Graph Theory Notes

Juan Alberto Rodríguez Velázquez
Departament d'Enginyeria Informàtica i Matemàtiques
Universitat Rovira i Virgili

On this notes

This material contains the lecture notes of the course on graph theory taught during the academic year 2023-2024 for students of the degree in Engineering Mathematics and Physics at the Universitat Rovira i Virgili.

Contents

Chapter 1

Graphs: Basic concepts

Configurations of nodes and connections between them often appear in different contexts. Formally, these configurations are combinatorial structures called graphs, which are used to represent “networks” of different types, such as electrical circuits, communication networks, organic molecules, social networks, etc.

1.1 Basic concepts

Definition 1. A *graph* $G=(V,E)$ is an ordered pair, where V is a non-empty finite set and E is a set of non-ordered pairs $\{u,v\}$ of elements belonging to V with $u \neq v$.

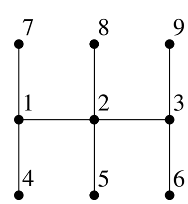
- The elements in V are called *vertices*, or *nodes*, of G and the elements in E are called *edges* of G.
- The *order* of G is the number of vertices and the *size* of G is the number of edges.

If there is no ambiguity, an edge $\{u,v\}$ can be denoted by uv and, in this case, we will write $uv \in E$ instead of $\{u,v\} \in E$. Notice that for every graph of order n and size m,

$$0 \leq m \leq \binom{n}{2}.$$

Example 1. The following figure shows a drawing of a graph $G=(V,E)$ where the vertex set is $V=\{1,2,3,4,5,6,7,8,9\}$ and the edge set is

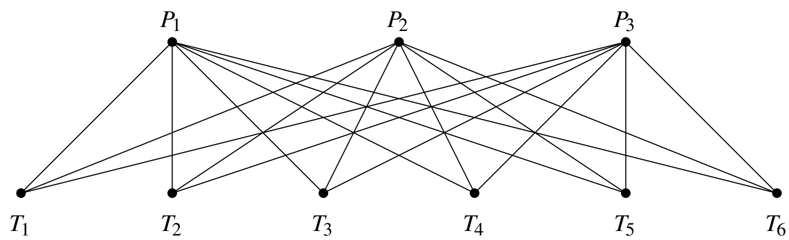
$$E=\{\{1,2\},\{1,4\},\{1,7\},\{2,5\},\{2,8\},\{2,3\},\{3,6\},\{3,9\}\}.$$



The order of G is $n=9$ and the size is $m=8$.

On occasion, as in the case of the previous example, the edges are represented with straight lines, but this is not necessary. The representation of the edges only has to indicate that there is connection between the corresponding vertices. That is, the concept of graph is a topological concept, and do not matter the form or the size that we give to the edges when representing them.

Example 2. Let us consider a company where there are various people, P_1,P_2,P_3, and a series of tasks that they have to do, $T_1,\ldots,T_6$, with the possibility that a person could do more than one task, depending on their capacities. The following figure shows a model in graph theory terms that represents the situation of individuals that could do different tasks, and tasks that could be done by different individuals.



The vertices and the edges can have additional attributes depending on the problem being modelled. These attributes could be colour, distance, cost, or any another attribute useful to the model in question. A *network* is a graph on which a set of additional attributes has been defined. For the moment we will define some basic concepts of the graphs that do not depend on additional attributes.

Definition 2. Two vertices $u,v \in V$ of a graph $G=(V,E)$ are *adjacent* if and only if the edge uv exists, that is, if and only if $uv \in E$. The adjacency of the vertices u, v is denoted by $u \sim v$. In this case it is said that the edge $a = uv$ joins or connects the vertices u and v, which are its *endpoints*. Other denominations that indicate that the vertices are adjacent can be:

- The vertices u, v and the edge uv are *incident*.
- The vertices u and v are *neighbours*.

Furthermore, it is said that two edges are *adjacent* if they share an endpoint.

Definition 3. The set of neighbours or *open neighbourhood* of a vertex v of a graph $G=(V,E)$ is given by

$$N(v) = \{u \in V:\ v \sim u\} = \{u \in V:\ \{u,v\} \in E\}.$$

The *degree* $\delta(v)$ of v is defined as the number of edges that are incident to v. That is,

$$\delta(v) = |N(v)|.$$

The vertices of degree zero are called *isolated vertices*.

Example 3. Let us consider the graph of Example 1. In this case $N(1) = \{2,4,7\}$, $N(3) = \{2,6,9\}$ and $N(2) = \{1,3,5,8\}$. Observe that $\delta(1) = \delta(3) = 3$, $\delta(2) = 4$ and the remaining vertices have degree one.

It is evident that for every vertex v of a graph of order n it follows that

$$0 \leq \delta(v) \leq n-1.$$

This observation can be used to refute the existence of some graphs.

Example 4. No graph with the sequence of degrees $2,2,2,3,3,4,8$ exists. In effect, if such a graph existed, say $G = (V,E)$, it would mean that $n = |V| = 7$ and, if there was a vertex v_0 of degree 8, in virtue of the previous inequality, $6 = n-1 \geq \delta(v_0) = 8$, which is impossible.

Exercise 1. Prove that any graph $G = (V,E)$ with a at least two vertices always has a minimum of two vertices of the same degree. In other words there is no graph of order greater than or equal to 2 with all the degrees different from each other.

Solution: Let $G = (V,E)$ be a graph of order n ($n \geq 2$). Since for each vertex $v \in V$ it follows that $0 \leq \delta(v) \leq n-1$, only degrees $0,1,\ldots,n-1$ can exist. As there cannot be vertices of degree 0 and $n-1$ at the same time, the degree sequence is in any of the following groups: $\{0,1,\ldots,n-2\}$ or $\{1,\ldots,n-1\}$. In any case, by the box principle, we can conclude that there are at least two vertices of the same degree. □

An interesting observation is that this exercise can be interpreted in terms of encounters and greetings between attendees at a meeting. We can state that in all meetings there is a minimum of two people that have made the same number of greetings (we are assuming that all meeting are attended by at least two people). This is easy to see using a model in graph theory terms: the vertices represent the meeting attendees and two vertices are adjacent if and only if the corresponding people greet each other. The degree of each vertex is the number of greetings by the corresponding person. To prove the statement, it is only necessary to apply the result shown in the previous exercise.

Exercise 2. Mr. Andrew and his wife invited four couples to a party. When everyone had arrived, some of the people in the room greeted (shook hands with) other people of the group. Of course, nobody shook their spouse's hand and nobody shook hands twice with another person. It's possible that there were people who did not greet anybody.

At the end, Mr. Andrew realized that none of his guests (his wife included) had greeted the same number of people. Use graph theory to interpret and answer each one of the following questions:

(a) Is it possible that Mr. Andrew also shook hands with a different number of people than the others?

(b) Is it possible that the Mr. Andrew only shook hands an odd number of times?

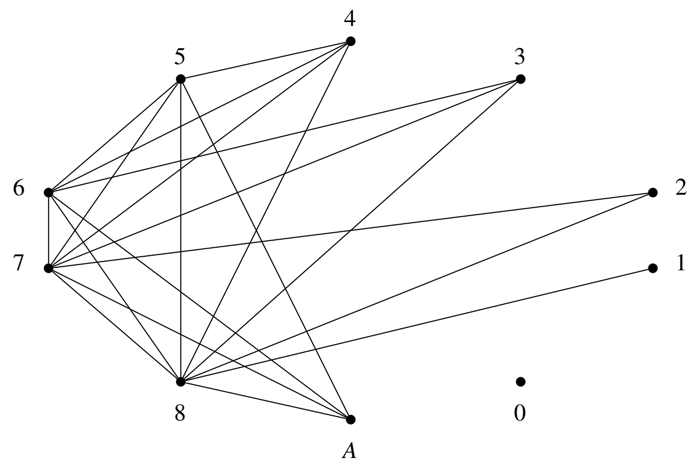
(c) Is there anyone who did not shake anybody's hand?

(d) How many times did Mr. Andrew shake hands? And Mrs. Andrew?

Solution: The graph G is the graph whose vertices are the people where two vertices are adjacent if and only if the corresponding people have greeted each other. As the order of the graph is $n = 10$ we see that the degrees belong to the set $\{0, 1, 2, ..., 8, 9\}$.

Now, if nobody shook hands with their spouse, no vertex can have degree 9. So the set of degrees is $\{0, 1, 2, ..., 8\}$.

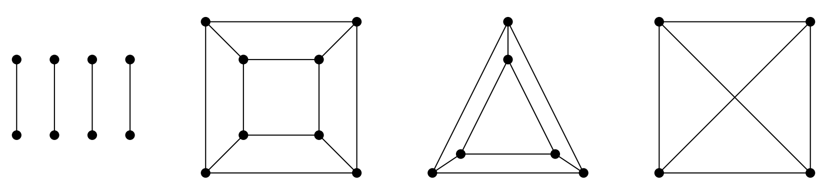
In this case the graph can be represented in the following way:



The couples are: $8-0$, $7-1$, $6-2$, $5-3$ and $4-A$. As a result, Mr and Mrs Andrew have greeted 4 people each and there is a person that did not shake anybody's hand. □

Definition 4. A graph is *regular* if all the vertices have the same degree; if the common degree is δ, then we say that the graph is δ-*regular*.

Example 5. The following graphs are regular.



1.2 Degree sum formula

In this sub-section we will study how the degrees of the vertices and the number of edges of the graph are related.

Proposition 1. (Degree sum formula) *For any graph $G = (V, E)$ of size m,*

$$m = \frac{1}{2} \sum_{v \in V} \delta(v).$$

That is, the sum of the degrees of the vertices of G equals two times the number of edges.

Proof. It is only necessary to count the number of edges of a graph from the edges that each vertex contributes. Each vertex v contributes $\delta(v)$ edges to the global computation, so that as a whole the number of edges would be $\sum_{v\in V}\delta(v)$. As each edge is shared by the two endpoint, each edge has been counted twice. Therefore, $2m=\sum_{v\in V}\delta(v)$. □

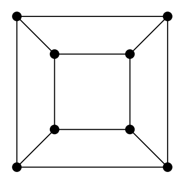
Example 6. Let us consider a graph with degree sequence $2,2,2,2,2,2,3,1$. According to the degree sum formula, the number of edges of the graph can be calculated easily without any other information:

$$m=\frac{1}{2}\sum_{v\in V}\delta(v)=\frac{1}{2}(2+2+2+2+2+2+3+1)=8.$$

The result is a direct consequence of the degree sum formula.

Corollary 2. *All δ-regular graphs of order n have a size of $m=\dfrac{n\delta}{2}$.*

Example 7. The cube is a 3-regular graph of order $n=8$ and size $m=\dfrac{3\cdot 8}{2}=12$.



Exercise 3. Does a 5-regular graph with an odd-number order exist?

Solution: The previous corollary gives us the answer to this question. In effect, if a 5-regular graph of an odd order existed, $n=2k+1$, then its measure would be $m=\frac{5(2k+1)}{2}=5k+\frac{5}{2}$, which is a contradiction. Therefore, graphs with this characteristic do not exist. □

Corollary 3. *In all graphs, the number of vertices of an odd degree is even.*

Proof. Let P be the set of vertices of even degree and let I be the set of vertices of odd degree. For every $v\in P$ there exists an integer k_v such that $\delta(v)=2k_v$, and for every $v\in I$ there exists an integer k_v such that $\delta(v)=2k_v+1$. According to the degree sum formula,

$$\begin{aligned}
2m &= \sum_{v\in P\cup I}\delta(v)\\
&=\sum_{v\in P}\delta(v)+\sum_{v\in I}\delta(v)\\
&=\sum_{v\in P}2k_v+\sum_{v\in I}(2k_v+1)\\
&=2\left(\sum_{v\in P}k_v+\sum_{v\in I}k_v\right)+\sum_{v\in I}1\\
&=2\left(\sum_{v\in P}k_v+\sum_{v\in I}k_v\right)+|I|.
\end{aligned}$$

It follows that $|I| = 2\left(m - \left(\sum_{v\in P} k_v + \sum_{v\in I} k_v\right)\right)$. □

Example 8. No graph exists with degree sequence $1,3,3,2,2,2,4$. In effect, if it existed, there would be an odd number of vertices of an odd degree, which contradicts the previous corollary.

Exercise 4. Given a graph $G = (V,E)$, if k_i is the number of vertices of degree i, indicate what relations there are between the following numerical values:

$$|E|, \quad |V|, \quad \sum_i ik_i, \quad \sum_i k_i.$$

Solution: The relations are

$$|V| = \sum_{i\geq 0} k_i \text{ and } 2|E| = \sum_{i\geq 0} ik_i.$$

□

Exercise 5. Let $G = (V,E)$ be a graph of order $n \geq 10$ such that all the vertices are of degree strictly greater than 5. Prove that the number of edges of the graph is greater than or equal to 30.

Solution: By hypothesis $\delta(v) \geq 6$, $\forall v \in V$. Now we can apply the degree sum formula:

$$2m = \sum_{v\in V} \delta(v) \geq \sum_{v\in V} 6 = 6|V| \geq 6\times 10 = 60$$

Therefore, the desired result follows. □

Exercise 6. A graph has order $n = 20$ and size $m = 62$. Every vertex has degree 3 or 7. How many vertices have degree 3?

Solution: Let x_3 and x_7 be the number of vertices of degree 3 and 7, respectively. Then we have $n = x_3 + x_7$ and $2m = 3x_3 + 7x_7$. Therefore, $x_3 = 4$ and $x_7 = 16$. □

Exercise 7. Let $G = (V,E)$ be a graph of n vertices, t of which are of degree k and the rest, of degree $k+1$. Show that $t = (k+1)n - 2m$, where m is the number of edges.

Solution: We can write $V = V_k \cup V_{k+1}$, where V_k is the set of vertices of degree k and V_{k+1} is the set of vertices of degree $k+1$. Applying the degree sum formula we obtain the relation $2m = k|V_k| + (k+1)|V_{k+1}| = kt + (k+1)(n-t)$, so that $t = (k+1)n - 2m$. □

Exercise 8. Let $G = (V,E)$ be a 3-regular graph of order n. Let $X \subseteq V$ such that $|X| = \frac{2n}{5}$ and each vertex of $V \setminus X$ has at least two neighbours in X. Prove that between the vertices of X there are no adjacencies.

Solution: Let c be the number of edges going from X to $V \setminus X$. Notice that $|V \setminus X| = |V| - |X| = n - \frac{2n}{5} = \frac{3n}{5}$. As each vertex in $V \setminus X$ has at least two neighbours in X, we deduce that $2 \cdot \frac{3n}{5} = 2|V \setminus X| \leq c$, and in addition, as G is a 3-regular graph, $c \leq 3|X| = 3 \cdot \frac{2n}{5}$. Thus, $c = \frac{6n}{5}$ and, therefore, each vertex in X has exactly three neighbours in $V \setminus X$. □

1.3 Some families of graphs

- *Empty Graphs*. The empty graph N_n of order $n \geq 1$ is the graph of n vertices and 0 edges; so that $N_n = (V, \varnothing)$, with $|V| = n$. The graph N_1 is called a *trivial graph*.

- *Cycles.* The cycle graph of order $n \geq 3$ is $C_n = (V, E)$, where $V = \{v_1, \ldots, v_n\}$ and

$$E = \{v_1v_2, v_2v_3, \ldots, v_{n-1}v_n, v_nv_1\}.$$

The cycles are 2-regular graphs and their order coincides with their size, $m = n$.

- *Path.* The path graph $P_n = (V, E)$ of order $n \geq 2$ is a graph that is defined by $V = \{v_1, \ldots, v_n\}$ and

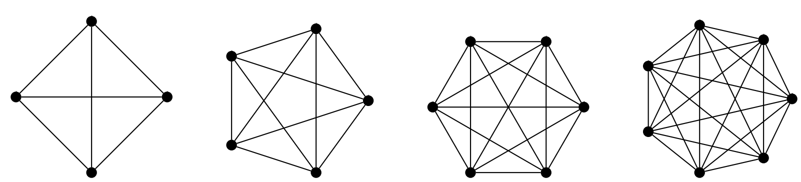
$$E = \{v_1v_2, v_2v_3, \ldots, v_{n-1}v_n\}.$$

The graph P_n can be obtained by the elimination of one edge of the cycle graph C_n. The graph P_n is not regular, has two vertices of degree 1, called path endpoints, and the other vertices have degree 2. Thus the relation between order and size is $m = n - 1$.

- *Complete Graphs.* The complete graph K_n is a graph of n vertices with all possible edges. That is, the complete graph is a $(n-1)$-regular graph of order n. Therefore,

$$m = \binom{n}{2} = \frac{1}{2}n(n-1).$$

Example 9. In the following figures we can observe the representation of complete graphs K_4, K_5, K_6 and K_7, respectively.

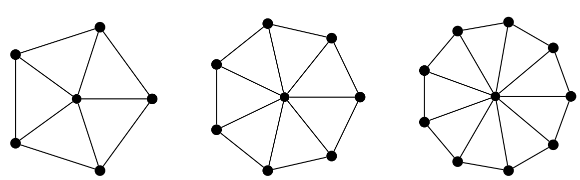


In particular, note that $K_1 = N_1$, $K_2 = P_2$ and $K_3 = C_3$.

- *Wheel graph.* A wheel graph W_n of order n $(n \geq 4)$ has an single vertex of degree $n-1$ and if this vertex and its incident edges are deleted, then we obtain a cycle of order $n-1$. The size of W_n is

$$m = 2(n-1).$$

Example 10. The following figure shows the wheel graphs W_6, W_8 and W_{10}, respectively.

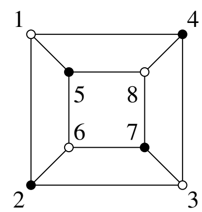


1.4 Bipartite graphs

Definition 5. A non-empty graph $G = (V,E)$ is *bipartite* if $V = V_1 \cup V_2$, with $V_1 \cap V_2 = \varnothing$, so that the existing edges only connect vertices in V_1 with vertices in V_2.

In a bipartite graph we can colour the vertices with two colours so that two adjacent vertices have different colours. It is easy to see that the path graphs P_n and the cycles of even order C_{2k} are bipartite graphs.

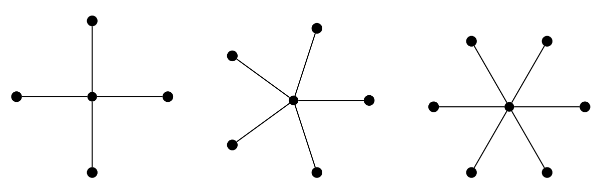
Example 11. The cube is a bipartite graph. In this case, $V_1 = \{1,3,6,8\}$ and $V_2 = \{2,4,5,7\}$.



Complete bipartite graphs: The *complete bipartite graph*, denoted by $K_{r,s} = (V_1 \cup V_2, E)$, is a bipartite graph where $|V_1| = r$, $|V_2| = s$, with all the possible edges connecting vertices in V_1 with vertices in V_2. Therefore, the order of $K_{r,s}$ is $n = r+s$ and its size is $m = r \cdot s$. The vertices in V_1 are all of degree s and those in V_2 are of degree r.

Star graphs: The *star graph* of order n ($n \geq 3$) is the complete bipartite graph $K_{1,n-1}$. The size of $K_{1,n-1}$ is $m = n-1$.

Example 12. This figure represents the stars $K_{1,4}$, $K_{1,5}$ and $K_{1,6}$, respectively.



Exercise 9. Show that if a bipartite graph $G = (V_1 \cup V_2, E)$ is regular, then $|V_1| = |V_2|$.

