_h\} \in E$ and $\{q, s_{h+1}\} \in E$. To show that this is possible we need the theorem's assumption $P(E)$, which was defined as:

$$(\forall u, v : u \neq v : \ \{u, v\} \notin E \ \Rightarrow \ deg(u) + deg(v) \geq n \ ) \quad .$$

Applying this to p, q and using $\{p, q\} \notin E$ we obtain:

(3) $\qquad deg(p) + deg(q) \geq n \quad .$

Let $x \ = \ \#\{ i \in [2\,..\,n{-}1) \mid \{p, s_i\} \in E \}$ and let $y \ = \ \#\{ j \in [2\,..\,n{-}1) \mid \{q, s_{j+1}\} \in E \}$; now we calculate:

$deg(p) + deg(q) \ \geq \ n$

$\Leftrightarrow$ $\quad$ { definition of deg (twice), using $s_0 = p$ and $s_1 = q$ }

$\#\{ i \in [1\,..\,n) \mid \{s_0, s_i\} \in E \} + \#\{ j \in [2\,,\,n] \mid \{s_1, s_j\} \in E \} \ \geq \ n$

$\Leftrightarrow$ $\quad$ { split off $i = 1$ and $i = n{-}1$, using $\{s_0, s_1\} \notin E$ and $\{s_{n-1}, s_n\} \in E$ }

$1 + \#\{ i \in [2\,..\,n{-}1) \mid \{s_0, s_i\} \in E \} + \#\{ j \in [2\,,\,n] \mid \{s_1, s_j\} \in E \} \ \geq \ n$

$\Leftrightarrow$ $\quad$ { split off $j = 2$ and $j = n$, using $\{s_1, s_2\} \in E$ and $\{s_1, s_n\} \notin E$ }

$1 + \#\{ i \in [2\,..\,n{-}1) \mid \{s_0, s_i\} \in E \} + 1 + \#\{ j \in [3\,..\,n) \mid \{s_1, s_j\} \in E \} \ \geq \ n$

$\Leftrightarrow$ $\quad$ { dummy transformation $j := j{+}1$ }

$1 + \#\{ i \in [2\,..\,n{-}1) \mid \{s_0, s_i\} \in E \} + 1 + \#\{ j \in [2\,..\,n{-}1) \mid \{s_1, s_{j+1}\} \in E \} \ \geq \ n$

$\Leftrightarrow$ $\quad$ { definitions of x and y, using $s_0 = p$ and $s_1 = q$ }

$1 + x + 1 + y \ \geq \ n$

$\Leftrightarrow$ $\quad$ { calculus }

$x + y \ \geq \ n - 2 \quad .$

So, the number of indexes i in the interval $[2\,..\,n{-}1)$ for which $\{p, s_i\} \in E$ plus the number of indexes i in the range $[2\,..\,n{-}1)$ for which $\{q, s_{i+1}\} \in E$ is at least $n - 2$. The size of the interval $[2\,..\,n{-}1)$, however, only is $n - 3$; hence the two sets of indexes have a non-empty intersection: there exists an index h, $h \in [2\,..\,n{-}1)$, such that both $\{p, s_h\} \in E$ and $\{q, s_{h+1}\} \in E$.

For every such an index h, $[\, s_0, s_h, \cdots, s_2, s_1, s_{h+1}, \cdots, s_{n-1}, s_n \,]$ is a Hamilton cycle in the graph (V, E). Because we have shown the existence of such an h, we conclude the existence of a Hamilton cycle in (V, E), which was our goal.

remark: The existence of an index h in the interval $[2\,..\,n{-}1)$ implies that this interval is nonempty, that is, $2 < n{-}1$, which boils down to $4 \leq n$. Hence, the proofs of the Theorem presented here are only valid for graphs with at least 4 nodes. It can be easily verified that the Theorem also holds for $n = 3$: the complete 3-graph –a "triangle"– is the only one satisfying predicate P, and a "triangle" is a Hamilton cycle. For $n < 3$ the Theorem does not hold.