Solution: As each edge has an endpoint in V_1 and the other in V_2, then the size of G is $m = \sum_{v \in V_1} \delta(v)$ and also is $m = \sum_{v \in V_2} \delta(v)$. Thus, if G is a δ-regular graph, then $m = |V_1|\delta = |V_2|\delta$. Notice that our definition of bipartite graphs does not include the case of empty graphs. Therefore, $|V_1| = |V_2|$. □

Exercise 10. Among all bipartite graphs of order n, what would be the graph of maximum number of edges?

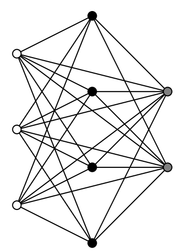
Solution: Let $G = (V_1 \cup V_2, E)$ be a bipartite graph. If x is the number of vertices in V_1, then $n - x$ will be the number of vertices in V_2. The total number of edges will be $m = x(n-x)$. Therefore, it is necessary to maximize the function $f(x) = x(n-x)$.

The solution will be $m = \frac{n^2}{4}$ if n is even and $m = \frac{n^2-1}{4}$ if n is odd. Hence, $G \cong K_{\frac{n}{2},\frac{n}{2}}$ for n even, and $G \cong K_{\frac{n-1}{2},\frac{n+1}{2}}$ for n odd. □

Complete multipartite graphs: The idea of bipartite graphs can be generalised to the *k-partites* graphs. In this case there is a partition $(V_1, \ldots, V_k)$ of the set of vertices, such that the edges connect vertices that belong to different sets of the partition and there are no edges connecting vertices of the same set V_i.

In the case of a complete *k-partite graph* there is a partition of the set of vertices in k subsets, of cardinal $n_1, n_2, \ldots, n_k$ respectively, and all possible edges exist, with the condition that there is no edge that connects vertices of a same subset. The corresponding graph is represented by $K_{n_1,\ldots,n_k}$.

Example 13. The illustration corresponds to the graph $K_{3,4,2}$.



Exercise 11. Consider a complete k-partite graph $K_{n_1,\ldots,n_k}$. What is the order and the size of the graph, and what are the degrees of the different vertices? For which values is the graph regular?

Solution: The order of the graph is $n = n_1 + n_2 + \cdots + n_k$. The size of the graph is

$$\frac{1}{2}(n_1(n-n_1) + n_2(n-n_2) + \ldots + n_k(n-n_k)).$$

The graph is regular if $n_i = \frac{n}{k}$, that is, when all the n_i are equal. Therefore, the order of the graph has to be divisible by k. □

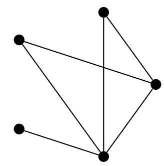
1.5 Graphic sequences. Havel-Hakimi Theorem

In the previous sub-sections we have seen some examples of graphs and their degree sequences. Now the question is if "given a sequence s of non-negative integers, would it be possible to construct a graph whose degree sequence was s". In Example 4 we saw a case where this is not possible. In addition, from the consequences of the degree sum formula it is also easy to realize that this is not always possible.

Definition 6. A sequence of non-negative integers, $s : d_1, d_2, \ldots, d_n$ is called a *graphic sequence* if there exists a graph $G = (V, E)$ of order n such that s is the degree sequence of G.

Example 14. The sequence $s : 4, 3, 2, 2, 1$ is a graphic sequence. It corresponds to the graph,

Example 15. The sequence $s : 4, 3, 3, 2, 1$ is not a graphic sequence, since it has an odd number of odd numbers (See Corollary 3).



From the definition of degree of a vertex and of the degree sum formula, two necessary conditions can be established so that a sequence $s : d_1, d_2, \ldots, d_n$ of integers is graphic:

(i) $0 \leq d_i \leq n-1$, for $1 \leq i \leq n$.

(ii) $\displaystyle\sum_{i=1}^{n} d_i$ has to be even.

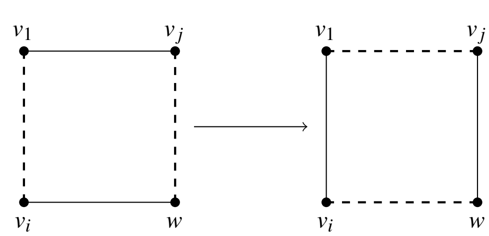
Now, these conditions are not sufficient. For instance, the sequence $5,5,3,3,3,1$ is not graphic, as for a graph of order $n=6$ it is no possible the existence of 2 vertices of degree $n-1=5$ and one vertex of degree 1. Obviously, in a non-complete graph the existence of k vertices of degree $n-1$ implies that the minimum degree is at least k.

Theorem 4 (Characterization of *Havel-Hakimi,* Havel (1955) and Hakimi (1961))**.** *A sequence $s : d_1, d_2, \ldots, d_n$ of non-negative integers, with $d_1 \geq d_2 \geq \ldots \geq d_n$, is a graphic sequence if and only if the sequence $d_2 - 1, d_3 - 1, \ldots, d_{d_1+1} - 1, d_{d_1+2}, \ldots, d_n$ is graphic.*

Proof. To test the sufficiency, let us assume that the sequence $s' : d_2 - 1, d_3 - 1, ..., d_{d_1+1} - 1, d_{k+2}, ..., d_n$ is graphic. Thus, a graph G' exists such that s' is its degree sequence and, therefore, d_1 vertices $v_2, v_3, ..., v_{d_1+1}$ of G' exist whose degrees are $d_2 - 1, d_3 - 1, ..., d_{d_1+1} - 1$, respectively. We can construct a new graph G from G' adding a new vertex v that will connect to the vertices $v_2, v_3, ..., v_{d_1+1}$. As the degree of v is d_1 and the degrees of $v_2, v_3, ..., v_{d_1+1}$ have increased by one, we can conclude that the degree sequence of G is $d_1, d_2, \ldots, d_n$.

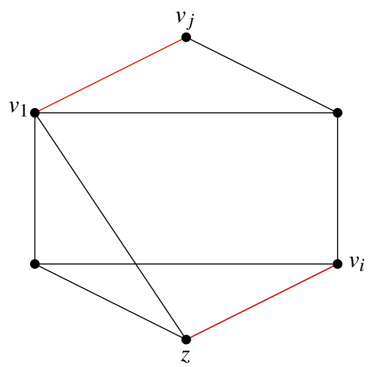
To test the necessity let us assume that $s : d_1, d_2, \ldots, d_n$ is an ordered graphic sequence, $d_1 \geq d_2 \geq \ldots \geq d_n$. Let G be a graph whose degree sequence is s. Let $D : d_2, d_3, \ldots, d_{d_1+1}$. We can consider a vertex v_1 of G of degree $\delta(v_1) = d_1$ and let D' be the degree sequence of the neighbours of v_1 ordered from greatest to least. If $D = D'$, then we simply delete from G the vertex v_1 and the edges adjacent to it and we get a graph G' whose degree sequence is $d_2 - 1, d_3 - 1, \ldots, d_{d_1+1} - 1, d_{d_1+2}, \ldots, d_n$.

We now assume that $D \neq D'$. As both sequences have d_1 numbers, there exists a number in D that is not in D' and vice versa. Thus, there exists a vertex $v_j \in N(v_1)$ of degree $\delta(v_j) \notin D$ and there exists $v_i \notin N(v_1)$ of degree $\delta(v_i) \in D$. As the sequences are ordered from greatest to least, $\delta(v_i) > \delta(v_j)$. Therefore, there exists $w \in N(v_i) \setminus N(v_j)$.

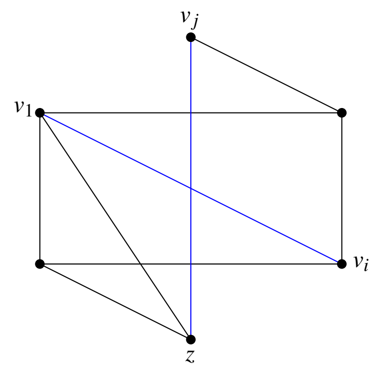


Now we modify the graph G by deleting edges v_1v_j and v_iw and adding edges v_1v_i and v_jw as shown in the previous figure. This operation does not change the degrees of the vertices of the graph but it increases by one the number of neighbours of one neighbour of v_1 whose degrees in G does not belong to D. We can repeat this process until we obtain a graph where the degrees of all the neighbours of v_1 are in D. By deleting vertex v_1 from this graph we get another graph G' whose degree sequence is $d_2-1, d_3-1, ..., d_{d_1+1}-1, d_{d_1+2}, \dots, d_n$. $\square$

Example 16. An example to show the case $D \neq D'$ in the previous proof.



Graph G.



Graph G modified.

The Havel-Hakimi theorem leads to a recursive algorithm that allows one to decide if a sequence is graphic.

Havel-Hakimi Algorithm:
Input: a sequence of integers $s: d_1, d_2, \dots, d_n$
Output: it tells if the sequence is graphic.
Algorithm:

If there exists $d_i > n-1$, then the sequence is not graphic, *end*.
While there is no $d_i < 0$ and s is not identically 0.
  Order s in descending order.
  Delete d_1 of s and subtract 1 unit from the following d_1 elements.
endwhile
If there exists $d_i < 0$, then the sequence is not graphic, *end*.
If the resulting sequence is identical to 0,
  then s is a graphic sequence. *end*

Example 17. The following table shows a simulation of the algorithm for the sequence s : $2,2,4,3,3,2,3,5$.

Iteration	Sequence	Operation
Initially	$2,2,4,3,3,2,3,5$	Initial sequence
	$5,4,3,3,3,2,2,2$	Order sequence
1	$3,2,2,2,1,2,2$	First sub-sequence
	$3,2,2,2,2,2,1$	Order sub-sequence
2	$1,1,1,2,2,1$	Second sub-sequence
	$2,2,1,1,1,1$	Order sub-sequences
3	$1,0,1,1,1$	Third sub-sequence
	$1,1,1,1,0$	Order sub-sequences
4	$0,1,1,0$	Fourth sub-sequence
	$1,1,0,0$	Order sub-sequences
5	$0,0,0$	Fifth sub-sequence
End		

Therefore, the sequence is graphic.

Exercise 12. Determine if the following sequences are graphic.

(1) $s : 5,5,7,6,4,2,4,5$

(2) $s : 2,3,4,5,6,7,8,8,8,8,8,8,8,8,8,8$

(3) $s : 6,6,4,4,1,4,3$

Solution:

(1) It is a graphic sequence, and applying the algorithm of Havel-Hakimi we get:

Iteration	Sequence	Operation
Initially	5,5,7,6,4,2,4,5	
	7,6,5,5,5,4,4,2	Order sequence
1	5,4,4,4,3,3,1	First sub-sequence
2	3,3,3,2,2,1	Second sub-sequence
3	2,2,1,2,1	Third sub-sequence
	2,2,2,1,1	Alternated sub-sequences
4	1,1,1,1	Fourth sub-sequence
5	0,1,1	Fifth sub-sequence
	1,1,0	Alternated sub-sequences
6	0,0	Sixth sub-sequence
7	0	Seventh sub-sequence
End		

(2) It is not graphic since it has three odd numbers.

(3) Not a graphic sequence. Applying the Havel-Hakimi Algorithm we obtain:

$6,6,4,4,1,4,3$

$6,6,4,4,4,3,1$

$5,3,3,3,2,0$

$2,2,2,1,-1$

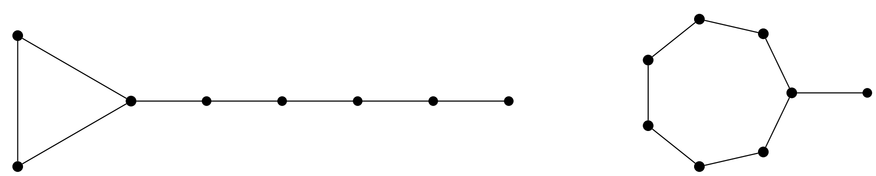
□

Exercise 13. Give examples of two graphs with different structures that have the same degree sequence.

Solution: The sequence $2,2,2,2,2,2$ corresponds to a cycle C_6 and also to the graph composed of two copies of the cycle C_3. □

Exercise 14. Show that the sequence of integers $2,2,2,2,2,2,2,3,1$ is the degree sequence of a graph. Propose two different examples of graphs that have this degree sequence.

Solution: The Havel-Hakimi Algorithm would be applied to show that the sequence is graphic. Two possible graphs that have this degree sequence are:



□

Exercise 15. Consider the sequence $d-2, d-2, d-1, d-1, d-1, d-1, d+2$, where d is an integer. For which values of d is the sequence graphic?

Solution: As the degree of the vertices cannot be negative, it has to obey $d \geq 2$; in addition, as the sequence has 7 numbers, it has to obey $d+2 \leq 6$. Therefore $d \in \{2,3,4\}$.

For $d=2$, the sequence $0,0,1,1,1,1,4$ is graphic since applying the Havel-Hakimi algorithm it returns the sequence $0,0,0,0,0,0$.

For $d=3$ the sequence is $1,1,2,2,2,2,5$, which is not graphic due to having an odd number of odd numbers.

Finally, for $d=4$ the sequence is $2,2,3,3,3,3,6$ and the Havel-Hakimi algorithm gives:
$6,3,3,3,3,2,2$
$2,2,2,2,1,1$
$1,1,2,1,1$
$2,1,1,1,1$
$0,0,1,1$
$1,1,0,0$
$0,0,0$
Therefore, the sequence is graphic. □

Exercise 16. Study if there can be graphs with the following degree sequences:

(a) 3, 3, 3, 3, 3, 4, 4

(b) 6, 3, 3, 2, 2, 2

(c) 4, 4, 3, 2, 1

Solution:

(a) No, because there is an odd number of odd numbers.

(b) No, because it does not obey the relation $0 \leq \delta(v) \leq n-1$.

(c) No, by applying the Havel-Hakimi algorithm. □

Exercise 17. Determine the values of n and d such that the sequence

$$d, d+1, d+2, \ldots, d+n-1$$

is graphic.

Solution: As the sequence has n numbers and the maximum degree is less than or equal to $n-1$, the only possibility is $d=0$ and the sequence is

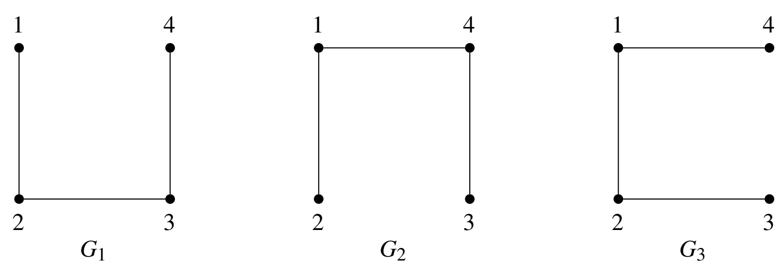
$$0, 1, 2, \ldots, n-1.$$

Now, for $n \geq 2$ there cannot be a vertex of degree 0 and another of degree $n-1$. Therefore, the sequence cannot be graphic, except in the case $n=1$ which would correspond to a trivial graph K_1. □

1.6 Isomorphic graphs

In this subsection we will consider the problem of deciding when two graphs are structurally equivalent, this is, when they have the same patterns of connection. If two graphs have a small order and are given by their graphic representation, on occasion it is easy to decide if both constitute different representations of the same graph. Although, in general, a practical algorithm does not exist that allows one to decide if two graphs of the same order have the same structure or not. This problem of decision is known as a "graph-isomorphism problem."

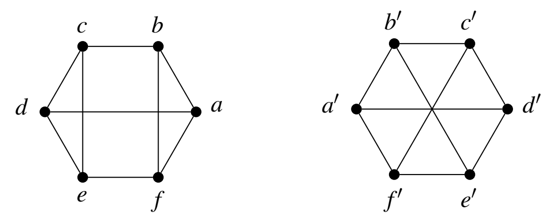
Example 18. Consider the following graphs.



They all have the same set of vertices $V = \{1,2,3,4\}$ and, in addition, the sets of edges are $E_1\{\{1,2\},\{2,3\},\{3,4\}\}$, $E_2 = \{\{1,2\},\{1,4\},\{3,4\}\}$, $E_3 = \{\{1,2\},\{2,3\},\{1,4\}\}$. These graphs are different given that they are the respective sets of edges. Despite this, the three graphs have the same structure.

Even though the structure of two graphs might be the same, the labelling cannot be irrelevant; it can be key for some models and applications. For example, if a graph is the model for a communications project and the vertices of the graph are concrete populations, then it is of great practical importance to know which vertices are connected to each other.

Example 19. Although the graphs in the following figure have the same order, the same size and both are 3-regular, they do not have the same structure. It is easy to check that the one on the left has triangles while the one on the right does not.



By means of the following definition we can formalise when two graphs are structurally equivalent.

Definition 7. Let $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ be two graphs.

- G_1 and G_2 are *identical* if and only if $V_1 = V_2$ and $E_1 = E_2$.
- G_1 and G_2 are *isomorphic* (denoted $G_1 \cong G_2$) if and only if there exists a bijection

$$\varphi : V_1 \to V_2$$

  that preserves the adjacencies and the non-adjacencies, that is,

$$u \sim v \Leftrightarrow \varphi(u) \sim \varphi(v).$$

  In this case, it is said that φ is a graph *isomorphism*.

In general, the brute-force search for all the possible isomorphisms between two graphs is not feasible. For two graphs of order n there are $n!$ bijective applications between the sets of vertices. Therefore, it is necessary to identify some "parameters" of the graphs that, in some cases, allow us to discard the existence of isomorphism. These parameters or properties are called invariants.

Definition 8. A *graph invariant* is a property of the graph that is preserved by isomorphisms.

As the number of vertices, the number of edges and the degree sequences are the same for any pair of isomorph graphs, these are graph invariants. Therefore, we can establish some necessary conditions of isomorphism.

Observation 5. *If an isomorphism $\varphi : V_1 \to V_2$ of the graphs $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ exists, then:*

(1) *G_1 and G_2 have the same order.*

(2) *G_1 and G_2 have the same size.*

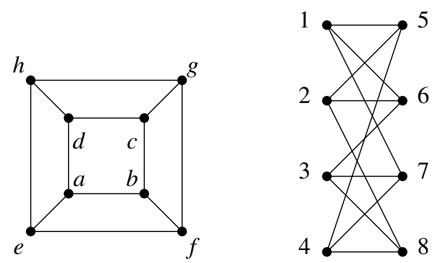
(3) *$\delta(x) = \delta(\varphi(x))$ for every $x \in V_1$.*

In many cases these conditions allow one to ensure that two graphs are not isomorphs. Now, as we can see in Example 19, these conditions are not sufficient for two graphs to be isomorphic. In this example the invariant used for discounting the isomorphism was the number of triangles in the graphs.

To the extent that we incorporate new concepts and study new properties of the graphs, we will continue to incorporate new invariants that are useful in practice for discounting graph isomorphisms.

The following exercise shows us an example of graph isomorphism.

Exercise 18. Determine if the graphs represented in the figure are isomorphic.

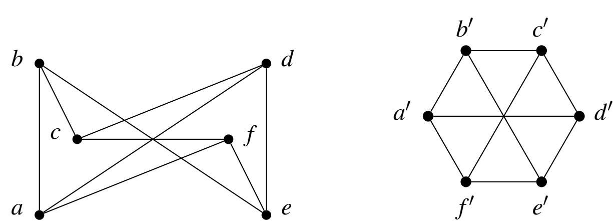


Solution: The answer is affirmative and the isomorphism is:

$$\begin{array}{l} a \to 1,\ \ b \to 5 \\ c \to 2,\ \ d \to 6 \\ h \to 3,\ \ e \to 7 \\ f \to 4,\ \ g \to 8 \end{array}$$

□

Exercise 19. Determine if the following graphs are isomorphic.



Solution: The graphs are isomorphic and the isomorphism is:

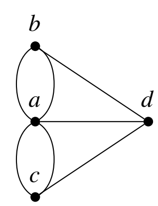
$$\begin{array}{l} a \to a',\ \ d \to d' \\ b \to b',\ \ e \to e' \\ c \to c',\ \ f \to f' \end{array}$$

Note that both graphs are isomorphic to the complete bipartite graph $K_{3,3}$. □

1.7 Multigraphs, pseudographs, hypergraphs and digraphs

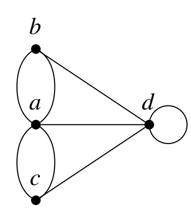
Definition 9. A *multigraph* $G = (V,E)$ is an ordered pair where V is a non-empty finite set and E is a multi-set of non-ordered pairs $\{u,v\}$ of elements of V with $u \neq v$. In other words, a *multigraph* is a graph that admits multiple edges.

Example 20. In the following multigraph there are two edges connecting the vertices a and b, and two edges connecting a and c. This multigraph could serve as a model of a network of roads where there are two different roads that connect village a with village b and another two that connect a with c. In this case $G = (V,E)$ where $V = \{a,b,c,d\}$ and $E = \{ab,ab,ac,ac,ad,bd,cd\}$.



Definition 10. A *pseudograph* $G = (V,E)$ is an ordered pair where V is a non-empty finite set and E is a multi-set of non-ordered pairs $\{u,v\}$ of elements of V.

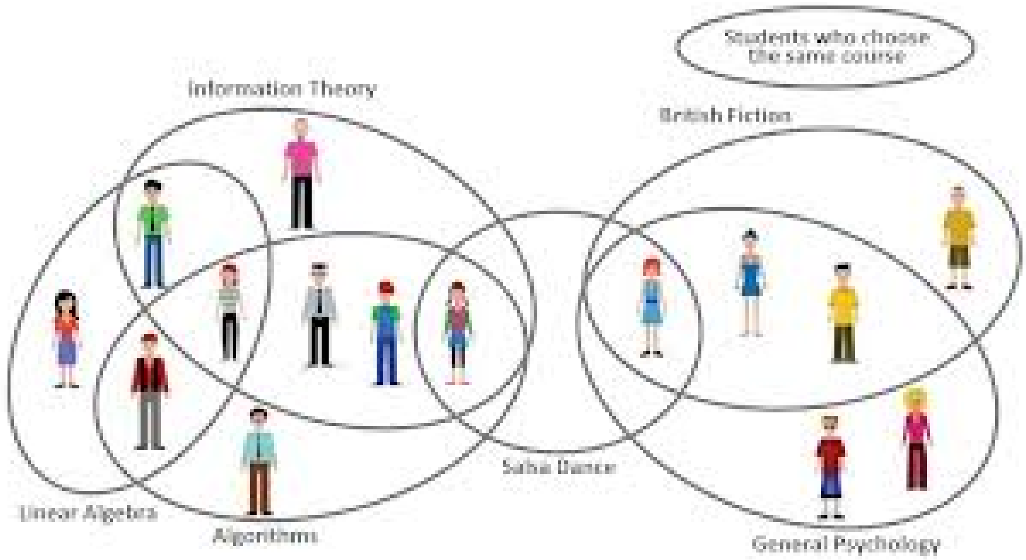
Note that a pseudograph is a multigraph that admits edges of type $\{u,u\}$ connecting a vertex with itself (*loops*), possibly in multiple forms, and also multiple edges between pairs of vertices.



Example 21. One example of loop is the edge $\{d,d\}$ of the pseudograph of the previous figure. The degrees of the vertices are counted according to the original definition; so each loop increases the degree of the vertex by 2. In this case the degree of the vertex d is $\delta(d) = 5$.

Definition 11. A *hypergraph* $G = (V,E)$ is an ordered pair where V is a non-empty finite set and the elements of E are non-empty subsets of V called *hyperedges*.

Example 22. In the following example of hypergraph each node is a student and the hyperedges are the groups of students that choose a specific subject.

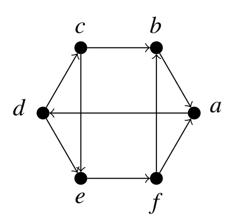


The *simple graphs* are those that correspond to the original definition, by comparison with these new extensions.

There are applications in which the situations are not described properly if orientations or directions of travel are not assigned to the edges of the graph; this leads to *oriented* graphs, also called *directed* or *digraphs*.

Definition 12. A *digraph* or *directed graph* $G = (V,E)$ is an ordered pair where V is a non-empty finite set and $E \subseteq V \times V$. That is, the edges are ordered pairs of elements of V called *arcs*.

Example 23. The set of arcs of the following digraph is:



$$E = \{(c,b),(b,a),(a,d),(d,c),(d,e),(e,f),(f,a),(f,b),(c,e)\}.$$

In a digraph $G = (V,E)$, an arc of the form (u,u) is called an oriented loop. In addition, given an arc $a = (u,v)$, vertex u is called the *origin* and v is called *the endpoint*. Similarly to the case of graphs, the *order* of the digraph is the number of vertices and the *size* is the number of arcs.

For each vertex $v \in V$ of the digraph we define $g^+(v)$, *outdegree*, as the number of arcs that have the vertex v as their origin; or, in other words, the cardinal of the set $\{u \in V : (v,u) \in E\}$.

Analogously, the *indegree* $g^-(v)$ is the number of arcs whose endpoint is the vertex v or, similarly, the cardinal of the set $\{u \in V : (u,v) \in E\}$.

Given a digraph $G = (V, E)$, we define the *underlying graph* (V, E') so that $\{u, v\} \in E'$ if and only if $(u, v) \in E$ or $(v, u) \in E$.

We can also consider combinations of the previous concepts and hybrids, in which, for example, not all the edges are oriented.

Exercise 20. Propose a version for digraphs of the degree sum formula in terms of the indegrees and outdegrees of the vertices.

Solution: The size m of a digraph $G = (V, E)$ is calculated by means of the following formula:

$$2m = \sum_{v \in V} g^+(v) + \sum_{v \in V} g^-(v),$$

Or also,

$$m = \sum_{v \in V} g^+(v) = \sum_{v \in V} g^-(v).$$

□

1.8 Representation and storage of graphs

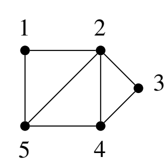
Neither the abstract representation nor the graphic representation of a graph are appropriate for describing a graph, if one wants to manipulate it by means of a computer program. We need to propose alternative methods for description and storage. First, it is always necessary to label the vertices. Next, matrices associated with the graph can be constructed, or the adjacency list.

Adjacency matrix

Definition 13 (Adjacency matrix)**.** Let $G = (V, E)$ be a graph and let $V = \{v_1, v_2, \ldots, v_n\}$. The *adjacency matrix* of G is defined as $A = (a_{ij})$ where

$$a_{ij} = \begin{cases} 1 & \text{if} \quad v_i \sim v_j; \\ 0 & \text{otherwise.} \end{cases}$$

Example 24. The adjacency matrix of the graph



is the following square matrix of order 5×5:

$$A = \begin{pmatrix} 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 & 0 \end{pmatrix}$$

Note that the adjacency matrix is symmetric and if we add the elements of a specific row we get the degree of the vertex corresponding to this row. Likewise, if we add all the elements of the matrix we get twice the number of the edges. Later we will see that the adjacency matrix allows us to obtain valuable information on the structure of the graph by applying methods of linear algebra.

In the case of multigraphs, pseudographs and digraphs, we must take into account that:

1. The elements of the diagonal could be positives if there are loops.
2. The matrix could contain values greater than one if there are multiple edges.
3. The matrix could be asymmetric in the case of digraphs.

Below we will establish the definition of adjacency matrix of a digraph.

Definition 14 (Adjacency matrix of a digraph)**.** Let $G = (V,E)$ be a digraph and let $V = \{v_1, v_2, ..., v_n\}$. The *adjacency matrix* of G is defined as $A = (a_{ij})$ where

$$a_{ij} = \begin{cases} 1 & \text{if} \quad (v_i, v_j) \in E; \\ 0 & \text{otherwise.} \end{cases}$$

Example 25. The adjacency matrix of the digraph shown in Example 23 is

$$A = \begin{pmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 & 0 \end{pmatrix}$$

In this case the vertices are ordered alphabetically so that the first row of the matrix corresponds to vertex a.

In the case of digraphs, the sum of the elements of a specific row of the matrix gives us the outdegree of the corresponding vertex, while the sum of all the elements of the matrix gives us the size of the digraph.

One of the advantages of the adjacency matrix is the simplicity of the data structure for storage, since it can be stored in a two-dimensional table (*array*).

For a graph of order n it would be $[1\ldots n, 1\ldots n]$ integer.

From the characteristics of this storage structure the following properties can be deduced:

(i) It is a very easy structure to manipulate and the time needed to access each position is constant.

The main disadvantage of the adjacency matrix is the unnecessarily occupied space. The space needed to store a graph of order n is proportional to n^2. As the number of edges is, at most, $\frac{1}{2}n(n-1)$, there will always be zeros in the matrix and storage spaces occupied unnecessarily.

(ii) If the graph is not directed, the matrix will be symmetric and still have more space occupied unnecessarily. In these cases a triangular matrix can be used to store the graph with a savings of 50% in the occupied space.

Definition 15 (Adjacency matrix of a hypergraph)**.** Let $H = (V,E)$ be a hypergraph and let $V = \{v_1, v_2, ..., v_n\}$. The *adjacency matrix* of H is defined as $A = (a_{ij})$ where the element a_{ij} is the number of hyperedges of H that contain v_i and v_j at the same time.

Example 26. Let $H = (V,E)$ be the hypergraph defined in the following way: $V = \{1,2,3,4,5,6\}$ and $E = \{A_1,A_2,A_3\}$, where $A_1 = \{1,2,3\}$, $A_2 = \{2,3,4,5\}$ and $A_3 = \{1,5,6\}$.

In this case the adjacency matrix is

$$A = \begin{pmatrix} 0 & 1 & 1 & 0 & 1 & 1 \\ 1 & 0 & 2 & 1 & 1 & 0 \\ 1 & 2 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 & 0 \end{pmatrix},$$

where the n-th row (or column) of the matrix corresponds to the n-th vertex of the hypergraph.

Incidence matrix and Laplacian matrix

Definition 16 (Incidence matrix of a digraph)**.** Let $A = (V,E)$ be a digraph where $V = \{v_1, v_2, ..., v_n\}$ and $E = \{e_1, e_2, ..., e_m\}$. The *incidence matrix* of G is defined as the matrix $I_G = (I_{ij})$ where

$$I_{ij} = \begin{cases} -1 & \text{if } v_i \text{ is the origin of } e_j; \\ 1 & \text{if } v_i \text{ is the endpoint of } e_j; \\ 0 & \text{otherwise.} \end{cases}$$

The rows of I_G correspond to the vertices of G and the columns correspond to the arcs.

Definition 17 (Laplacian matrix of a graph)**.** Let $A = (V,E)$ be a graph where $V = \{v_1, v_2, ..., v_n\}$. The *Laplacian matrix* of G is defined as the matrix $L = D - A$, where A is the adjacency matrix of G and $D = diag(\delta_1, \delta_2, ..., \delta_n)$ is the diagonal matrix whose elements of the diagonal are the degrees of the vertices.

Example 27. The Laplacian matrix of the graph shown in Example 24 is the following square matrix of order 5×5:

$$L = \begin{pmatrix} 2 & -1 & 0 & 0 & -1 \\ -1 & 4 & -1 & -1 & -1 \\ 0 & -1 & 2 & -1 & 0 \\ 0 & -1 & -1 & 3 & -1 \\ -1 & -1 & 0 & -1 & 3 \end{pmatrix}$$

Although the adjacency and Laplacian matrices of a graph seem to contain the same information on the graph, there are some properties of the graphs that can only be deduced from one of these matrices.

An orientation of the edges of a graph G is the allocation of a direction to each one of the edges of G. Therefore, if we give an orientation to a graph we obtain a digraph.

Theorem 6. *Given an orientation of the edges of a graph G, let I_G be the incidence matrix of G. Then the Laplacian matrix of G obeys*

$$L = I_G I_G^t.$$

Proof. The result is obtained directly from the product of the matrices I_G and I_G^t.

$$(I_G I_G^t)_{ij} = \sum_{l=1}^{m} I_{il} I_{jl} = \begin{cases} \delta_i & \text{if } i = j; \\ -1 & \text{if } v_i v_j \text{ is an edge of } G; \\ 0 & \text{otherwise.} \end{cases}$$

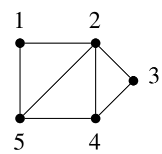
□

Adjacency list

To avoid the main problem of the matrix of adjacency (unnecessarily occupied space) one can opt for storing the graph as a list of adjacencies.

Given the graph $G = (V, E)$, the *adjacency list* of a simple graph is defined as a list of vertices adjacent to a given vertex.

Example 28. For the graph,

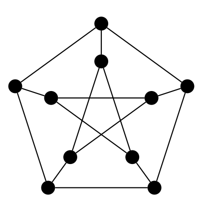


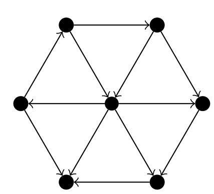
the adjacency list is

$$\begin{array}{lll} 1 & : & 2,5 \\ 2 & : & 1,3,4,5 \\ 3 & : & 2,4 \\ 4 & : & 2,3,5 \\ 5 & : & 1,2,4 \end{array}$$

As in the case of the adjacency matrix, the representation depends on the specific ordering of the vertices. In addition, this structure can also be generalised for multigraphs, pseudographs and oriented graphs.

Exercise 21. Consider the representation in matrix and adjacency list of the following graphs.





Determine the memory space occupied in each case.

Solution: The first graph is the Petersen graph that has order 10 and size 15. In the representation in adjacency matrix it would occupy $10^2 = 100$ memory units. The list of adjacencies would occupy $10 + 2 \cdot 15 = 40$ memory units.

The second graph is a directed graph of order 7 and size 12. In the representation in adjacency matrix it would occupy $7^2 = 49$ memory units. The list of adjacencies would occupy $7 + 12 = 19$ memory units. $\square$

Exercise 22. Depending on the size m of a graph of order n it can be more efficient to use the representation in adjacency matrix than the representation in list of adjacencies. Calculate, in terms of n and m, when it is more efficient to use one type of representation or another.

Solution: The adjacency matrix needs n^2 memory units, while the adjacency list needs $n + 2m$ (if the space necessary to store the links is not taken into account). Therefore, it will be better to use the adjacency list if $n + 2m < n^2$ or $m < \frac{n^2 - n}{2}$.

It can be seen that $\frac{n^2-n}{2}$ is the number of edges of a complete graph of n vertices. This shows that the adjacency list always occupies less memory that the adjacency matrix, except when the graph is complete, in which case they use the same memory. $\square$

Chapter 2

Graph operations

Graphs are basic combinatory structures and, as occurs in other disciplines of mathematics, the substructures and the basic operations with the elementary structures play a fundamental role both from a theoretical point of view, and in practical applications. In this chapter we will do a brief introduction to operations with graphs, which will allow us to obtain graphs whose structure can be quite complex compared to elementary graphs. To go into greater depth in the study of product graphs we recommend the following books: [6, 7].

2.1 Elementary operations

Given a graph $G = (V, E)$ diverse operations can be done.

• Remove a vertex $u \in V$. In this way we get the graph $G' = G - u$, which is the graph (V', E'), where $V' = V \setminus \{u\}$ and E' is the set of edges of G non-incident with u. This operation can be generalised to a set $W \subset V$. That is,

$$G' = G - W = (V \setminus W, \{\{a, b\} \in E : \ a, b \notin W\}).$$

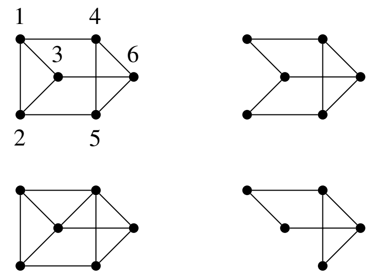
Obviously, this operation only has meaning if the graph is not trivial.

• Remove an edge $a \in E$. This is how to get a graph, with the same vertices, defined by $G' = G - a = (V, E \setminus \{a\})$; the operation can be trivially generalised to a subset of edges $B \subseteq E$, in which case $G - B = (V, E \setminus B)$.

• Add an edge $\{u, v\}$, with u and v being two non-adjacent vertices. In this way we get the graph $G' = (V, E \cup \{\{u, v\}\})$. This new graph can be represented by $G + uv$. The process can be generalised to sets of more than one edge.

The condition of non-adjacency of the vertices is fundamental, since the contrary would create a multiple edge and, therefore, would not be in the domain of simple graphs, as they have been defined.

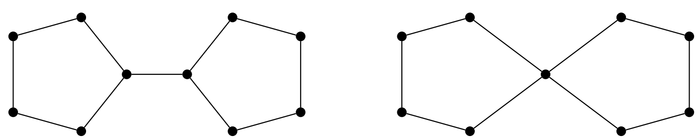
Example 29. On the first graph, effect operations to remove vertices and edges, and to add edges. The fourth graph is the result of removing vertex 2 from the original; the second graph results from removing edge $\{1, 2\}$ and the third graph is generated by the addition of the new edge $\{3, 4\}$.



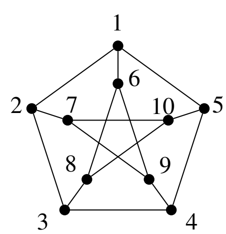
• Contract an edge $a = \{u, v\}$. In this case where edge a is removed, identifying in an single new vertex w the two endpoint vertices u, v, that disappear and, finally, the vertex w inherits exclusively the adjacencies of vertices u, v.

The operation of contraction can be applied to an entire set of edges.

Example 30. The following figure on the right shows the results of contracting an edge of the graph on the left.

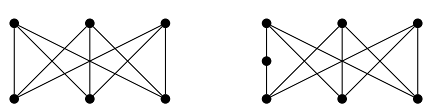


Example 31. In the following figure the *Petersen graph* is presented. If the edges $\{1,6\},\{2,7\},\{3,8\},\{4,9\}$ and $\{5,10\}$ are contracted, you get the complete graph K_5.



• Elementary subdivision of an edge $a = \{u, v\}$. In this case a vertex of degree 2 is inserted in edge a. That is, given $w \notin V$, the following operations are done: elimination of edge a, addition of the new vertex w, and addition of the new edges $\{u, w\}$ and $\{w, v\}$.

Example 32. The figure shows the result of doing an elementary subdivision of an edge of the graph $K_{3,3}$.



2.2 Subgraphs

Definition 18. A graph $H = (V', E')$ is a *subgraph* of the graph $G = (V, E)$ if $V' \subseteq V$ and $E' \subseteq E$.

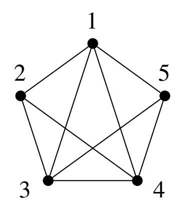
With the previous notation, if two vertices of H are not adjacent in G, then they they are not adjacent in H, but it is possible that two vertices of H are adjacent in G, and are not in H.

- Given a graph $G = (V, E)$ and a considered a subset $S \subseteq V$; the *subgraph generated* or *induced* by S is defined as the graph $\langle S \rangle = (S, E')$, in such a way that $\{u, v\} \in E' \Leftrightarrow \{u, v\} \in E$ and $u, v \in S$. Thus, the set of edges of $\langle S \rangle$ are those that, being in G, connect vertices belonging to S.

- Let $G = (V, E)$ and $H = (V', E')$ be two graphs; it is said that H is a *spanning subgraph* of G if $V' = V$ and $E' \subseteq E$.

The degree of a vertex u relative to a subgraph H is denoted by $\delta_H(u)$.

Example 33. Consider the graph represented in the figure:



An example of a subgraph can be the following: $H = (V', E')$, where $V' = \{1, 2, 4, 5\}$ and $E' = \{\{2, 4\}, \{4, 5\}\}$. Note that $\delta_G(4) = 4$ and, on the other hand, $\delta_H(4) = 2$.