□

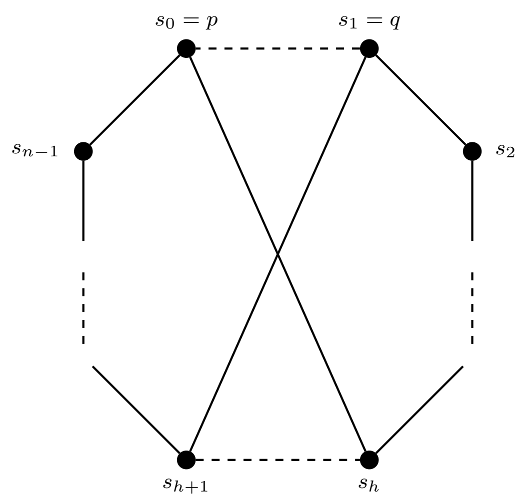

Figure 17: a Hamilton cycle, without edge $\{p, q\}$

2.8 Ramsey's theorem

2.8.1 Introduction

We are having a party at which every two guests either do know each other or do not know each other. If the number of guests at the party is "large enough" then the party has at least 5 guests all of which either do know one another or do not know one another. How large must the party be for this to be true?

F.P. Ramsey has developed some theory for the treatment of problems like this. This theory makes it possible to draw rather global conclusions about undirected graphs, independently of their actual structure.

To illustrate this we present a simple theorem that represents his work. In the above example the guests at the party can be considered the nodes of an undirected graph. Any two nodes are connected by an edge if and only if the two corresponding guests do know each other. A set of 5 guests all of which do know one another then amounts to a subgraph of size 5 that is *complete*, that is, we say that the whole graph *contains a complete* 5*-graph*. How do we formulate, on the other hand, that from a set of 5 guests every two guests do *not* know each other? Well, this means that the whole graph contains 5 nodes every two of which are *not* connected.

We might as well, however, consider the *complement* graph, in which two nodes are connected by an edge if and only if the two corresponding guests do *not* know each other. As a matter of fact, the problem as stated is *symmetric* in the notions of "knowing each other" and "not knowing each other". It is, therefore, awkward to destroy this symmetry by representing the one concept by the presence of edges and the other concept by their absence. Moreover, we have two possibilities here, the choice between which is irrelevant.

To restore the symmetry we, therefore, consider a complete undirected graph, of which the set of edges has been partitioned into two subsets − or more than two, in the more general case of Ramsey's theory −. The one subset then represents the pairs of guests who know each other and the other subset represents the pairs of guests who do not know each other.

Partitioning a set into (disjoint) subsets can be represented conveniently by *coloring*. In our case, partitioning the edges of a complete graph into two subsets can be represented by coloring each edge with one out of two colors. (And, of course, with more than two colors we can represent partitionings into more than two subsets.) Now, an edge of the one color may represent a pair of guests who do know each other, whereas an edge of the other color may represent a pair of guests who do not know each other. Thus, the symmetry between "knowing" and "not knowing" is restored.

2.8.2 Ramsey's theorem

We consider finite, complete, undirected graphs only. For the sake of brevity, we will use "k-graph" for the "complete k-graph", for any natural k, $2 \le k$. Formally, a coloring of a graph's edges is a function from the set of edges to the set of colors used, $\{\mathsf{red}, \mathsf{blue}\}$, say, if two colors are sufficient. So, a coloring is a function of type $E \rightarrow \{\mathsf{red}, \mathsf{blue}\}$, and if c is such a coloring, then for any edge $\{u, v\}$ we have either $c(\{u,v\}) = \mathsf{red}$ or $c(\{u,v\}) = \mathsf{blue}$, but not both simultaneously, as we presume that $\mathsf{red} \ne \mathsf{blue}$: every edge has only one color. In what follows we use variables c and d to denote colorings.

Again for brevity's sake, we say that the k-graph "contains a red m-graph" if the nodes of the k-graph contain a subset of m nodes such that all edges connecting these nodes are red; that is, these m nodes together with their edges constitute a completely red m-graph as a subgraph of the k-graph, for any k, m with $2 \le m \le k$.

As an example, notice that the 2-graph has two nodes only, connected by one single edge; hence, the proposition "the k-graph contains a red 2-graph" is equivalent to the proposition "the k-graph contains at least one red edge".