A spanning subgraph would be, for example, $H = (V', E')$, where $V' = \{1, 2, 3, 4, 5\}$ and $E' = \{\{1, 2\}, \{1, 5\}, \{1, 4\}, \{3, 4\}\}$.

The subgraph induced by the set $\{3, 4, 5\}$ is an isomorph to C_3, the subgraph induced by $\{1, 2, 5\}$ is an isomorph to P_3, and the subgraph induced by $\{2, 5\}$ is an isomorph to N_2.

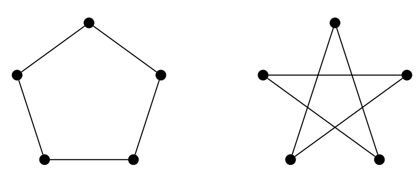
2.3 Complement

Definition 19. The *complement* of a graph $G = (V, E)$ is defined as the graph G^c that is constructed on the same set of vertices, so that two vertices are adjacent in G^c if and only if they are not adjacent in G.

Of course, it follows that the complement of the complement is the original:

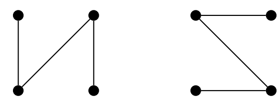
$$(G^c)^c = G.$$

Example 34. The following figure shows the graph C_5 and its complement, which goes back to being C_5.

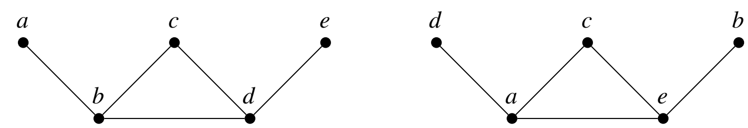


Example 35. $(K_n)^c = N_n$ and $(N_n)^c = K_n$.

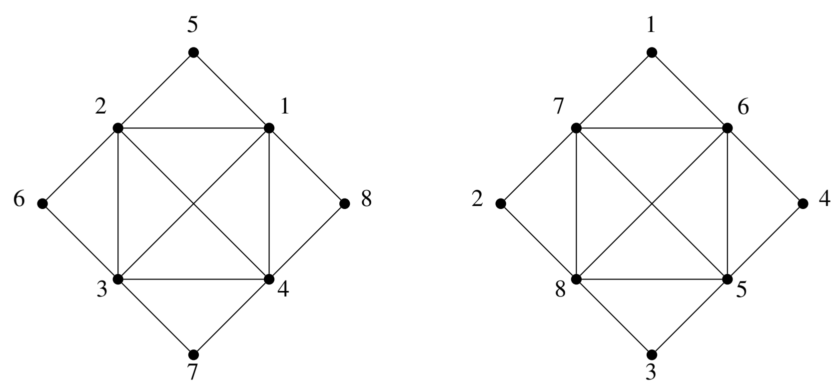
Example 36. The following figure shows that the path of order 4 is also self-complementary: $(P_4)^c \cong P_4$



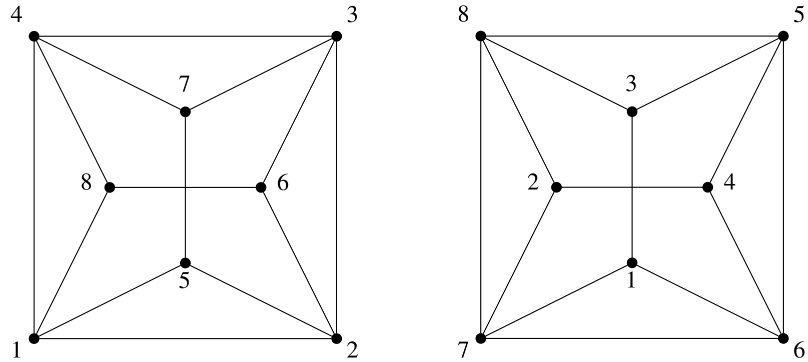
Example 37. A graph G that is isomorph to its complement: $G \cong G^c$.



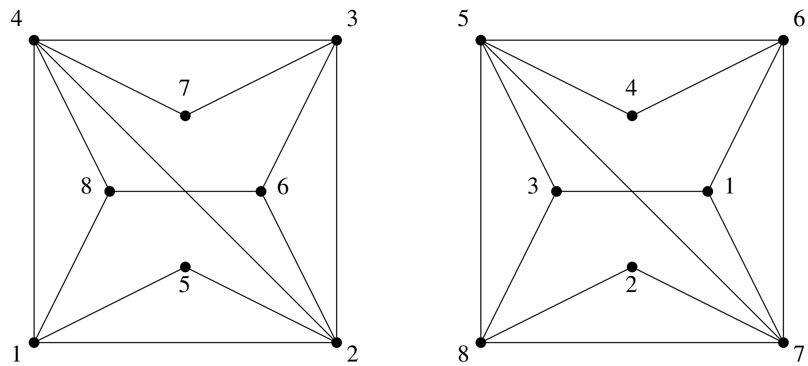
Example 38. A graph G (on the left) that is isomorphic to its complement (on the right): $G \cong G^c$.



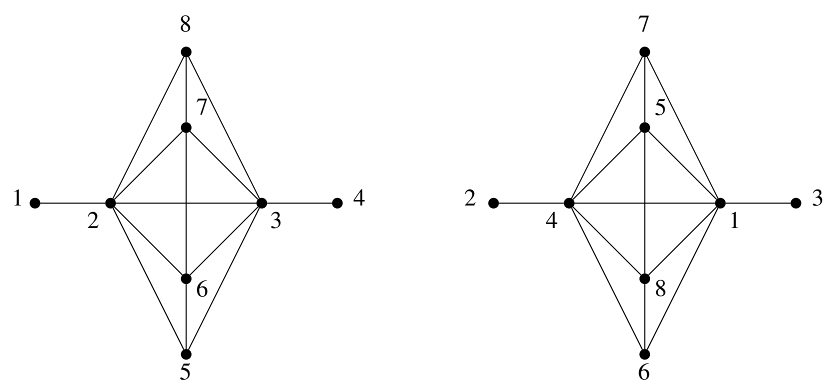
Example 39. A graph G (on the left) that is isomorphic to its complement (on the right): $G \cong G^c$.



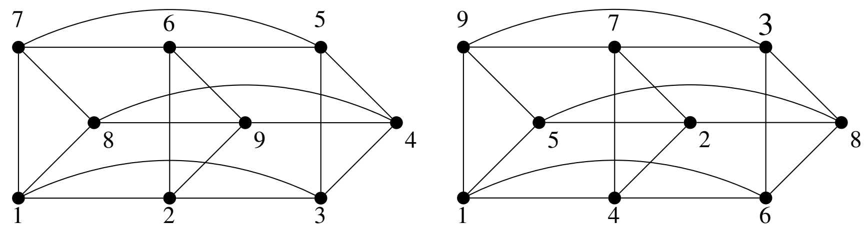
Example 40. A graph G (on the left) that is isomorphic to its complement (on the right): $G \cong G^c$.



Example 41. A graph G (on the left) that is isomorphic to its complement (on the right): $G \cong G^c$.



Example 42. In graph theory, an $n \times n$ *rook's graph* is a graph representing the valid moves of a chess rook on a board of $n \times n$ squares. Each vertex represents a square of the board, and each edge represents a valid move from one square to another.



The graph on the left represents the 3×3 rook's graph associated with the following board. On the right is its complement.

7	6	5
8	9	4
1	2	3

Observation 7. *Let $G = (V, E)$ be a graph of order n. For every vertex $v \in V$,*

$$\delta_{G^c}(v) = n - 1 - \delta_G(v).$$

Observation 8. *Note that two graphs are isomorphic if and only if their respective complements are isomorphic. That is,*

$$G \cong H \longleftrightarrow G^c \cong H^c.$$

Exercise 23. Determine a formula to calculate the size of G^c in terms of the order and the size of G.

Solution: The order of G^c coincides with the order of G and the size is

$$m(G^c) = \binom{n(G)}{2} - m(G).$$

□

Exercise 24. Determine the size of the complement of P_n and of the complement of C_n.

Solution: The size of the complement of P_n is $\binom{n}{2} - (n-1) = \frac{1}{2}(n-1)(n-2)$ and the size of the complement of C_n is $\binom{n}{2} - n = \frac{1}{2}n(n-3)$. □

Exercise 25. If $G = (V, E)$ is a graph of order $n = 6$, show that G or its complement G^c contains some triangle.

Solution: Let α be one of the six vertices of G. We distribute the five remaining vertices in two "boxes" in this way: in box 1 put those adjacent to α, and in box 2 put those that are not adjacent to α. Since $5 > 2 \cdot 2$, we can affirm that in one of the boxes there are at least three vertices. We assume that in box 1 we will find vertices β, γ, δ. If two of them, say β and γ, are adjacent, then α, β, γ form a triangle. If no pair of vertices between β, γ, δ are adjacent, then β, γ, δ are mutually adjacent in the complement (they form a triangle in G^c). An analogous reasoning can be made if box 2 is the one that contains three (or more) vertices. □

2.4 Line graph

Definition 20. The *line graph* of a graph $G=(V,E)$ with size $m \geq 1$ is a graph $L(G)=(E,E')$ such that the vertices of $L(G)$ are the edges of G and two vertices are adjacent in $L(G)$ if and only if the corresponding edges have a vertex in common.

Example 43. $L(K_{1,r})=K_r$, $L(P_n)=P_{n-1}$ and $L(C_n)=C_n$. The line graph of K_4 is $L(K_4)=K_6-F$, that is, $L(K_4)$ is the graph that one obtains after eliminating an adjacency to each vertex of the complete graph K_6. In this case F denotes a "pairing" of vertices and when subtracting it indicates that we have erased the corresponding edges formed by the pairs of vertices.

If $a=\{u,v\}$ is an edge of G, then the degree of a in $L(G)$ is

$$\delta(a)=\delta(u)+\delta(v)-2.$$

Thus, if G is regular and non-empty, then $L(G)$ is regular. The converse is not true. For example, $L(K_{1,5})$ is 4-regular while $K_{1,5}$ is not regular.

Exercise 26. Let G be a graph of size $m \geq 1$. Find the maximum integer r such that K_r is a subgraph of $L(G)$.

Solution: Let $\Delta(G)=\max\{\delta(v):\ v \in V(G)\}$. We know that two vertices of $L(G)$ are adjacent if the corresponding edges in G have a common vertex. Furthermore, $L(K_3) \cong K_3$. Therefore, if G is free of triangles, then $r=\Delta(G)$ and, if G has triangles, then $r=\max\{\Delta(G),3\}$. □

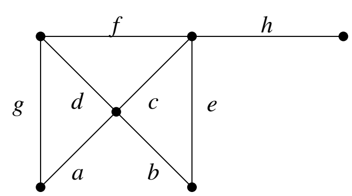
Exercise 27. Determine a formula for the size of $L(G)$ in terms of the degrees of G.

Solution: To calculate the size of $L(G)$ it is sufficient to observe that two edges of G are adjacent in $L(G)$ if and only if they are incident to a single vertex of G. Therefore the size of $L(G)$ can be calculated from the degree sequence of G:

$$m(L(G))=\sum_{v \in V(G):\, \delta(v)\geq 2}\binom{\delta(v)}{2}=\sum_{v \in V(G)}\frac{\delta(v)(\delta(v)-1)}{2}.$$

□

Exercise 28. Let G be the graph of the figure.



(a) Draw the line graph of G, denoted by $L(G)$.

(b) Calculate the number of edges of the complement graph of $L(G)$.

Solution:

(a) The line graph of G is shown below.

(b) $m((L(G))^c)=\binom{8}{2}-m(L(G))=28-17=11.$ □

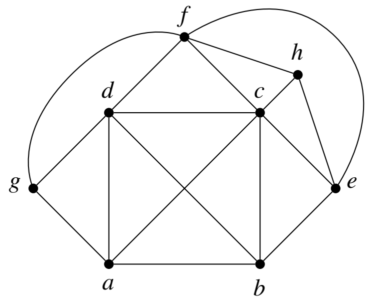


Figure 2.1: The line graph of G

2.5 Union of graphs

Definition 21. The *union* of two graphs $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$, where $V_1 \cap V_2 = \varnothing$, is the graph

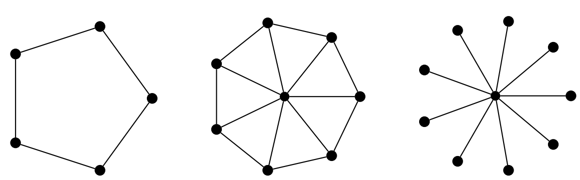
$$G_1 \cup G_2 = (V_1 \cup V_2, E_1 \cup E_2).$$

Example 44. $(K_r \cup K_s)^c = K_{r,s}$.

Notice that in the definition of union of two graphs G_1 and G_2, the interception of the vertex sets of G_1 and G_2 has to be empty. Now, in the example above, if $r = s$, we have $(K_r \cup K_r)^c = K_{r,r}$. In such a case, we are considering two different copies of K_r , i.e., two different graphs which are isomorphic to K_r.

Example 45. The graph of the following figure can be expressed as

$$G = C_5 \cup W_8 \cup K_{1,9}.$$



2.6 Join of graphs

Definition 22. Let $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ be two graphs such that $V_1 \cap V_2 = \varnothing$. The *join* $G_1 + G_2$ is the graph that has the vertices and the edges of the original graphs, in addition to the edges that connect all the vertices of G with all the vertices of H:

$$G_1 + G_2 = (V_1 \cup V_2, (E_1 \cup E_2 \cup \{\{u, v\} \mid u \in V_1,\ v \in V_2\})).$$

Example 46. The following graphs can be obtained as the join of known graphs.

- $K_{r+s} = K_r + K_s$.
- $K_{r,s} = N_r + N_s$

Notice that in the definition of join of two graphs G_1 and G_2, the interception of the vertex sets of G_1 and G_2 has to be empty. Now, in the example above, if $r = s$, we have $N_r + N_r = K_{r,r}$. In such a case, we are considering two different copies of N_r , i.e., two different graphs which are isomorphic to N_r.

Exercise 29. Determine a formula for the size of $G+H$.

Solution: The graph $G+H$ is obtained by taking a copy of graph G and one of graph H and later joining each vertex of G with all of those in H. Therefore, the size of $G+H$ is

$$m(G+H) = m(G) + m(H) + n(G) \cdot n(H).$$

□

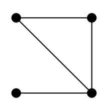
Exercise 30. Determine a formula for the size of $(G+H)^c$.

Solution: The size of the complement of $G+H$ is

$$\begin{aligned} m((G+H)^c) &= \binom{n(G+H)}{2} - m(G+H) \\ &= \binom{n(G)+n(H)}{2} - m(G) - m(H) - n(G) \cdot n(H). \end{aligned}$$

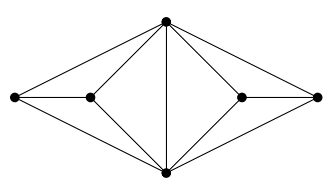
□

Exercise 31. Express the graph of the figure by means of graph operations.



Solution: $G = K_1 + (K_2 \cup K_1) = (P_3 \cup K_1)^c$. □

Exercise 32. Express the graph of the figure by means of graph operations.



Solution: $G = K_1 + (K_1 + (K_2 \cup K_2)) = K_2 + (K_2 \cup K_2) = (N_2 \cup C_4)^c$. □

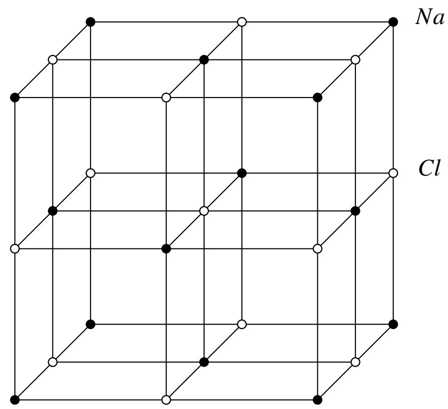
2.7 Cartesian product

Definition 23. Given two graphs $G = (V_1, E_1)$, $H = (V_2, E_2)$, the *Cartesian product* $G \square H = (V_1 \times V_2, E)$ is defined so that two vertices (g,h) and (g',h') are adjacent if and only if they satisfy *any* of the following conditions:

(1) $g = g'$ and $h \sim h'$, or

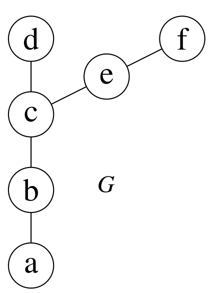
(2) $g \sim g'$ and $h = h'$

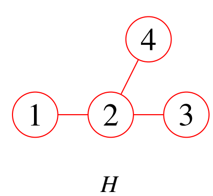
Example 47. The ionic compounds in solid state form reticular crystalline structures. The two main factors that determine the form of the crystalline network are the relative charges of the ions and their relative sizes. The following figure shows the crystalline network of NaCl. This representation corresponds to the graph $P_3 \square P_3 \square P_3$.

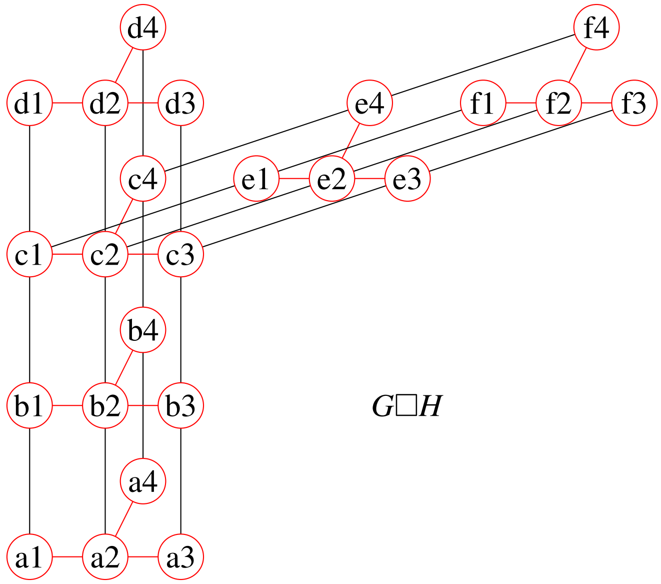


To imagine or graphically represent the cartesian product $G \square H$ we can think that by each vertex of H we put a copy of graph G and each vertex of the i-th copy of G will be adjacent to its twin in j-th copy of G if and only if i and j are adjacent in H.

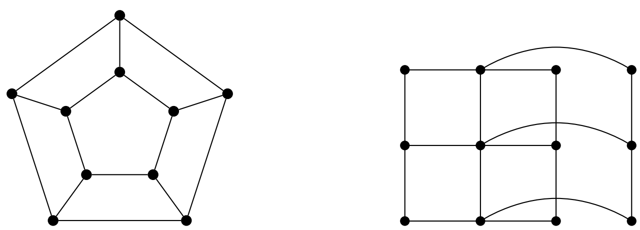
Example 48. In this figure an example of a cartesian product is shown where the vertices of $G \square H$ have been denoted by xy instead of (x,y).







Example 49. The following figure shows the graphs $C_5 \square K_2$ and $K_{1,3} \square P_3$.



The cartesian product is “commutative”, in the following sense.

Proposition 9. *For all pairs of graphs G and H,*

$$G \square H \cong H \square G.$$

Proof. The bijective application $\psi : V(G \square H) \longrightarrow V(H \square G)$ defined by $\psi(x,y) = (y,x)$ is an isomorphism of graphs. To check it take two vertices (a,b) and (u,v) adjacent in $G \square H$. There are two possibilities:

(1) $a = u$ and b adjacent to v in H.

(2) a adjacent to u in G and $b = v$.

In both cases the vertices $\psi(a,b) = (b,a)$ and $\psi(u,v) = (v,u)$ are adjacent in $H \square G$. □

Proposition 10. *The Cartesian product is associative.*

Proof. To show that the graphs $(G_1 \square G_2) \square G_3$ and $G_1 \square (G_2 \square G_3)$ are isomorphic, we will show that the bijective application

$$\psi : V((G_1 \square G_2) \square G_3) \longrightarrow V(G_1 \square (G_2 \square G_3))$$

defined by

$$\psi((u_1,u_2),u_3) = (u_1,(u_2,u_3))$$

is an isomorphism of graphs. We only have to prove that if $x, y \in V((G_1 \Box G_2) \Box G_3)$ are adjacent, then $\psi(x)$ and $\psi(y)$ are adjacent in $G_1 \Box (G_2 \Box G_3)$.

Let $x = ((u_1,u_2),u_3)$ and let $y = (v_1,v_2),v_3)$. There are two possibilities,

(1) $(u_1,u_2) = (v_1,v_2)$ and u_3 is adjacent to v_3 in G_3.

(2) (u_1,u_2) and (v_1,v_2) are adjacent in $G_1 \Box G_2$ and $u_3 = v_3$.

In case (1) the result is that $u_1 = v_1$ and (u_2,u_3) is adjacent to (v_2,v_3) in $G_2 \Box G_3$. Therefore, $\psi(x) = (u_1,(u_2,u_3))$ and $\psi(y) = (v_1,(v_2,v_3))$ are adjacent in $G_1 \Box (G_2 \Box G_3)$.

In case (2) there are two possibilities.

(2.1) $u_1 = v_1$, u_2 is adjacent to v_2 in G_2 and $u_3 = v_3$.

(2.2) u_1 and v_1 are adjacent in G_1, $u_2 = v_2$ and $u_3 = v_3$.

The subcase (2.1) leads us to $u_1 = v_1$ and the vertices (u_2,u_3) and (v_2,v_3) are adjacent in $G_2 \Box G_3$, while subcase (2.2) shows us that $(u_2,u_3) = (v_2,v_3)$ and the vertices u_1 and v_1 are adjacent in G_1. Therefore, case (2) also leads us to the fact that $\psi(x) = (u_1,(u_2,u_3))$ and $\psi(y) = (v_1,(v_2,v_3))$ are adjacent in $G_1 \Box (G_2 \Box G_3)$. □

Observation 11. *Given a vertex v of a graph G, we will denote the degree of v in G by $\delta_G(v)$. It follows that the degree of a vertex (a,b) in $G_1 \Box G_2$ is*

$$\delta_{G_1 \Box G_2}(a,b) = \delta_{G_1}(a) + \delta_{G_2}(b).$$

In consequence, a Cartesian product graph is regular if and only if the graph factors are regular.

Exercise 33. Determine a formula for the order and the size of $G \Box H$.

Solution: Let $G = (V_1,E_1)$ and $H = (V_2,E_2)$. The order of $G \Box H$ is given by

$$n(G \Box H) = |V_1 \times V_2| = n(G) \cdot n(H).$$

The size of $G \Box H$ is calculated by the degree sum formula:

$$\begin{aligned}
2m(G \Box H) = &\sum_{(a,b) \in V_1 \times V_2} (\delta_G(a) + \delta_H(b)) \\
= &\sum_{a \in V_1} \left(\sum_{b \in V_2} (\delta_G(a) + \delta_H(b)) \right) \\
= &\sum_{a \in V_1} (n(H) \cdot \delta_G(a) + 2m(H)) \\
= &2n(H) \cdot m(G) + 2m(H) \cdot n(G).
\end{aligned}$$

Therefore, $m(G \Box H) = n(G) \cdot m(H) + m(G) \cdot n(H)$. □

Definition 24. The family of hypercubes Q_k is defined by recurrence:

- $Q_k = Q_{k-1} \Box K_2$, $k \geq 2$.

- $Q_1 = K_2$.

The hypercube Q_k is also called a k-cube or hypercube of dimension k. An alternative definition of Q_k is the following: the set of vertices of Q_k is the set of binary words of length k and two vertices are adjacent if and only if the corresponding binary words only differ in one letter.

Exercise 34. Find the order and the size of Q_k.

Solution: The order of the hypercubes is $n = 2^k$ and the degree of each vertex is $\delta = k$. Thus, the size of Q_k is $m = \frac{n\delta}{2} = k2^{k-1}$. $\square$

Exercise 35. Find the size of the complement of $Q_3 \Box P_4$.

Solution: The size of $(Q_3 \Box P_4)^c$ is

$$m\left((Q_3 \Box P_4)^c\right) = \binom{32}{2} - m(Q_3 \Box P_4) = 496 - 72 = 424.$$

$\square$

Exercise 36. Let G be a δ-regular graph and let $H = (G \Box K_2) \Box C_4$.

(1) Find the degree of the vertices of H.

(2) Knowing that $\delta = 3$ and that the size of H is 192, find the order of G.

(3) With the data from the previous section, find the size of the complement of H.

Solution:

(1) The graph H is regular of degree $\delta + 3$.

(2) Let n be the order of G. As G is 3-regular, its size is $m = \frac{3n}{2}$. Thus, the size of H is $192 = 4(3n + n) + 8n = 24n$. Therefore, the order of G is $n = 8$.

(3) The size of the complement of H is $\binom{8n}{2} - 192 = \binom{64}{2} - 192 = 1824$. $\square$

Proposition 12. *If G and H are bipartite graphs, then $G \Box H$ is bipartite.*

Proof. Let $G = (V_1 \cup V_2, E)$ and $H = (V_1' \cup V_2', E')$. It is sufficient to observe that in $G \Box H$ the subgraphs $\langle V_1 \times V_1' \cup V_2 \times V_2' \rangle$ and $\langle V_1 \times V_2' \cup V_2 \times V_1' \rangle$ are empty and that the edges of $G \Box H$ go from the set $V_1 \times V_1' \cup V_2 \times V_2'$ to the set $V_1 \times V_2' \cup V_2 \times V_1'$. $\square$

In Chapter 4 we will see that the converse of the previous proposition is also true.

Corollary 13. *All hypercubes are bipartite.*

Exercise 37. Express the graph C_4 by means of graph operations.

Solution: $C_4 = N_2 + N_2 = (K_2 \cup K_2)^c = (N_2)^c \Box (N_2)^c = K_2 \Box K_2$. $\square$

Exercise 38. Let G be a non-trivial graph of order $n(G)$, size $m(G)$ and maximum degree $\Delta(G) \geq 2$. Let $G_1 = G$ and $G_k = G_{k-1} \square K_2$, $k \geq 2$.

(a) Find the order of the line graph of G_k.

(b) A clique of a graph H is a subgraph of H that is an isomorphic to a complete graph. The "clique number" of H, denoted by $\omega(H)$, is the maximum number of vertices of a clique of H. Find $w(\mathcal{L}(G_k))$.

(c) The Randić Index $R(H)$ of a graph $H = (V,E)$ was defined in 1975 as

$$R(H) = \sum_{v_i, v_j \in E} \frac{1}{\sqrt{\delta(v_i)\delta(v_j)}} .$$

This invariant, sometimes called the connectivity index, has been successfully related with several physical, chemical, and pharmacological properties of organic molecules. In addition, this topological index has become one of the most popular molecular descriptors.

Assuming that $G = K_{r,s}$, determine the formula for $R(G_k)$ in terms of r and s.

Solution:

(a) The order of the line graph of G_k coincides with the number of edges of G_k. That is,

$$n(\mathcal{L}(G_k)) = m(G_k) = 2m(G_{k-1}) + n(G_{k-1}) \tag{2.1}$$

We have,

$$\begin{aligned}
m(G_2) &= 2m(G) + n(G) \\
m(G_3) &= 2m(G_2) + n(G_2) \\
&= 2(2m(G) + n(G)) + 2n(G) \\
&= 2^2 m(G) + 2 \cdot 2n(G) \\
m(G_4) &= 2m(G_3) + n(G_3) \\
&= 2(2^2 m(G) + 2 \cdot 2n(G)) + 2^2 n(G) \\
&= 2^3 m(G) + 3 \cdot 2^2 n(G) \\
&\vdots \\
m(G_k) &= 2^{k-1} m(G) + (k-1)2^{k-2} n(G).
\end{aligned}$$

It is simple to prove by induction that this formula is correct. As we already have tested that the base case is true, we assume that the formula is true for an arbitrary $k \geq 3$. Using (2.1) and the hypothesis we have

$$m(G_{k+1}) = 2(2^{k-1} m(G) + (k-1)2^{k-2} n(G)) + 2^{k-2} n(G) = 2^k m(G) + k2^{k-1} n(G)).$$

Therefore, the formula is correct.

(b) First, $\mathcal{L}(K_3) = K_3$. Thus, if $G = K_3$, then $w(\mathcal{L}(G)) = 3$. On the other hand, if $G \neq K_3$, then all cliques of the graph $\mathcal{L}(G)$ can be obtained from a set of edges incident to the same vertex of G, except the triangles of G that transform into triangles in $\mathcal{L}(G)$. Thus, if G is free of triangles, then $w(\mathcal{L}(G)) = \Delta(G)$ and, if G has triangles, then $w(\mathcal{L}(G)) = \max\{\Delta(G), 3\}$.

Let $k \geq 2$. As the product by K_2 does not create new triangles, the value of $w(\mathcal{L}(G_k))$ coincides with the maximum degree of G_k. Therefore, $w(\mathcal{L}(G_k)) = \Delta(G) + k - 1$.

(c) First, in $K_{r,s}$ there are rs edges and the endpoints of each one of them have degree r and s.

In $G_2 = K_{r,s} \square K_2$ there are $2rs + r + s$ edges. Notice that there are $2rs$ edges whose vertices have degrees $r+1$ and $s+1$. In addition, there are r edges whose vertices both have degrees $s+1$ and there are s edges whose vertices both have degrees $r+1$

In $G_3 = G_2 \square K_2$ there are $2^2rs + 2\cdot 2r + 2\cdot 2s$ edges. Notice that there are 2^2rs edges whose vertices have degrees $r+2$ and $s+2$. In addition, there are $2\cdot 2r$ edges whose vertices both have degrees of $s+2$ and there are $2\cdot 2s$ edges whose vertices both have degrees $r+2$.

In $G_4 = G_3 \square K_2$ there are $2^3rs + 3\cdot 2^2r + 3\cdot 2^2s$ edges. Notice that there are 2^3rs edges whose vertices have degrees $r+3$ and $s+3$. In addition, there are $3\cdot 2^2r$ edges whose vertices both have degrees $s+3$ and there are $3\cdot 2^2s$ edges whose vertices both have degrees $r+3$.

$\vdots$

In general, in $G_k = G_{k-1} \square K_2$ there are $2^{k-1}rs + (k-1)\cdot 2^{k-2}r + (k-1)\cdot 2^{k-2}s$ edges. There are $2^{k-1}rs$ of these edges whose vertices have degrees $r+k-1$ and $s+k-1$. In addition, there are $(k-1)\cdot 2^{k-2}r$ edges whose vertices both have degrees $s+k-1$ and there are $(k-1)\cdot 2^{k-2}s$ edges whose vertices both have degrees $r+k-1$.

Thus,

$$R(G_k) = \frac{2^{k-1}rs}{\sqrt{(r+k-1)(s+k-1)}} + \frac{(k-1)\cdot 2^{k-2}r}{s+k-1} + \frac{(k-1)\cdot 2^{k-2}s}{r+k-1}.$$

□

2.8 Strong product

Definition 25. Given two graphs $G = (V_1, E_1)$ and $H = (V_2, E_2)$, the *strong product* $G \boxtimes H = (V_1 \times V_2, E)$ is defined so that the vertices (g,h), (g',h') are adjacent if and only if *any* of the following conditions are fulfilled:

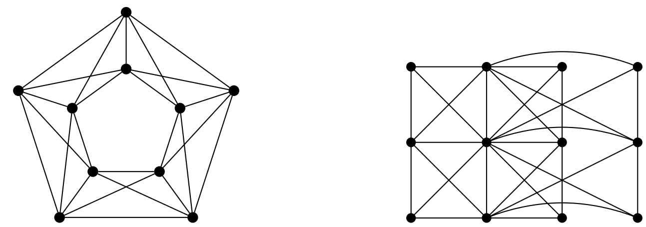
(1) $g = g'$ and $h \sim h'$,

(2) $g \sim g'$ and $h = h'$,

(3) $g \sim g'$ and $h \sim h'$.

Example 50. The following figure shows the graphs $C_5 \boxtimes K_2$ and $K_{1,3} \boxtimes P_3$.

Example 51. In general, for complete graphs we have $K_r \boxtimes K_s = K_{rs}$.



From the definition of the strong product of two graphs it can be deduced that $G\Box H$ is always a subgraph of $G\boxtimes H$, the order of $G\boxtimes H$ is $n(G\boxtimes H) = n(G)\cdot n(H)$, and the degree of a vertex (a,b) in $G\boxtimes H$ is

$$\delta_{G\boxtimes H}(a,b) = \delta_G(a)+\delta_H(b)+\delta_G(a)\cdot\delta_H(b).$$

Therefore, the strong product of regular graphs is a regular graph.

Exercise 39. Determine a formula for the size of $G\boxtimes H$.

Solution: Let $G=(V_1,E_1)$ and $H=(V_2,E_2)$. Applying the degree sum formula we obtain the size of $G\boxtimes H$:

$$\begin{aligned}
2m(G\boxtimes H) &= \sum_{(a,b)\in V_1\times V_2} \delta_{G\boxtimes H}(a,b)\\
&= \sum_{a\in V_1}\sum_{b\in V_2}(\delta_G(a)+\delta_H(b)+\delta_G(a)\cdot\delta_H(b))\\
&= \sum_{a\in V_1}(n(H)\delta_G(a)+2m(H)+2\delta_G(a)m(H))\\
&= 2n(H)\cdot m(G)+2m(H)\cdot n(G)+4m(G)\cdot m(H).
\end{aligned}$$

Therefore, the size of $G\boxtimes H$ is

$$m(G\boxtimes H) = n(H)\cdot m(G)+m(H)\cdot n(G)+2m(G)\cdot m(H).$$

$\square$

Observation 14. *Note that the strong product is "commutative", in the following sense:*

$$G\boxtimes H\cong H\boxtimes G.$$

In addition, the strong product is associative.

We will leave the proof of this observation as an exercise.

Exercise 40. Find the order and size of G:

(1) $G=[(K_{2,3}\cup K_{3,3})^c\Box(P_3\boxtimes K_6)]+K_{2,5}$

(2) $G=[(K_3+Q_3)\boxtimes(Q_3\Box K_{3,3})]^c$

Solution: From the definition of G the following can be deduced:

(1) The order is

$$\begin{aligned} n(G) &= n((K_{2,3} \cup K_{3,3})^c \square (T_3 \boxtimes K_6)) + 7 \\ &= (11 \cdot 18) + 7 = 198 + 7 = 205 \end{aligned}$$

and the size is

$$\begin{aligned} m(G) =& m\left((K_{2,3} \cup K_{3,3})^c\right) n(P_3 \boxtimes K_6) + n\left((K_{2,3} \cup K_{3,3})^c\right) m(P_3 \boxtimes K_6) + 10 + 198 \cdot 7 \\ =& (40 \cdot 18 + 11 \cdot 117) + 10 + 198 \cdot 7 \\ =& 3403. \end{aligned}$$

(2) The order is $n(G) = n(K_3 + Q_3) n(Q_3 \square K_{3,3}) = 11 \cdot 48 = 528$ and the size is

$$\begin{aligned} m(G) &= \binom{528}{2} - m((K_3 + Q_3) \boxtimes (Q_3 \square K_{3,3})) \\ &= \binom{528}{2} - (m(K_3 + Q_3) n(Q_3 \square K_{3,3}) + n(K_3 + Q_3) m(Q_3 \square K_{3,3}) + 2m(K_3 + Q_3) m(Q_3 \square K_{3,3})) \\ &= \binom{528}{2} - (39 \cdot 48 + 11 \cdot 144 + 2 \cdot 39144) \\ &= 124440. \end{aligned}$$

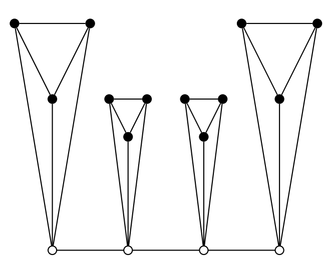
□

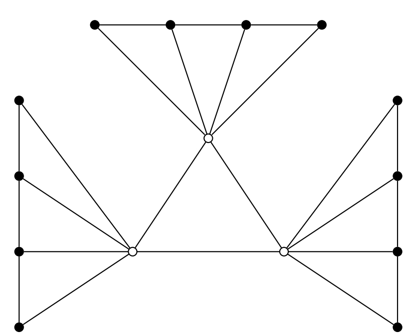
2.9 Corona product

Definition 26. Let G and H be two graphs. The *corona product* $G \odot H$ is defined from G and H by taking a copy of G and $n(G)$ copies of H, and joining (with an edge) each vertex of the i-th copy of H with the i-th vertex of G.

Notice that the corona product is not commutative and that $K_1 \odot H = K_1 + H$.

Example 52. The following figure shows the corona graphs $P_4 \odot C_3$ and $C_3 \odot P_4$.





Exercise 41. Determine a formula for the order and the size of $G \odot H$.

Solution: The order of $G \odot H$ is

$$n(G \odot H) = n(G) + n(G)n(H) = n(G)(1 + n(H))$$

and the size is

$$m(G \odot H) = m(G) + n(G)m(H) + n(G)n(H).$$

□

Exercise 42. Find the order and size of $G = (C_8 \odot N_6) \square (C_7)^c$.

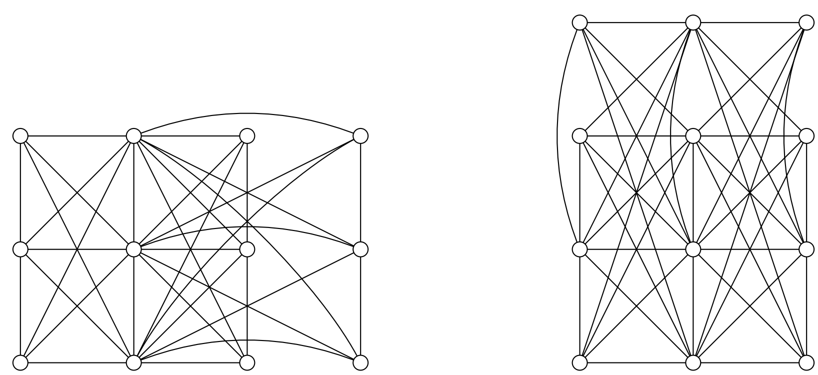
Solution: The order is $n(G) = n(C_8 \odot N_6) \cdot 7 = (8 + 8 \cdot 6) \cdot 7 = 392$ and the size is $m(G) = 56 \cdot 14 + 56 \cdot 7 = 1176$. □

2.10 Lexicographic product graphs

Definition 27. The *lexicographic product* of two graphs G and H is the graph $G \circ H$ whose vertex set is $V(G \circ H) = V(G) \times V(H)$ and $(g,h)(g',h') \in E(G \circ H)$ if and only if $gg' \in E(G)$ or $g = g'$ and $hh' \in E(H)$.

Notice that for any $g \in V(G)$ the subgraph of $G \circ H$ induced by $\{g\} \times V(H)$ is isomorphic to H. For simplicity, we will denote this subgraph by H_g. Moreover, the neighbourhood of $(g,h) \in V(G) \times V(H)$ will be denoted by $N_{G \circ H}(g,h)$ instead of $N_{G \circ H}((g,h))$.

Example 53. Two lexicographic product graphs: $K_{1,3} \circ P_3$ and $P_3 \circ K_{1,3}$.