The proposition "the k-graph contains a red m-graph" depends on the parameters k and m, of course, but also on the actual coloring. So, it is a predicate with three parameters. Calling this predicate Rd, we define it as follows, together with a similar predicate Bl, for the color blue, for all k, c, m with $2 \le m \le k$:

$$Rd(k,c,m) \;\Leftrightarrow\; \text{"the } k\text{-graph with coloring } c \text{ contains a red } m\text{-graph"} \text{ , and:}$$

$$Bl(k,c,m) \;\Leftrightarrow\; \text{"the } k\text{-graph with coloring } c \text{ contains a blue } m\text{-graph"} \text{ .}$$

These predicates are monotonic, in the following way. Suppose $Rd(k,c,m)$, for some k, c, m. Then we also have $Rd(k{+}1, c, m)$, provided we consider the coloring of the $(k{+}1)$-graph as an extension of the coloring of the k-graph − both colorings being denoted here by the very same $c-$, just as the $(k+1)$-graph can be viewed as an extension of the k-graph. For this purpose we consider the $k+1$ nodes of the $(k+1)$-graph as a set of k nodes, forming a k-graph, plus one additional node, which may remain anonymous. Every coloring of the $(k+1)$-graph thus induces a coloring of the k-graph; as stated, function c denotes either coloring.

Ramsey's theorem now is about a function R, say, defined as follows, for all m, n with $2 \leq m$ and $2 \leq m$:

$$R(m,n) \;=\; \text{“the smallest of all natural numbers } k \text{ satisfying:} \\ (\forall c :: Rd(k,c,m) \vee Bl(k,c,n))\text{''} \quad .$$

The function value $R(m,n)$ is only well-defined, of course, if at least one natural number k exists satisfying $(\forall c :: Rd(k,c,m) \vee Bl(k,c,n))$: only then we can speak of the smallest such number. Notice that, by definition, if $R(m,n) = p$ then for every k, $p \leq k$, the k-graph contains at least one red m-graph or contains at least one blue n-graph (or both).

The following theorem states that such natural numbers exist and provides an upper bound for $R(m,n)$.

2.9 Theorem. (Ramsey)

$$R(m,n) \leq \binom{m+n-2}{m-1} \quad \text{, for all } m,n : 2 \leq m \wedge 2 \leq n \quad .$$

□

Notice that, by definition, $R(m,n)$ is symmetric in m and n, that is, we have: $R(m,n) = R(n,m)$, because for every coloring c satisfying $Rd(k,c,m) \vee Bl(k,c,n)$ a coloring d exists –which one?– satisfying $Rd(k,d,n) \vee Bl(k,d,m)$. The expression $\binom{m+n-2}{m-1}$ does not look symmetric, at least, not at first sight. Yet, it is, because binomial coefficients satisfy the following, general property:

$$\binom{m+n}{m} = \binom{m+n}{n} \quad \text{, for all } m,n : 1 \leq m \wedge 1 \leq n \quad ,$$

as a result of which we also have: $\binom{m+n-2}{m-1} = \binom{m+n-2}{n-1}$.

Proof of the Theorem: By Mathematical Induction on the value of $m+n$; that is, the Induction Hypothesis is:

$$R(p,q) \leq \binom{p+q-2}{p-1} \quad \text{, for all } p,q : 2 \leq p \wedge 2 \leq q \wedge p+q < m+n \quad .$$

We distinguish 3 cases.

Firstly, $2 \le m \wedge n = 2$: We consider the m-graph. Let c be the coloring in which all edges of the m-graph are red, so this particular c yields $Rd(m, c, m)$. For every *other* coloring c we have that not all edges of the m-graph are red, so the m-graph contains at least one blue edge, which means $Bl(m, c, 2)$, for all other c. Combining these cases we obtain $(\forall c :: Rd(m, c, m) \vee Bl(m, c, 2))$, from which we conclude that $R(m, n) \le m$. (Actually, we have $R(m, n) = m$ because no smaller graph contains a red m-graph, but the upper bound is all we need.) Now $m = \binom{m+2-2}{m-1}$, so we conclude $R(m, n) \le \binom{m+2-2}{m-1}$, as required.

Secondly, $m = 2 \wedge 2 \le n$: By symmetry with the previous case.