Notice that for any $g \in V(G)$ the subgraph of $G \circ H$ induced by $\{g\} \times V(H)$ is isomorphic to H. For simplicity, we will denote this subgraph by H_g. Moreover, the neighbourhood of $(g,h) \in V(G) \times V(H)$ will be denoted by $N_{G \circ H}(g,h)$ instead of $N_{G \circ H}((g,h))$.

As a direct consequence of the definition of lexicographic product graph, the open neighbourhood of (g,h) in $G \circ H$ is given by

$$N_{G \circ H}(g,h) = N_G(g) \times V(H) \cup \{g\} \times N_H(h).$$

Thus, the degree of (g,h) in $G \circ H$ is given by

$$\delta_{G \circ H}(g,h) = \delta_G(g)n(H) + \delta_H(h).$$

Exercise 43. Determine a formula for the order and the size of $G \circ H$.

Solution: Let $G = (V_1, E_1)$ and $H = (V_2, E_2)$. The order of $G \circ H$ is

$$n(G \circ H) = |V_1||V_2| = n(G)n(H)$$

and the size is

$$\begin{aligned} 2m(G \circ H) &= \sum_{(g,h) \in V_1 \times V_2} \delta_{G \circ H}(g,h) \\ &= \sum_{g \in V_1} \sum_{h \in V_2} (\delta_G(g)n(H) + \delta_H(h)) \\ &= \sum_{a \in V_1} \big((n(H))^2 \cdot \delta_G(g) + 2m(H)\big) \\ &= 2(n(H))^2 \cdot m(G) + 2m(H) \cdot n(G). \end{aligned}$$

Therefore, the size of $G \circ H$ is

$$m(G \circ H) = m(G) \cdot (n(H))^2 + n(G) \cdot m(H).$$

$\square$

Observe that the lexicographic product is not commutative. In general,

$$G \circ H \ncong H \circ G.$$

Exercise 44. Prove that for any graph G and any integer $n \geq 1$,

$$G \circ K_n \cong K_n \boxtimes G.$$

Solution: Observe that for any graph G and any integer $n \geq 1$ we have

$$V(G \circ K_n) = V(G) \times V(K_n) = V(G \boxtimes K_n).$$

Furthermore,

$$E(G \circ K_n) = \{\{(a,b),(a',b')\} : \ a = a' \text{ or } a \sim a'\} = E(G \boxtimes K_n).$$

Therefore, $G \circ K_n \cong G \boxtimes K_n \cong K_n \boxtimes G$. $\square$

Chapter 3

Walks and connectivity

Various applications of graphs lead to problems where it is necessary to establish ways to traverse or explore the graphs through its nodes and edges. Simple examples are communication networks. In this chapter we introduce some basic concepts and present some network exploration algorithms that will allow us to establish the bases for later study of problems related to the concept of distance.

3.1 Concepts and basic results

Definition 28. A *walk* in a graph $G = (V, E)$ is a sequence of vertices $v_1, v_2, \ldots, v_k$ with the property that $\{v_i, v_{i+1}\} \in E$ for every $i \leq k-1$. A walk of endpoints v_1 and v_k is called a $v_1 - v_k$ *walk* or, also, a walk between v_1 and v_k.

On occasion, and to avoid any confusion, the walks $v_1, v_2, \ldots, v_k$ will be denoted by $v_1 v_2 \ldots v_k$, without the use of commas.

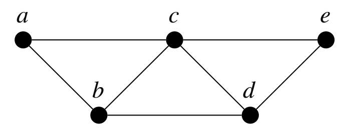
Definition 29. If $G = (V, E)$ is a digraph, an *oriented* or *directed* walk is a sequence of vertices $w_1, w_2, \ldots, w_k$ with the property that $(w_i, w_{i+1}) \in E$ for every $i \in \{1, \ldots, k-1\}$.

In a multigraph it is possible to go from one vertex to an adjacent one by more than one edge; thus, it is necessary to indicate a route as a sequence of edges, $a_1, a_2, \ldots, a_{k-1}, a_k$, in which two consecutive edges in the sequence have to share an endpoint.

Definition 30.

- A $u - v$ walk is *closed* if the endpoints coincide, that is, if $u = v$; in the opposite case we say that it is *open*.
- The *length of a walk R*, $\ell(R)$, is the number of edges that it is composed of. It also includes those that may be repeated.
- A trivial *walk* is that formed by a single vertex.

Example 54. Consider the graph of the figure.



A walk of length 4 is a,b,c,d,b. A closed walk of length 6 is a,b,c,d,e,c,a and another of length 3 is a,b,c,a.

Example 55. In the graph $G = C_3 \cup P_5$ there is no possible walk between the vertices C_3 and P_5; the vertices in question are not mutually accessible in the graph.

Definition 31. A walk is a *trail* if all the edges are different. The following types of trail can be highlighted:

- A *path*, if vertices are not repeated.
- A *circuit*, if it is closed.
- A *cycle* is a circuit (closed) that, by deleting the first vertex, is also a walk (does not repeat vertices). The graphs that do not contain cycles are called *acyclic*.

Example 56. Consider the graph on the right in the figure of Example 54.
Path: a,b,c,d.
Cycle: b,c,d,b.
Circuit: a,b,c,d,e,c,a.

Proposition 15. *Given two different vertices u, v of a graph $G = (V,E)$, all $u-v$ walks contain a path, $u-v$ that is, a walk between the vertices without repetition of any vertices.*

Proof. Let R be a walk $u-v$ of the graph G. Since u and v are different, we have R is open, and let for example $R:\ u = w_0w_1\dots w_k = v$, being able to contain repeated vertices. If none are repeated, we have reached the result, since R fulfils the condition required. We assume, now, that there is some vertex repeated, and let $i < j$ such that $w_i = w_j$; then we eliminate the vertices $w_iw_{i+1}\dots w_{j-1}$, and a new $u-v$ walk R_1 results, with a strictly shorter length. If R_1 is a path, we are finished ; in the opposite case, the process continues and it necessarily arrives at the indicated conclusion, since the initial length is finite. $\square$

Proposition 16. *If in a graph $G = (V,E)$ there are two different walks that connect a pair of vertices, then the graph contains at least one cycle.*

Proof. Consider two different $u-v$ walks: $R_1 : u = w_1w_2\dots w_p = v$ and $R_2 : u = t_1t_2\dots t_q = v$. Informally, the walks can share vertices until arriving at the first vertex where they diverge, that is, they begin to be different. Let i be the first index for which $w_i \neq t_i$. Sooner or later the routes have to converge, finally, on the common terminal vertex, which is v, and it will do this in the same vertex or before; but, in any case, from a specific place afterward, the two routes will go back to sharing vertices (and edges). Now, at some vertex following the previous bifurcation they will have to merge together for the first time. Let j, k be minimum such that $i < j \leq p$, $i < k \leq q$ and $w_j = t_k$. Then, in agreement with Proposition 15, the walk $w_{i-1}w_iw_{i+1}\dots w_j$ contains a path $w_{i-1}w'_1\dots w'_{p'-1}w_j$ and the walk $t_{k-1}t_{k-2}\dots t_iw_{i-1}$ contains a path $t_{k-1}t'_1\dots t'_{q'-1}w_{i-1}$. Therefore, the walk $w_{i-1}w'_1\dots w'_{p'-1}w_jt_{k-1}t'_1\dots t'_{q'-1}w_{i-1}$ is a cycle. $\square$

Theorem 17. *If all the vertices of a graph $G = (V,E)$ have degree greater than or equal to 2, then G contains a cycle.*

Proof. Let $u_1, u_2, \ldots, u_k$ be a path of maximum length in G. As u_k has a degree greater than or equal to 2, there exists at least one vertex $a \neq u_{k-1}$ that is adjacent to u_k. By the maximality of the length of the path selected, it follows that $a = u_i$ for some $i \in \{1, \ldots, k-2\}$. We have obtained the cycle, $u_i, u_{i+1}, \ldots, u_k, u_i$. □

Exercise 45. Determine if the following statements are true or false.

(1) All acyclic graphs without isolated vertices contain vertices of degree 1.

(2) There are no acyclic graphs whose complement is also acyclic.

(3) If in a graph of order n the maximum degree of the vertices is less than $n-2$, then the complement graph contains at least one cycle.

Solution:

(1) True. If it did not have any vertex of degree 1, then all the vertices would have a degree greater than or equal to 2 and according to Theorem 17 it would have to has a cycle.

(2) False. For example, P_3 and its complement $K_1 \cup K_2$ are acyclic.

(3) True. If G^c is the complement of G, then $\delta_{G^c}(v) = n - 1 - \delta_G(v) \geq 2$ for all $v \in V$ and, by Theorem 17, G contains a cycle.

□

Theorem 18. *Let v_i and v_j be two vertices of a graph G. Let A be the adjacency matrix of G. Then the number of walks of length k in G, from v_i to v_j, is the element of the position (i, j) of A^k.*

Proof. We proceed by induction on k. The result is true for $k = 1$, since $A^1 = A(G)$.

Hypothesis: Let us assume that the result is true for $k \geq 1$.

Now, let v_h and v_j be adjacent vertices. The number of walks of length $k+1$ from v_i to v_j and that contain v_h is $\left(A^k\right)_{ih} \cdot a_{hj}$. Therefore, the number of walks of length $k+1$ from v_i to v_j is

$$\sum_{v_h \sim v_j} \left(A^k\right)_{ih} \cdot a_{hj} = \sum_{h=1}^{n} \left(A^k\right)_{ih} \cdot a_{hj} = \left(A^{k+1}\right)_{ij}.$$

Therefore, the result follows. □

Exercise 46. Which information about the graph offers us the trace of A^2?

Solution: The degrees of the vertices of the graph are in the diagonal of A^2. Therefore, according to the degree sum formula, the size of the graph is

$$m = \frac{1}{2} Tr(A^2).$$

□

3.2 Connected graphs

Definition 32. A graph $G = (V, E)$ is *connected* if for each pair of vertices u and v of G there exists a $u - v$ path.

Example 57. The following graphs are connected: K_n, P_n, $K_{r,s}$, C_n, W_n. The following graphs are not connected: $K_n \cup K_{r,s}$, $(P_3 + C_4)^c$, N_n $(n \geq 2)$.

Proposition 19. *For all pairs of graphs G and H, the graph $G + H$ is connected.*

Proof. Let $u, v \in V(G+H)$. If $u \in V(G)$ and $v \in V(H)$, then u and v are adjacent in $G+H$. If $x, y \in V(G)$, then for any vertex $w \in V(H)$, we have that axwy as a path in $G+H$. Analogously, it can be observed that if $x, y \in V(H)$, then there is an $x - y$ path in $G+H$. Therefore, $G+H$ is connected. □

Proposition 20. *$G \odot H$ is connected if and only if G is connected.*

Proof. Let $V = \{v_1, \dots, v_n\}$ be the set of vertices of G and let $H_i = (V_i, E_i)$ be the i-th copy of H in $G \odot H$. Notice that for every $i \in \{1, \dots, n\}$, every vertex of H_i is adjacent to v_i.

(Necessity) Let $G \odot H$ be connected. In this case, for any $a \in V_i$ and $b \in V_j$, $i \neq j$, there exists an $a - b$ path $av_i \dots v_j b$ in $G \odot H$. Thus, for any pair of vertices $v_i, v_j \in V$ there exists a $v_i - v_j$ path in G. Therefore, G is connected.

(Sufficiency) Let G be connected and let $a \in V_i$ and $b \in V_j$, $i \neq j$. As there exists a $v_i - v_j$ path $v_i \dots v_j$ in G, an $a - b$ path $av_i \dots v_j b$ also exists in $G \odot H$. In addition, if $i = j$, then $av_i b$ is a path in $G \odot H$. Therefore, $G \odot H$ is connected. □

Proposition 21. *$G \Box H$ is connected if and only if G and H are connected.*

Proof. (Necessity) Let $G \Box H$ be connected and let $a, u \in V(G)$ and $b, v \in V(H)$. As $G \Box H$ is connected, there exists a path Q from (a, b) to (u, v). Therefore, the projection $P_G(Q)$ of the vertices belonging to Q on the set of vertices of G is a walk from a to u. Then, by Proposition 15, there exists an $a - u$ path in G. Analogously, it can be proved that there exists a $b - v$ path in H. Therefore, G and H are connected.

(Sufficiency) Let G and H be two connected graphs and let $(a, b), (u, v)$
$\in V(G \Box H)$. By the connectivity of G and H, there exists a path $aa_1a_2 \dots a_k u$ from a to u in G and a path $bb_1b_2 \dots b_l v$ from b to v in H. Then

$$(a,b)(a_1,b)\dots(a_k,b)(u,b)(u,b_1)\dots(u,b_l)(u,v)$$

is a path from (a, b) to (u, v) in $G \Box H$. Therefore, $G \Box H$ is connected. □

The following result is a direct consequence of the previous proposition. To show it, it is only necessary to observe that $G \Box H$ is connected if and only if $G \boxtimes H$ is.

Corollary 22. *$G \boxtimes H$ is connected if and only if G and H are connected.*

Proposition 23. *Let G be a graph of order at least two and let H be a graph. The lexicographic product graph $G \circ H$ is connected if and only if G is connected.*

Proof. (Necessity) Let $G \circ H$ be connected and let $g, g' \in V(G)$ and $h, h' \in V(H)$. As $G \circ H$ is connected, there exists a path Q from (g,h) to (g',h'). Therefore, the projection $P_G(Q)$ of the vertices belonging to Q on the set of vertices of G is a walk from g to g'. Then, by Proposition 15, there exists a $g - g'$ path in G. Therefore, G is connected.

(Sufficiency) Let G be a connected non-trivial graph and let $(g,h),(g',h') \in V(G \circ H)$ be two different vertices. If $g = g'$ then for any $g'' \in N_G(g)$ there exists the path $(g,h),(g'',h),(g,h')$ from (g,h) to (g',h'). From now on, assume $g \neq g'$. If $g' \in N_G(g)$, then (g,h) and (g',h') are adjacent. Finally, if $g' \notin N_G(g)$, then by the connectivity of G, there exists a path $ga_1a_2\ldots a_kg'$ from g to g' in G. Hence, $(g,h)(a_1,h)\ldots(a_k,h)(g',h')$ is a path from (g,h) to (g',h') in $G \circ H$. Therefore, $G \circ H$ is connected. □

Proposition 24. *If G is connected, then $L(G)$ is connected.*

Proof. Let $a = \{u_i, v_i\}$ and $b = \{u_j, v_j\}$ be two edges of G. As G is connected, there exists a path from u_i to u_j and a path from u_i to v_j in G. Hence, there exists a path from a to b in $L(G)$ and, as a result, $L(G)$ is connected. □

Note that the converse of the previous proposition in not true. For example, the graph $L(K_4 \cup N_5) = L(K_4)$ is connected, while $K_4 \cup N_5$ is not.

When a graph G is not connected, it will be necessary to determine those subgraphs of G that are connected and that are of a maximum nature with respect to this property.

Definition 33. In the set of vertices V of a graph $G = (V,E)$ the following relation is defined:

- $x\mathcal{R}_G v$ for every $x \in V$.
- Given two different vertices $x, y \in V$, $x\mathcal{R}_G y$ if and only if there exists an $x - y$ path in G.

Proposition 25. *For any non-trivial graph G, the relation $\mathcal{R}_G$ is an equivalence relation.*

In consequence, a partition of $V = V_1 \cup \ldots \cup V_k$ is established, where the vertices of a same class are mutually accessible by some path and vertices of different classes are inaccessible.

Definition 34. The *components* of a graph G are the subgraphs $G_i = \langle V_i \rangle$ generated for each one of the classes of equivalence of G with regards to the relation $\mathcal{R}_G$. Thus, all graphs can be expressed as a union of its components:

$$G = G_1 \cup \ldots \cup G_k$$

Notice that a graph is connected if and only if it has a single component.

Comparing the order and the size of a graph with the orders and sizes of its components provides the following relation:

Observation 26. *If a graph has of order n and size m, with k components, of respective orders n_i and sizes m_i $(i = 1, \ldots, k)$, then:*

$$n = \sum_{i=1}^{k} n_i, \quad m = \sum_{i=1}^{k} m_i .$$

Exercise 47. All graphs with degree sequence $4,4,3,3,3,3,3,3$ are connected.

Solution: Let us assume that some non-connected graph G exists with this sequence of degrees. Note that G has order $n = 8$. As there are vertices of degree 4, some component G_i will have order $n_i \geq 5$ and as the minimum degree is 3, any other component G_j will have order $n_j \geq 4$. Thus, $n \geq n_i + n_j \geq 9 > 8 = n$, which is a contradiction. □

The following result establishes the minimum size of a connected graph.

Proposition 27. *If G is a connected graph of order n and size m, then*

$$m \geq n-1.$$

Proof. We proceed by induction with regard to the order n of the graph. If $n = 1$, then the result is immediate.

Let us assume that the property is true for every connected graph of order $n-1 \geq 1$. Let $G = (V,E)$ be a connected graph of order n and size m.

If G has a vertex v of degree 1, then the graph $G' = G - v$ is a connected graph of order $n-1$ and size $m-1$. Applying the hypothesis of induction to the graph G' it must be true that $m-1 \geq n-2$, from where we determine that $m \geq n-1$, which is what we wanted to prove.

If every vertex of G has degree greater than or equal to 2, then by the degree sum formula we have,

$$2m = \sum_{v \in V} \delta(v) \geq 2n$$

and, therefore, $m \geq n > n-1$. □

As a direct consequence of this result we can obtain the following consequence:

Proposition 28. *If G is a graph of order n, size m and k components, then*

$$m \geq n-k.$$

Exercise 48. Show that no connected graph has the degree sequence $3,3,2,1,\ 1,1,1,1,1$.

Solution: First, it can be verified, by using the Havel-Hakimi algorithm, that the sequence $3,3,2,1,1,1,1,1,1$ is graphic. Let G be a graph with this degree sequence.

As G has order $n = 9$, if it were connected it would have size $m \geq 8$. But, by the degree sum formula, $m = \frac{1}{2}(3+3+2+1+1+1+1+1+1) = 7$. Therefore, G it is not connected. □

Exercise 49. Prove that a non-trivial graph G is connected if and only if 0, as eigenvalue of the Laplacian matrix of G, has multiplicity 1.

Solution: ($\Rightarrow$) Let G be a connected graph, $V(G) = \{1,\ldots,n\}$, and let $B = (\overrightarrow{u_1},\ldots,\overrightarrow{u_k})$ a basis of $\ker(L(G))$. We can assume that these vectors are normalized in such a way that its maximum component is 1. In particular, let $\overrightarrow{u_l} = (x_1,\ldots,x_n)$ such that $x_j = \max\{x_i\} = 1$. In this case, $\delta(j) = \sum_{i \sim j} x_i$, which implies that $x_i = 1$ for every $i \sim j$. By the same argument we can conclude that $x_r = 1$ for every $r \sim i \in N(j)$, and by the connectivity of G, we deduce that $\overrightarrow{u_l} = (1,1,\ldots,1)$ for every eigenvector in B. Therefore, $k = \dim(\ker(L(G))) = 1$.

($\Leftarrow$) If $G_1, \ldots, G_k$ are the components of G, then there exist k eigenvectors $\overrightarrow{u_1}, \ldots, \overrightarrow{u_k}$, associated to the eigenvalue 0, which are linearly independent, and are defined by $\overrightarrow{u_l} = (x_1^l, \ldots, x_n^l)$ where $x_i^l = 1$ if $i \in V(G_l)$, while $x_i^l = 0$ for $i \notin V(G_l)$. Therefore, if $k \neq 1$, then $\dim(\ker(L(G))) \neq 1$. □

The statement of the previous exercise can be generalized as follows.