Thirdly, $3 \le m \wedge 3 \le n$: We need an additional property, to be proved later; the need of this property is inspired by a well-known property of binomial coefficients:

$$
\begin{array}{ll}
 & R(m, n) \\
\le & \quad \{ \bullet \text{ property of } R\text{, see below, using } 3 \le m \wedge 3 \le n \} \\
 & R(m-1, n) + R(m, n-1) \\
\le & \quad \{ \text{ Induction Hypothesis (twice) } \} \\
 & \binom{m+n-3}{m-2} + \binom{m+n-3}{m-1} \\
= & \quad \{ \text{ property of binomial coefficients } \} \\
 & \binom{m+n-2}{m-1} \quad .
\end{array}
$$

□

In the above proof of the Theorem we have used the following property of R, which constitutes the core of the proof.

property: $R(m, n) \le R(m-1, n) + R(m, n-1)$, for all $m, n : 3 \le m \wedge 3 \le n$.

proof: Let $k = R(m-1, n) + R(m, n-1)$. To prove that $R(m, n) \le k$ it suffices to prove $Rd(k, c, m) \vee Bl(k, c, n)$, for all colorings c. Therefore, let c be a coloring of the k-graph. Let v be a node of the k-graph and in what follows dummy u also ranges over the nodes of the k-graph. We define subsets X and Y of the nodes, as follows:

$$X = \{ u \in V \mid u \ne v \wedge c(\{u, v\}) = \mathsf{red} \} \text{ , and:}$$
$$Y = \{ u \in V \mid u \ne v \wedge c(\{u, v\}) = \mathsf{blue} \} \text{ .}$$

Then X and Y and $\{v\}$ partition the nodes of the k-graph, so we have:

$$\#X + \#Y + 1 \;=\; R(m{-}1\,,n) + R(m\,,n{-}1) \quad .$$

From this it can be derived that $R(m{-}1\,,n) \leq \#X \;\vee\; R(m\,,n{-}1) \leq \#Y$, by contraposition:

$$
\begin{array}{ll}
 & \#X < R(m{-}1\,,n) \;\;\wedge\;\; \#Y < R(m\,,n{-}1) \\
\Leftrightarrow & \quad \{ \text{ all values here are integers } \} \\
 & \#X \leq R(m{-}1\,,n) - 1 \;\;\wedge\;\; \#Y \leq R(m\,,n{-}1) - 1 \\
\Rightarrow & \quad \{ \text{ monotonicity of addition } \} \\
 & \#X + \#Y \;\leq\; R(m{-}1\,,n) + R(m\,,n{-}1) - 2 \\
\Leftrightarrow & \quad \{ \text{ all values here are integers } \} \\
 & \#X + \#Y + 1 \;<\; R(m{-}1\,,n) + R(m\,,n{-}1) \\
\Rightarrow & \quad \{ \; < \text{ is irreflexive } \} \\
 & \#X + \#Y + 1 \;\neq\; R(m{-}1\,,n) + R(m\,,n{-}1) \quad ,
\end{array}
$$

which settles the issue.

We now prove the required property, $Rd(k,c,m) \vee Bl(k,c,n)$, by distinguishing the two cases of this disjunction.

Case $R(m{-}1\,,n) \leq \#X$: From the definition of R we conclude, for our coloring c, that either $Rd(\#X,c\,,m{-}1)$ or $Bl(\#X,c\,,n)$. If $Rd(\#X,c\,,m{-}1)$ then X contains a red $(m{-}1)$-graph. By definition of X, we also have $c(\{u,v\}) = \mathsf{red}$, for all $u \in X$; hence, $X \cup \{v\}$ contains a red m-graph, which implies $Rd(k,c,m)$ as well. If, on the other hand, $Bl(\#X,c\,,n)$ then we also have $Bl(k,c,n)$. In both cases we have $Rd(k,c,m) \vee Bl(k,c,n)$, which concludes this case.

Case $R(m\,,n{-}1) \leq \#Y$: By symmetry.

□

2.8.3 A few applications

A party containing 5 guests all knowing one another or all not knowing one another can now represented as a complete graph containing a red 5-graph or a blue 5-graph. So, asking for the smallest such party is asking for the value of $R(5,5)$. As upper bound for $R(5,5)$, Ramsey's theorem now gives $\binom{8}{4}$, which equals 70. Further investigation of this problem has revealed that $R(5,5) \in [\,43\,,\,49\,]$; what is the actual value of R(5,5) still is an open problem!

*　　　*　　　*

The smallest complete graph containing, independently of the coloring, at least one *monochrome* triangle is the complete 6-graph. Ramsey's theorem yields $R(3,3) \leq 6$. For the complete 5-graph a coloring exists such that the graph does *not* contain a monochrome triangle; hence, $R(3,3) \geq 6$. So, $R(3,3) = 6$.

2.9 Trees

2.9.1 Undirected trees

An *(undirected) tree* is an undirected graph that is connected and acyclic. As we will see, on the one hand trees are the *smallest* connected graphs: removal of an edge from a tree always results in a graph that is not connected anymore. On the other hand, trees are the *largest* acyclic graphs: adding an additional edge to a tree results in a graph containing at least one cycle.

Although trees may be infinite, usually only finite trees are considered. Without further notice we confine our attention to finite trees.

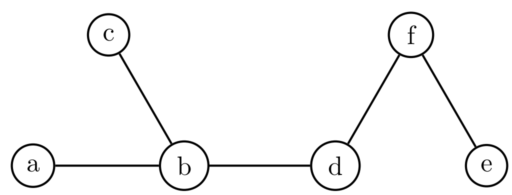

Figure 18: A (labeled) tree

In a connected graph, a *leaf* is a node with exactly one neighbor, that is, a node the degree of which equals 1. In Figure 18, for example, the leaves are a, c, e.

2.10 Lemma. Removal of a leaf and its (unique) associated edge from a connected graph yields a connected graph.

Proof. Let v be a leaf in a graph, and let u, w be nodes different from v. Then, no path connecting u and w contains v. Hence, every such path still exists in the graph resulting from removal of v and its associated edge.
□

2.11 Lemma. Every finite, acyclic, and connected graph with at least 2 nodes contains at least one leaf.

Proof. By contraposition: we prove that a finite, connected graph without leaves contains at least one cycle. Let $(V, \sim)$ be such a graph, with $\#V \geq 2$. Because the graph is connected every node has at least one neighbor, so $deg(v) \geq 1$, for every node v; because the graph contains no leaves we even have $deg(v) \geq 2$, for all v.

Now we construct an infinite sequence $s_{i:0 \leq i}$ of nodes, as follows. Choose $s_0 \in V$ arbitrarily, and choose $s_1 \in V$, such that $s_0 \sim s_1$. Note that this is possible because V is assumed to have at least 2 nodes, and because $deg(s_0) \geq 2$. Next, for all

$i, 0 \leq i$, we choose $s_{i+2} \in V$ such that $s_{i+1} \sim s_{i+2}$ and $s_i \neq s_{i+2}$. This is possible because $deg(s_{i+1}) \geq 2$, which follows from the assumption that the degree of every node is at least 2.

Thus, we have defined an infinite path s, starting at s_0 and with the property that $s_i \neq s_{i+2}$, for all $i, 0 \leq i$. The set V of nodes, however, is finite. Therefore[4], we have $s_p = s_q$, for some p, q with $0 \leq p < q$. Hence, the sub-path $[\, s_p, \cdots, s_q \,]$ is a cycle connecting s_p to itself, which concludes the proof.
□

A direct corollary of this lemma is that every tree with at least 2 nodes contains at least one leaf; after all, every tree is acyclic and connected.

2.12 Theorem. A tree with n, $1 \leq n$, nodes contains $n-1$ edges.

Proof. By Mathematical Induction on n. A tree with 1 node has 0 edges −after all, every edge connects two *different* nodes−, and $1-1=0$. Now let $(V, \sim)$ be a tree with $\#V = n+1$, where $1 \leq n$. By (the corollary to) the previous lemma this tree has a leaf u, say, so $deg(u) = 1$. Hence, there is exactly one node v, say, with $u \sim v$, so the one-and-only edge involving u is $\{u, v\}$. Now let $(W, \approx)$ be the graph obtained from $(V, \sim)$ by removal of leaf u and its edge $\{u, v\}$. This means that $W = V \backslash \{u\}$ and that $w \approx x \Leftrightarrow w \sim x$, for all $w, x \in W$.