Proposition 29. *A non-trivial graph G has k connected components if and only if* 0*, as eigenvalue of the Laplacian matrix of G, has multiplicity k.*

3.3 Graph exploration algorithms

To determine some specific properties of a graph, it is often necessary to explore each one of the vertices and edges of the graph in a given order. For instance, there are two basic procedures to determine if a graph is connected: the *Depth First Search algorithm* (*DFS*) and the *Breadth First Search algorithm* (*BFS*). These two algorithms are integral parts of numerous algorithms used in computer science, operations research, and other engineering disciplines.

DFS Algorithm

The DFS algorithm begins the exploration of a graph from a pre-determined vertex and then explores all others via the edges, without repetition, in a pre-determined way. Essentially, it aims to advance as deeply as possible, without exploring all the possibilities that come from the initial vertex. In this way, from the initial vertex it proceeds to an adjacent vertex not previously visited, from this to a third one not yet visited, and continuing like this successively. If it arrives at a vertex from which it cannot continue, it is necessary to go backwards and delete it. This process repeats until the initial vertex is deleted, at which point the exploration will have finished. As might be expected, the DFS algorithm can be completed with additional actions and formulate new variants.

Formulation of the DFS algorithm

Necessary structures for the formulation of the algorithm:

- A graph $G = (V, E)$ represented by an adjacency list.
- A set P of the vertices that have been visited, in the order done, that allows for going backwards. The correct data structure is the one of a stack with the usual operations: *stack*(P, v), *unstack*(P), *peak*(P).
- A table of vertices ($state$) that registers the vertices that are being visited.
- A list R that contains an up-to-date list of visited vertices (and which will, in the end, coincide with the set of all vertices).

Whenever it is possible to visit more than one vertex, the one chosen will be the one with the lower index of agreement to the ordering of the available vertices.

The stack P is begun with the starting vertex and the vertices visited are stacked on top. When it reaches a vertex from which it cannot continue, it will continue unstacking vertices until it finds one from which it can begin advancing again, thus re-initiating the process. The process ends when the stack is empty.

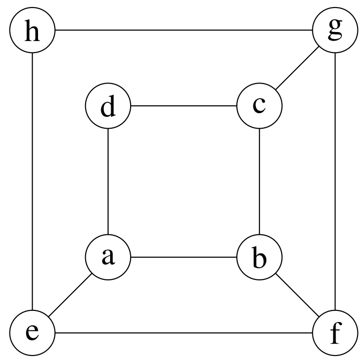
Algorithm DFS
Imput: $G(V,E)$, $v \in V$
Output: R, a list of visited vertices

```
DFS Algorithm (G, v)
  Start
    P ← ∅
    R ← [v]
    for w ∈ V
        State [w] ← 0
    endfor
    state [v] ← 1
    stack (P, v)
    while P ≠ ∅
            w ← peak (P)
            If w is adjacent to u with state [u] = 0
                then stack (P, u)
                        state [u] ← 1
                        add (R, u)
                  else unstack (P)
            endif
    endwhile
    Return (R)
  End
```

When applying the DFS algorithm each vertex is visited twice, once when it is added to the stack and again when it is deleted. In addition, when a vertex of the stack is reached, all the adjacent ones are analysed to look for the vertices that have not yet been visited. Therefore, if the graph G has order n and size m, the total number of operations will be proportional to $2n+2m$ and, in consequence, the algorithm will have a complexity of $O(n+m)$.

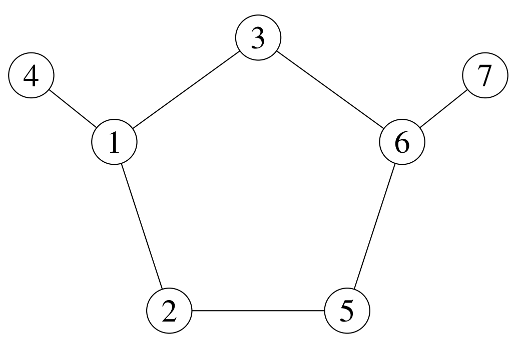
Example 58. Consider the graph represented in the following figure.

This table registers the operation of the Depth First Search algorithm (DFS) for this graph, with starting vertex $v = a$.



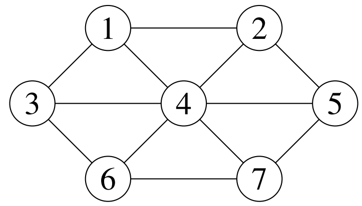
P	Added vertex	Deleted vertex	R
a	a	-	$[a]$
ab	b	-	$[a,b]$
abc	c	-	$[a,b,c]$
$abcd$	d	-	$[a,b,c,d]$
abc	-	d	$[a,b,c,d]$
$abcg$	g	-	$[a,b,c,d,g]$
$abcgf$	f	-	$[a,b,c,d,g,f]$
$abcgfe$	e	-	$[a,b,c,d,g,f,e]$
$abcgfeh$	h	-	$[a,b,c,d,g,f,e,h]$
$abcgfe$	-	h	$[a,b,c,d,g,f,e,h]$
$abcgf$	-	e	$[a,b,c,d,g,f,e,h]$
$abcg$	-	f	$[a,b,c,d,g,f,e,h]$
abc	-	g	$[a,b,c,d,g,f,e,h]$
ab	-	c	$[a,b,c,d,g,f,e,h]$
a	-	b	$[a,b,c,d,g,f,e,h]$
$\varnothing$	-	a	$[a,b,c,d,g,f,e,h]$

Exercise 50. Find the list of vertices which are visited by the DFS algorithms in the graph of the figure, where 6 is the starting vertex.



Solution: The list of visited vertices is $R = [6,3,1,2,5,4,7]$. □

Exercise 51. Find the list of vertices which are visited by the DFS algorithm in the following graph, where 1 is the starting vertex.



Solution: The list of visited vertices is $R = [1,2,4,3,6,7,5]$. □

Exercise 52. Use the DFS algorithm to obtain the set S of edges which are visited in the graph of Exercise 50, where 1 is the starting vertex. Build also the table of operation of the algorithm with the following header:

P	Added edge	S

Solution: The table of operation of the algorithm will be:

P	Added edge	S
1	-	$\varnothing$
12	$\{1,2\}$	$[\{1,2\}]$
125	$\{2,5\}$	$[\{1,2\},\{2,5\}]$
1256	$\{5,6\}$	$[\{1,2\},\{2,5\},\{5,6\}]$
12563	$\{6,3\}$	$[\{1,2\},\{2,5\},\{5,6\},\{6,3\}]$
1256	-	$[\{1,2\},\{2,5\},\{5,6\},\{6,3\}]$
12567	$\{6,7\}$	$[\{1,2\},\{2,5\},\{5,6\},\{6,3\},\{6,7\}]$
1256	-	$[\{1,2\},\{2,5\},\{5,6\},\{6,3\},\{6,7\}]$
125	-	$[\{1,2\},\{2,5\},\{5,6\},\{6,3\},\{6,7\}]$
12	-	$[\{1,2\},\{2,5\},\{5,6\},\{6,3\},\{6,7\}]$
1	-	$[\{1,2\},\{2,5\},\{5,6\},\{6,3\},\{6,7\}]$
14	$\{1,4\}$	$[\{1,2\},\{2,5\},\{5,6\},\{6,3\},\{6,7\},\{1,4\}]$
1	-	$[\{1,2\},\{2,5\},\{5,6\},\{6,3\},\{6,7\},\{1,4\}]$
$\varnothing$	-	$[\{1,2\},\{2,5\},\{5,6\},\{6,3\},\{6,7\},\{1,4\}]$

□

BFS Algorithm

The BFS algorithm is similar to the DFS, with the important difference that from the initial vertex, it first makes a systematic visit to all its adjacent vertices that have not yet been visited, chosen by the order of labeling of the vertices.

The formulation of the basic algorithm can be completed with additional actions to formulate new variants.

Formulation of the BFS algorithm

Necessary structures for the formulation of the algorithm:

- A graph $G = (V,E)$ represented by the adjacency list.

- A set Q of the vertices that have been visited, in the order done. The proper data structure is the that of a queue with the usual operations: *add*(Q,v), *eliminate*(Q), *first*(Q).
- A table of vertices (*state*) that registers the vertices that have been visited.
- A list R that contains the visited vertices up to the moment (and that, in the end, will coincide with the set of all the vertices).

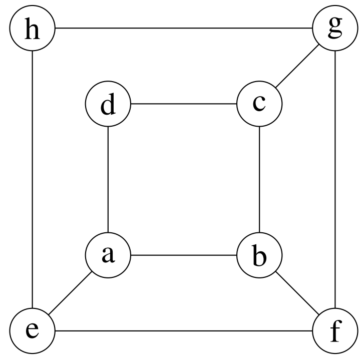
When it is possible to visit more than one vertex, the one that will always be chosen is the one with the minimum index in the ordering of the available vertices, according to the general ordering of the vertices of the graph.

The queue Q begins with the starting vertex and the vertices that are visited are added to the queue. When all the vertices adjacent to the starting one have been visited, they are deleted from the queue and the search continues with the next vertex. The process finishes when the queue is empty.

BFS Algorithm
Imput: $G(V,E)$, $v \in V$
Output: R, a list of visited vertices

```
BFS Algorithm (G, v)
  Start
    Q ← ∅
    R ← [v]
    for w ∈ V
            state [w] ← 0
    endfor
    state [v] ← 1
    add (Q, v)
    While Q ≠ ∅
            w ← first (Q)
            for u adjacent to w
                if state [u] = 0
                    then  add (Q, u)
                          state [u] ← 1
                          add (R, u)
                endif
            endfor
            delete (Q)
    endwhile
    return (R)
  End
```

Note that each vertex is added to the queue once. In addition, when it accesses a vertex of the queue, its list of adjacencies is analysed to look for the vertices not yet visited. In total, $2m$ edges will have been analysed, with m being the size of the graph. Thus, the algorithm will have a complexity of $O(n+m)$.

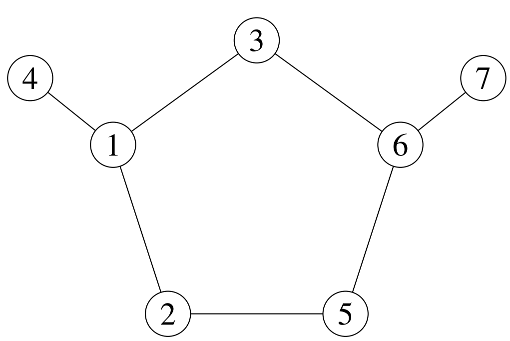


Example 59. Consider the graph represented in the figure.

This table registers the operation of the Breadth First Search algorithm (BFS) for this graph, with starting vertex $v = a$.

Q	Added vertex	Deleted vertex	R
a	a	-	$[a]$
ab	b	-	$[a,b]$
abd	d	-	$[a,b,d]$
$abde$	e	-	$[a,b,d,e]$
bde	-	a	$[a,b,d,e]$
$bdec$	c	-	$[a,b,d,e,c]$
$bdecf$	f	-	$[a,b,d,e,c,f]$
$decf$	-	b	$[a,b,d,e,c,f]$
ecf	-	d	$[a,b,d,e,c,f]$
$ecfh$	h	-	$[a,b,d,e,c,f,h]$
cfh	-	e	$[a,b,d,e,c,f,h]$
$cfhg$	g	-	$[a,b,d,e,c,f,h,g]$
fhg	-	c	$[a,b,d,e,c,f,h,g]$
hg	-	f	$[a,b,d,e,c,f,h,g]$
g	-	h	$[a,b,d,e,c,f,h,g]$
$\varnothing$	-	g	$[a,b,d,e,c,f,h,g]$

Exercise 53. Apply the BFS algorithm to the graph represented in the figure, where the starting vertex is $v = 1$.



Solution: This table registers the operation of the BFS algorithm for this graph, with starting vertex $v = 1$.

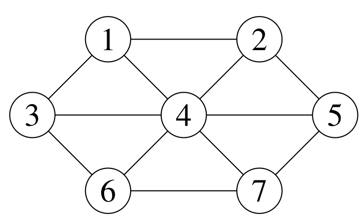
Q	Added vertex	Deleted vertex	R
1	1	-	[1]
12	2	-	[1,2]
123	3	-	[1,2,3]
1234	4	-	[1,2,3,4]
234	-	1	[1,2,3,4]
2345	5	-	[1,2,3,4,5]
345	-	2	[1,2,3,4,5]
3456	6	-	[1,2,3,4,5,6]
456	-	3	[1,2,3,4,5,6]
56	-	4	[1,2,3,4,5,6]
6	-	5	[1,2,3,4,5,6]
67	7	-	[1,2,3,4,5,6,7]
7	-	6	[1,2,3,4,5,6,7]
$\varnothing$	-	7	[1,2,3,4,5,6,7]

□

Exercise 54. Find the list of vertices which are visited by the BFS algorithm in the graph of Exercise 53 (page 56), where 6 is the starting vertex.

Solution: The list of visited vertices is $R = [6,3,5,7,1,2,4]$. □

Exercise 55. Find the list of vertices which are visited by the BFS algorithm in the graph, where 1 is the starting vertex.



Solution: The list of visited vertices is $R = [1,2,3,4,5,6,7]$. □

Exercise 56. Use the BFS algorithm to obtain the list S of edges which are visited in the graph of Exercise 53. Build also the table of operation for the algorithm with the following heading:

Q	Added edge	S

Solution: The table of operation of the algorithm will be:

Q	Added edge	S
1	-	$\varnothing$
12	$\{1,2\}$	$[\{1,2\}]$
123	$\{1,3\}$	$[\{1,2\},\{1,3\}]$
1234	$\{1,4\}$	$[\{1,2\},\{1,3\},\{1,4\}]$
234	-	$[\{1,2\},\{1,3\},\{1,4\}]$
2345	$\{2,5\}$	$[\{1,2\},\{1,3\},\{1,4\},\{2,5\}]$
345	-	$[\{1,2\},\{1,3\},\{1,4\},\{2,5\}]$
3456	$\{3,6\}$	$[\{1,2\},\{1,3\},\{1,4\},\{2,5\},\{3,6\}]$
456	-	$[\{1,2\},\{1,3\},\{1,4\},\{2,5\},\{3,6\}]$
56	-	$[\{1,2\},\{1,3\},\{1,4\},\{2,5\},\{3,6\}]$
6	-	$[\{1,2\},\{1,3\},\{1,4\},\{2,5\},\{3,6\}]$
67	$\{6,7\}$	$[\{1,2\},\{1,3\},\{1,4\},\{2,5\},\{3,6\},\{6,7\}]$
7	-	$[\{1,2\},\{1,3\},\{1,4\},\{2,5\},\{3,6\},\{6,7\}]$
$\varnothing$	-	$[\{1,2\},\{1,3\},\{1,4\},\{2,5\},\{3,6\},\{6,7\}]$

$\square$

Connectivity test

To check if a graph $G=(V,E)$ is connected, one can use, for example, the DFS algorithm. Taking into account that the DFS algorithm gives back the list R of all the accessible vertices from one fixed vertex v. If the list contains all the vertices of the graph, then the graph will be connected. In the opposite case, it will not be connected.

Imput: $G(V,E)$, $v \in V$
Output: *TRUE* if G is connected, *FALSE* otherwise
Algorithm *TestConnection* (G,v)
  Start
  connected $\leftarrow$ TRUE
  $R \leftarrow$ DFS(G,v)
  If $|R| \neq |V|$
    then *connected* $\leftarrow$ FALSE
  endif
  Return (*connected*)
End

Note that the BFS algorithm can also be used as a test of connection.

Exercise 57. Apply a connectivity test to check if the graph G defined by the following adjacency matrix is connected.

$$B=\begin{bmatrix} 0 & 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 & 0 \\ 1 & 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 \end{bmatrix}$$

Solution: Assuming that the vertices of the graph form the set $V = \{a,b,c,d,e\}$, the DFS algorithm is applied:

P	Added vertex	Deleted vertex	R
a	a	-	[a]
ab	b	-	[a,b]
abc	c	-	[a,b,c]
abcd	d	-	[a,b,c,d]
abc	-	d	[a,b,c,d]
abce	e	-	[a,b,c,d,e]
abc	-	e	[a,b,c,d,e]
ab	-	c	[a,b,c,d,e]
a	-	b	[a,b,c,d,e]
∅	-	a	[a,b,c,d,e]

From the test we can deduce that the graph is connected. □

3.4 Spanning tree

Definition 35. A *tree* is a graph satisfying that between any two vertices there is a unique path that connects them.

It is evident that a tree is a connected graph that cannot contain cycles.

Definition 36. A *spanning tree* of a graph is a spanning subgraph that has a tree structure.

Example 60. In Figure 3.1, the graph on the right is a spanning tree of the 3-cube represented on the left.

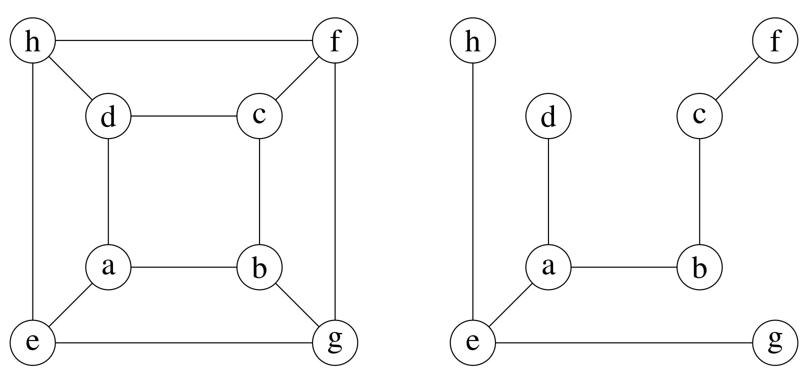


Figure 3.1: The 3-cube graph and a spanning tree.

In the case of connected graphs, the BFS and DFS exploration algorithms allow us to obtain spanning trees.

Exercise 58. Apply the DFS and BFS algorithms to determine two spanning trees of the 3-cube in Example 60.

Solution: The spanning tree on the left is obtained by applying the DFS algorithm, and the one on the right when applying the BFS algorithm. In both cases the starting point was vertex a. □

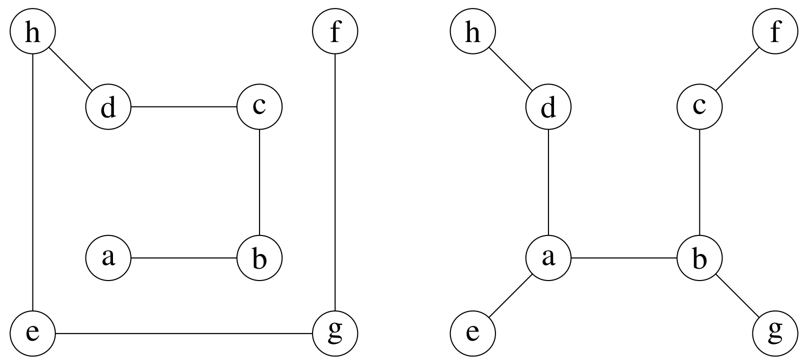


Figure 3.2: Two spanning trees of the 3-cube graph.

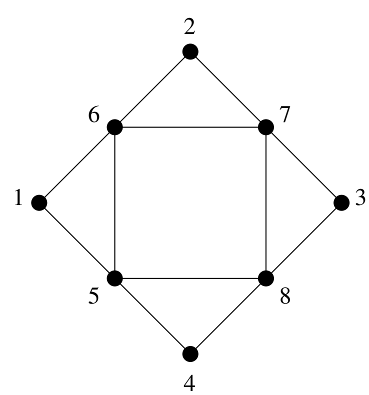
Exercise 59. Let $G = C_4 \odot N_1$.

(a) Draw the line graph of G.

(b) Apply an algorithm to determine a spanning tree of $L(G)$, beginning from a vertex of degree 2.