Because $v \in V$ we have $\#W = \#V - 1$, so $\#W = n$. The graph $(W, \approx)$ is a tree because removal of node u and its edge $\{u, v\}$ maintains connectedness of the remaining graph and, obviously, introduces no cycles. By Induction Hypothesis, tree $(W, \approx)$ contains $n-1$ edges, hence the original tree $(V, \sim)$ contains n edges.
□

Actually, (finite) trees can be characterized in many different way. This is illustrated by the following theorem, of which the above theorem is a special case.

2.13 Theorem. For a connected, undirected graph (V, E) the following propositions are equivalent.

(a) (V, E) is acyclic.

(b) For every $e \in E$ the graph $(V, E \backslash \{e\})$ is not connected.

(c) $\#E = \#V - 1$.

(d) For all nodes u, v a *unique* path exists connecting u to v on which every node occurs at most once.

□

In addition the following properties deserve to be mentioned, as they are sometimes useful. Recall that, by definition, a graph is connected if every two nodes are connected by *at least* one path.

[4]See the discussion on finite and infinite, in the chapter on functions.

2.14 Lemma. An undirected graph is acyclic if and only if every two nodes are connected by *at most* one path.

2.15 Lemma. An undirected graph is a tree if and only if every two nodes are connected by *exactly* one path.

2.16 Lemma. For every undirected graph (V,E):

(a) $(\forall v : v{\in}V : deg(v) \geq 2) \;\Rightarrow\; \#E \geq \#V$

(b) "(V,E) is connected" $\Rightarrow\; \#E \geq \#V-1$

(c) "(V,E) is acyclic" $\Rightarrow\; \#E \leq \#V-1$

□

Notice that the latter two propositions provide another proof that if (V,E) is a tree then $\#E = \#V-1$. Also, notice that the last proposition can also, by contraposition, be formulated as:

$$\#E \geq \#V \;\Rightarrow\; \text{“}(V,E)\text{ contains a cycle”} \quad .$$

As a consequence, by combination with proposition (a) of this lemma, we obtain:

$$(\forall v : v{\in}V : deg(v) \geq 2) \;\Rightarrow\; \text{“}(V,E)\text{ contains a cycle”} \quad .$$

2.9.2 Rooted trees

A *rooted tree* is a tree in which one node is identified separately. This designated node is called the *root* of the tree. For every node in a rooted tree we can define its *distance* as the length of the unique path connecting that node and the root. Thus, for example, the root itself has distance 0, and for every two neighboring nodes, their distances differ by 1. In the latter case, the node with the smaller distance is *closer* to the root than the node with the larger distance. All edges in a rooted tree can now be turned into directed arrows, either by directing them *towards* the root or by directing them *away from* the root. The choice between these two options is rather irrelevant but must be made; therefore, in this text we adopt the convention that all arrows are directed away from the root. For example, Figure 16 gives the tree from Figure 15, as a rooted tree.

As an example of an infinite rooted tree, Figure 17 shows the tree obtained with the natural numbers as nodes, 0 as the root, and $\{\,(n\,,n{+}1) \mid 0{\leq}n\,\}$ as the (directed) edges. Such a linear arrangement hardly deserves to be called a "tree", of course, but formally it *is* a tree. Usually, however, such a linear arrangement is called a "list".

A more interesting example is obtained as follows. The nodes are the positive naturals, the root is 1, and, for positive m,n, the pair (n,m) is an arrow if and only if $2*n = m \,\vee\, 2*n{+}1 = m$. (This relation can also be formulated as $n = m\,\mathsf{div}\,2$.) The graph thus obtained is a rooted tree: via the arrows every positive natural number can be obtained from 1 in a unique way. See Figure 18.

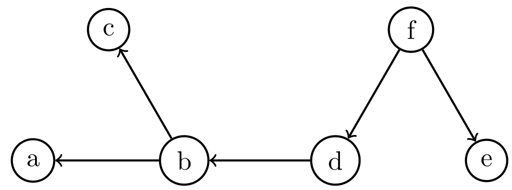

Figure 19: A rooted tree, with root f

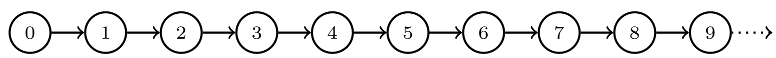

Figure 20: (Part of) the linear structure of the naturals

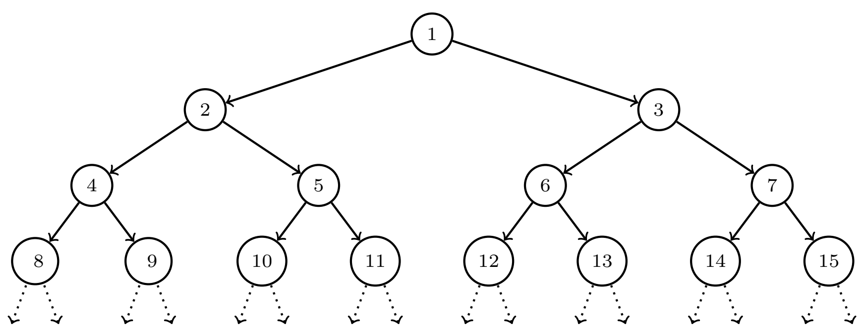

Figure 21: (Part of) the rooted, binary tree of the positive naturals

2.10 Exercises

1. How many edges does the complete undirected n-graph have, for all $n \geq 1$? Prove the correctness of your answer.

2. An undirected graph (V, E) is called "regular of degree d", for some natural number d, if $deg(v) = d$, for all $v \in V$. Prove that such a graph satisfies $d * \#V = 2 * \#E$.

3. For an undirected graph in which every node has degree 3, show that the total number of nodes is always even.

4. Let (V, E) be an undirected graph satisfying $\#V = 9$ and $\#E \geq 14$. Prove that V contains at least one node the degree of which is at least 4.

5. Given are two connected (undirected) graphs (V, E) and (W, F), such that $V \cap W = \emptyset$. Let $v \in V$ and $w \in W$. Prove that the graph $(V \cup W, \ E \cup F \cup \{\{v, w\}\})$ is connected.

6. Let (V, E) be a connected undirected graph, in which $V = \{v_1, v_2, \ldots, v_n\}$. Let $W = \{w_1, w_2, \ldots, w_n\}$. Prove that the undirected graph $(V \cup W, E')$ for E' defined by

$$E' = E \cup \{(v_i, w_i) \mid i = 1, 2, \ldots, n\}$$

   is connected.

7. Let $(V, \sim)$ be a connected undirected graph such that $v, w \in V$ with the following properties: $deg(v)$ and $deg(w)$ are odd and $deg(u)$ is even for all *other* nodes $u \in V$. Prove that the graph contains an Euler-path connecting v and w, that is, a path containing every edge of the graph exactly once.

8. For two connected undirected graphs (V_1, E_1) and (V_2, E_2) it is given that $V_1 \cap V_2 \neq \emptyset$. Prove that $(V_1 \cup V_2, E_1 \cup E_2)$ is connected.

9. Give an undirected graph having a cycle of length 3 and a cycle of length 4, but not a cycle of length 5.

10. For an undirected graph (V, E) every two non-empty subsets V_1 and V_2 of V satisfy:

$$(V_1 \cup V_2 = V) \Rightarrow (\exists v_1, v_2 : v_1 \in V_1 \wedge v_2 \in V_2 \wedge (v_1, v_2) \in E).$$

   Prove that (V, E) is connected.

   (Hint: for a node v consider $V_1 = \{u \in V \mid \text{ there is a path from } v \text{ to } u\}$.)

11. Let (V, E) be an acyclic undirected graph with two distinct nodes $v_1, v_2 \in V$ such that for every node $u \in V$ there is either a path from u to v_1 or a path from u to v_2, but not both. Prove that $\#E = \#V - 2$.

12. A *chain* in a directed graph $(V, \rightarrow)$ is an infinite sequence $s_{i\,:\,0 \leq i}$ of nodes −that is, a function of type $\mathbb{N} \rightarrow V$− with the property $(\forall i : 0 \leq i : s_i \rightarrow s_{i+1})$.

    (a) Prove that finite and acyclic directed graphs do not contain chains.

    (b) As a consequence, prove that every finite and acyclic directed graph contains at least one node the out-degree of which is zero.

13. We consider a (finite) undirected graph in which the degree of every node is at least 3. Prove that this graph contains a cycle containing at least 4 nodes.

    (Hint: construct v_i for $i = 0, 1, 2, 3, \ldots$ such that $v_{i-1} \rightarrow v_i$ and $v_i \neq v_{i-1}$ for $i > 0$ and $v_i \neq v_{i-2}$ for $i > 1$)

14. Let (V, E) be a finite connected undirected graph. Let W be a finite set and $f : V \rightarrow W$ a bijective function. Prove that the undirected graph

    $$(V \cup W, E \cup \{(v, f(v)) \mid v \in V\})$$

    is connected.

15. An undirected graph contains two cycles of lengths n, m, respectively, that have exactly one edge in common. Prove that the graph also has a cycle of length $n + m - 2$.

16. For an undirected graph (V, E) every two non-empty subsets V_1 and V_2 of V satisfy:

    $$(V_1 \cup V_2 = V) \Rightarrow (\exists v_1, v_2 : v_1 \in V_1 \wedge v_2 \in V_2 \wedge (v_1, v_2) \in E).$$

    Prove that (V, E) is connected.

    (Hint: for a node v consider $V_1 = \{u \in V \mid$ there is a path from v to $u\}$.)

17. We consider an (finite) undirected graph with n nodes, for $n \geq 3$. The degree of every node in this graph is at least 1 and the graph contains a node of degree $n-2$. Prove that this graph is connected.

18. We consider two undirected trees (V, E) and (W, F), with $V \cap W = \emptyset$. Let $v_0, v_1 \in V$ and $w_0, w_1 \in W$. Prove that the graph
    $(V \cup W\,,\ E \cup F \cup \{\{v_0, w_0\}, \{v_1, w_1\}\})$ contains a cycle.

19. Let (V, E) be an undirected tree, and $v \in V$. Choose $v' \notin V$ and define $V' = V \cup \{v'\}$ and $E' = E \cup \{\{v, v'\}\}$. Prove that (V', E') is a tree.

∗20. We consider the complete 6-graph in which every edge has been co loured either red or blue. Prove that, independent of the chosen coloring, this graph contains at least 2 *monochrome* triangles.

21. Prove that an undirected graph with n nodes and in which the number of edges is greater than $n^2/4$ contains at least one triangle.

22. Give an example of an undirected graph containing an Euler cycle, but not containing a Hamilton cycle.

23. Give an example of an undirected graph, with at least 4 nodes, and containing a cycle that is both an Euler cycle and a Hamilton cycle.

24. Give an example of an undirected graph containing an Euler cycle and a Hamilton cycle that are different.

25. For $i = 1, 2$ let (V_i, E_i) be a finite undirected graph for which $(v_i, v_i') \in E_i$ is an edge on a Hamilton cycle of (V_i, E_i). Assume $V_1 \cap V_2 = \emptyset$. Prove that the graph

    $$(V_1 \cup V_2, E_1 \cup E_2 \cup \{(v_1, v_2), (v_1', v_2')\})$$

    admits a Hamilton cycle.

26. Give an example of a Hamilton cycle in an undirected graph in which every node has degree 3.

27. Among any group of 21 people, show that there are four people of which either every two of them once played chess together or every two of them never played chess together. (Hint: use Ramsey theory.)

28. Give an example of a connected, undirected graph with 6 nodes, in which the degree of every node equals 3. Also give an example of such a graph in which the degree of every node equals 4.

29. Give an example of an undirected graph with 7 nodes, in which the degree of every node equals 2, and consisting of exactly 2 connected components.

30. Prove that every undirected graph, with 5 nodes and in which the degree of every node equals 2, is connected.

31. Prove that every acyclic, undirected graph, with n nodes and $n-1$ edges, is connected.

32. Prove that every undirected graph, with n nodes and at least $(n^2 - 3*n+4)/2$ edges, is connected.

$*$33. Prove that every undirected graph, with n nodes and at least $(n^2 - 3*n+6)/2$ edges, contains a Hamilton cycle.