Solution:

(a) The graph $L(G)$ is

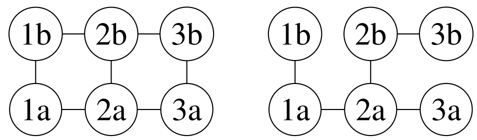


(b) When applying the BFS algorithm to $L(G)$, starting from vertex 1, one obtains the tree formed by edges $\{1,5\}$, $\{1,6\}$, $\{5,4\}$, $\{5,8\}$, $\{6,2\}$, $\{6,7\}$ and $\{8,3\}$.

□

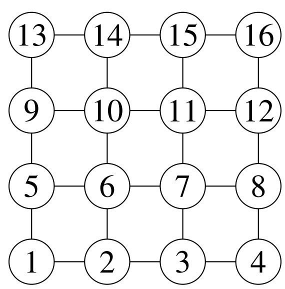
Exercise 60. Let $V(P_3) = \{1,2,3\}$ and let $V(K_2) = \{a,b\}$. Knowing that the central vertex of P_3 is denoted by 2, determine the spanning tree that is obtained by applying the BFS algorithm to the graph $P_3 \square K_2$ starting from vertex $2a$.

Solution: The following figure shows the graph $P_3 \square P_2$ on the left. To the right is the spanning tree obtained by applying the BFS algorithm starting from vertex $2a$.

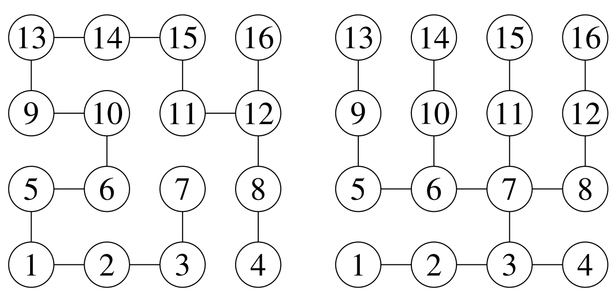


$\square$

Exercise 61. Determine the spanning trees that are obtained by applying the DFS and BFS algorithms to the graph in the figure starting from vertex $v = 7$.



Solution: The following figure shows the spanning trees obtained by applying the DFS and BFS algorithms (starting from vertex 7), respectively.



$\square$

Chapter 4

Distances in graphs

In the context of navigation in networks there is a problem of finding a critical route between two nodes of the network. In this chapter we study this problem and, to this end, define the concept of distance between two vertices of a connected graph and give some basic results related to this concept. Finally, the general problem of finding the distance between a vertex and the rest of the vertices of the graph will be resolved, as well as a path to span this distance, by means of the Dijkstra and Floyd algorithms.

4.1 Concepts and basic results

Definition 37. Given a connected graph G, the *distance*, $d_G(u,v)$, between two vertices u,v is the number of edges in a shortest path connecting them.

When it is clear in which graph the distance between two vertices u and v is calculated, we will write simply $d(u,v)$, in place of $d_G(u,v)$.

In a non-connected case, the distance between two vertices of a single component is defined as in the previous case. In the case of mutually inaccessible vertices, the conventional value ∞ is assigned.

Note that every connected graph $G=(V,E)$ satisfies the requirement that (V,d) is a metric space. That is, the distance $d : V\times V \longrightarrow \mathbb{N}$ satisfies the general properties of a metric for all $u,v,w\in V(G)$:

(i) $d(u,v)\geq 0$ and $d(u,v)=0$ if and only if $u=v$;

(ii) $d(u,v)=d(v,u)$;

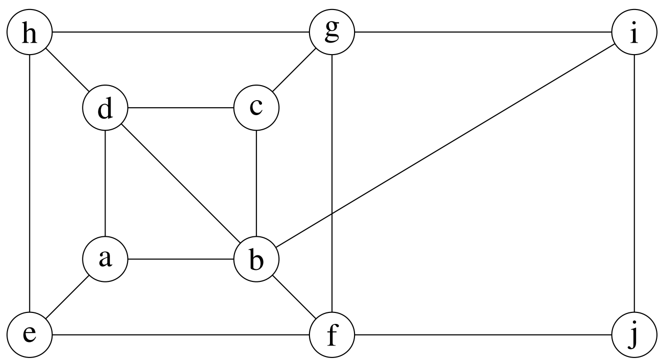
(iii) $d(u,v)\leq d(u,w)+d(w,v)$ (Triangular inequality).

Definition 38. Let $G=(V,E)$ be a connected graph and let $v\in V$.

- The *eccentricity* of v is $\varepsilon(v)=\max\limits_{u\in V}\{d(u,v)\}$.
- The *radius* of G is $r(G)=\min\limits_{v\in V}\{\varepsilon(v)\}$.
- The *diameter* of G is $D(G)=\max\limits_{v\in V}\{\varepsilon(v)\}$.

Example 61. Consider the graph G in the following figure.
In this case we have:

- $d(a,h) = d(a,c) = d(a,f) = d(a,i) = 2$,
- $d(a,g) = d(a,j) = 3$,
- $\varepsilon(a) = 3 = D(G)$,
- $\varepsilon(b) = r(G) = 2$.



Exercise 62. Determine the diameter of the following graphs: K_n, $K_{r,s}$, C_n, P_n.

Solution: $D(K_n) = 1$, $D(K_{r,s}) = 2$, $D(P_n) = n-1$, $D(C_n) = \dfrac{n}{2}$ if n is even and $D(C_n) = \dfrac{n-1}{2}$ if n is odd. $\square$

Proposition 30. *For all non-connected graphs G, G^c is connected and $D(G^c) \leq 2$.*

Proof. Let $G_1, G_2, \ldots, G_k$ be the components of G and let $x, y \in V(G)$ be two arbitrary vertices. If x and y are not adjacent in G, then they are adjacent in G^c. Now, if they are adjacent in G, then there exists a component G_i such that $x, y \in V(G_i)$, and so for every vertex $z \notin V(G_i)$ we have $d_{G^c}(x,z) = 1$ and $d_{G^c}(y,z) = 1$, which implies that $d_{G^c}(x,y) = 2$. Therefore, G^c is connected and $D(G^c) \leq 2$. $\square$

Exercise 63. Let $G = (V,E)$ be a graph of order n such that $\delta(u) + \delta(v) \geq n-1$ for every $u, v \in V$. Prove that G is connected and obtain an upper bound on the diameter of G.

Solution: For every pair of non-adjacent vertices $u, v \in V$ we have that

$$|N(u) \cup N(v)| = |N(u)| + |N(v)| - |N(u) \cap N(v)|.$$

Hence, $|N(u) \cup N(v)| \leq n-2$ and $|N(u)| + |N(v)| = \delta(u) + \delta(v) \geq n-1$.

As a result, $|N(u) \cap N(v)| \geq 1$, for every pair of non-adjacent vertices $u, v \in V$. Therefore, the graph G is connected and $D(G) \leq 2$. $\square$

Proposition 31. *Let G be a connected graph. If $D(G) = 3$, then $2 \leq D(G^c) \leq 3$.*

Proof. Since G is not an empty graph, G^c is not a complete graph. Now, let $u,v \in V$ such that $d_G(u,v) = D(G) = 3$ and let $x \in V$. Hence, $d_{G^c}(u,v) = 1$ and if $x \in N_G[u]$, then $x \in N_{G^c}(v)$, while if $x \notin N_G[u]$, then $x \in N_{G^c}(u)$. Therefore, G^c is connected and $D(G^c) \leq 3$. □

Proposition 32. *Let G be a connected graph. If $D(G) \geq 4$, then $D(G^c) = 2$.*

Proof. Since G is not an empty graph, G^c is not a complete graph. Now, let $x,y \in V$ be two different vertices. If $x \not\sim y$ in G, then $d_{G^c}(x,y) = 1$. If $x \sim y$ in G, then for any pair of vertices $u,v \in V$ such that $d_G(u,v) = D(G) \geq 4$ have that two cases.

- If $\{x,y\} \cap N_G[u] \neq \varnothing$, then $x,y \in N_{G^c}(v)$.
- If $\{x,y\} \cap N_G[u] = \varnothing$, then $x,y \in N_{G^c}(u)$.

Hence, $d_{G^c}(x,y) = 2$. Therefore, we can conclude that G^c is connected and $D(G^c) = 2$. □

Let G be a graph. A subset $S \subseteq V(G)$ is a *connected dominating set* of G if the subgraph induced by S is connected and $N_G(v) \cap S \neq \varnothing$ for every for every vertex $u \in V(G) \setminus S$. The *connected domination number* of G, denoted by $\gamma_c(G)$, is the minimum cardinality among all connected dominating sets of G.

Exercise 64. Show that the following statements hold for any connected graph $G \not\cong K_1$ such that G^c is connected.

(a) $D(G^c) \neq 2$ if and only if $\gamma_c(G) = 2$.

(b) $D(G) = D(G^c) = 3$ if and only if $\gamma_c(G) = \gamma_c(G^c) = 2$.

Solution: Fist, since G and G^c are connected and different from K_1, non of these graphs have isolated vertices, and so the following statemnets hold.

- $\gamma_c(G^c) \geq 2$ and $\gamma_c(G) \geq 2$.
- $D(G) \geq 2$ and $D(G^c) \geq 2$.

With these observations in mind, we proceed to prove the statements.

(a) ($\Leftarrow$) If $\gamma_c(G) = 2$, then for every connected dominating set $\{x,y\}$ we have that x and y are not adjacent in G^c and $N_{G^c}(x) \cap N_{G^c}(y) = \varnothing$, as every vertex is adjacent to x or y in G. Thus, $d_{G^c}(x,y) \geq 3$ and so $D(G^c) \geq 3$.

($\Rightarrow$) If $D(G^c) \neq 2$, then $D(G^c) \geq 3$. Hence, for every pair of vertices x,y such that $d_{G^c}(x,y) = D(G^c) \geq 3$ we have that x and y are adjacent in G and also the following statements hold.

- If $u \in N_{G^c}(x)$, then $u \in N_G(y)$.
- If $u \notin N_{G^c}(x)$, then $u \in N_G(x)$.

Therefore, $\{x,y\}$ is a connected dominating set of G, which implies that $\gamma_c(G) \leq 2$. Since we already know that $\gamma_c(G) \geq 2$, we can conclude that $\gamma_c(G) = 2$. □

(b) By Proposition 32 we already know that if G is a graph with $D(G) \geq 4$, then $D(G^c) = 2$. Hence, according to the conditions of the problem, the following equivalence holds

$$D(G) \neq 2 \quad \text{and} \quad D(G^c) \neq 2 \quad \longleftrightarrow \quad D(G) = D(G^c) = 3.$$

Therefore, by (a) we conclude that (b) holds. □

Proposition 33. *For any connected non-trivial graph G and and any graph H, the corona graph $G \odot H$ has diameter $D(G \odot H) = D(G) + 2$.*

Proof. Let $V = \{v_1, \dots, v_n\}$ be the vertex set of G and let $H_i = (V_i, E_i)$ be the i-th copy of H in $G \odot H$. First, for each $i \in \{1, \dots, n\}$, all the vertices of H_i are adjacent to v_i. Second, for every $a \in V_i$ and $b \in V_j$ we have that

$$d_{G\odot H}(a,b) = d_{G\odot H}(a,v_i) + d_{G\odot H}(v_i,v_j) + d_{G\odot H}(v_j,b) = 1 + d_G(v_i,v_j) + 1.$$

Therefore, since the farthest vertices in $G \odot H$ are not vertices of G, we can conclude that $D(G \odot H) = D(G) + 2$. □

Given two graphs G and H, and a sequence Q of ordered pairs in $V(G) \times V(H)$ of the form $(x_1,y_1), \dots, (x_k,y_k)$, the projection of Q on G, denoted by $P_G(Q)$ is given by the sequence $x_1, \dots, x_k$. Analogously, the projection $P_H(Q)$ of Q on H is given by $y_1, \dots, y_k$. For technical reasons, we will omit the consecutive repetition of vertices in $P_G(Q)$ and $P_H(Q)$. For instance, if Q is $(a,b),(a,d),(c,d),(e,f)$, then $P_G(Q)$ is given by a,c,e while $P_H(Q)$ is given by b,d,f.

In $G\Box H$, the copy of G associated to a vertex $v \in V(H)$ will be denoted by $G\Box\langle v\rangle$, and the copy of H associated to $u \in V(G)$ will be denoted by $\langle u\rangle\Box H$. Notice that

$$G\Box\langle v\rangle \cong \langle V(G) \times \{v\}\rangle \cong G \text{ and } \langle u\rangle\Box H \cong \langle \{u\} \times V(H)\rangle \cong H.$$

Proposition 34. *Let G and H be two connected graphs. For every $(u,v),(x,y) \in V(G\Box H)$,*

$$d_{G\Box H}((u,v),(x,y)) = d_G(u,x) + d_H(v,y).$$

Proof. Taking into account the triangular inequality and that $G\Box\langle v\rangle \cong G$ and $\langle x\rangle\Box H \cong H$, we obtain

$$\begin{aligned} d_{G\Box H}((u,v),(x,y)) &\leq d_{G\Box H}((u,v),(x,v)) + d_{G\Box H}((x,v),(x,y)) \\ &\leq d_{G\Box\langle v\rangle}((u,v),(x,v)) + d_{\langle x\rangle\Box H}((x,v),(x,y)) \\ &= d_G(u,x) + d_H(v,y). \end{aligned}$$

Reciprocally, each edge lying on a shortest path Q from (u,v) to (x,y) either corresponds to an edge lying on the projection of Q on G and to a node in the projection of Q on H, or corresponds to an edge lying on the projection of Q on H and to a node in the projection of Q on G. Hence,

$$\begin{aligned} d_{G\Box H}((u,v),(x,y)) &= |E(Q)| \\ &= |E(P_G(Q))| + |E(P_H(Q))| \\ &\geq d_G(u,x) + d_H(v,y). \end{aligned}$$

Therefore, the result follows. □

Corollary 35. *For any pair of connected graphs G and H,*

$$D(G\square H) = D(G) + D(H).$$

Exercise 65. Let $G = C_4 \odot N_1$. Find the diameter of $G\square L(G)$.

Solution: $D(G\square L(G)) = D(G) + D(L(G)) = 4 + 3 = 7$. □

Exercise 66. Given the graph $G = P_r\square P_s$:

(a) Determine the order, the size and the diameter of G.

(b) Let u and v be two vertices of G such that the distance between them coincides with the diameter of G. Determine the number of paths of minimum length to go from u to v.

Solution:

(a) The order of G is rs, the size is $r(s-1)+s(r-1)$ and the diameter is $r+s-2$.

(b) Let us assume that we draw the graph G so that the copies of P_r are vertical and those of P_s are horizontal. To go from the lower left corner to the upper right we have to take $r+s-2$ steps, of which $r-1$ are vertical and $s-1$ are horizontal. Thus, the number of ways to choose the vertical (horizontal) steps is $\binom{r+s-2}{r-1} = \frac{(r+s-2)!}{(r-1)!(s-1)!} = \binom{r+s-2}{s-1}$.

□

Exercise 67. In chemical graph theory, one of the most important invariants is the Wiener index. Given a connected graph G, the Wiener index, denoted by $W(G)$, is defined as the sum of the distances between all pair of vertices. Hence,

$$W(G) = \frac{1}{2}\sum_{u\in V(G)}\left(\sum_{v\in V(G)} d_G(u,v)\right).$$

Let G and H be two connected graphs. Obtain a formula for $W(G\square H)$ in terms of $|V(G)|$, $|V(H)|$, $W(G)$ and $W(G')$.

Solution: We proceed to show that $W(G\square H) = |V(H)|^2\cdot W(G) + |V(G)|^2\cdot W(H)$.

$$\begin{aligned}
W(G\square H) &= \frac{1}{2}\sum_{(g,h)\in V(G\square H)}\left(\sum_{(g',h')\in V(G\square H)} d_{G\square H}((g,h),(g',h'))\right)\\
&= \frac{1}{2}\sum_{g\in V(G)}\sum_{h\in V(H)}\left(\sum_{g'\in V(G)}\sum_{h'\in V(H)}(d_G(g,g')+d_H(h,h'))\right)\\
&= \frac{1}{2}\sum_{g\in V(G)}\sum_{h\in V(H)}\left(\sum_{g'\in V(G)}\sum_{h'\in V(H)} d_G(g,g')\right)+\frac{1}{2}\sum_{g\in V(G)}\sum_{h\in V(H)}\left(\sum_{g'\in V(G)}\sum_{h'\in V(H)} d_H(h,h')\right)\\
&= \frac{1}{2}|V(H)|^2\sum_{g\in V(G)}\left(\sum_{g'\in V(G)} d_G(g,g')\right)+\frac{1}{2}|V(G)|^2\sum_{h\in V(H)}\left(\sum_{h'\in V(H)} d_H(h,h')\right)\\
&= |V(H)|^2\cdot W(G) + |V(G)|^2\cdot W(H).
\end{aligned}$$

□

Proposition 36. *Let G and H be two connected graphs. For every $(g,h),(g',h') \in V(G\boxtimes H)$,*

$$d_{G\boxtimes H}((g,h),(g',h')) = \max\{d_G(g,g'),d_H(h,h')\}.$$

Proof. Let $g = x_0x_1...x_k = g'$ and $h = y_0y_1...y_l = h'$ be paths of minimum length from g to g' in G and from h to h' in H, respectively. We assume, without loss of generality, that $d_H(h,h') = l \leq k = d_G(g,g') = \max\{d_G(g,g'),d_H(h,h')\}$. Then

$$(g,h) = (x_0,y_0),\ldots,(x_l,y_l)(x_{l+1},y_l)...(x_k,y_l) = (g',h')$$

is a path from (g,h) to (g',h') and, therefore,

$$d_{G\boxtimes H}((g,h),(g',h')) \leq k = \max\{d_G(g,g'),d_H(h,h')\}.$$

Conversely, let Q be a shortest path from (g,h) to (g',h') in $G\boxtimes H$. Then each edge of Q is projected as an edge in G or in H. Thus, $d_G(g,g') \leq |E(P_G(Q))| \leq |E(Q)|$ and $d_H(h,h')\} \leq |E(P_H(Q))| \leq |E(Q)|$. Therefore,

$$\max\{d_G(g,g'),d_H(h,h')\} \leq |E(Q)| = d_{G\boxtimes H}((g,h),(g',h')).$$

□

Corollary 37. *For any pair of connected graphs G and H,*

$$D(G\boxtimes H) = \max\{D(G),D(H)\}.$$

Proposition 38. *Let G be a connected non-trivial graph and let H be a graph. For any pair of different vertices $(g,h),(g',h') \in V(G\circ H)$,*

$$d_{G\circ H}((g,h),(g',h')) = \begin{cases} d_G(g,g') & \text{if} \quad g \neq g', \\ \min\{2,d_H(h,h')\} & \text{if} \quad g = g'. \end{cases}$$

Proof. We differentiate two cases.

Case 1. $g \neq g'$. For any shortest path Q from (g,h) to (g',h'),

$$d_{G\circ H}((g,h),(g',h')) = |E(Q)| \geq |E(P_G(Q))| \geq d_G(g,g').$$

Now, for any shortest path $g = x_0,\ldots,x_k = g'$ we have that

$$(g,h) = (x_0,h),(x_1,h'),\ldots,(x_k,h') = (g',h')$$

is a path from (g,h) to (g',h'), which implies that

$$d_{G\circ H}((g,h),(g',h')) \leq k = d_G(g,g').$$

Therefore, $d_{G\circ H}((g,h),(g',h')) = d_G(g,g')$.

Case 2. $g = g'$. If h and h' are adjacent in H, then (g,h) and (g',h') are adjacent in $G\circ H$. Hence, $d_{G\circ H}((g,h),(g',h')) = 1 = \min\{2,d_H(h,h')\}$.

Now, assume that h and h' are not adjacent in H. In this case, (g,h) and (g',h') are not adjacent in $G\circ H$, and so $d_{G\circ H}((g,h),(g',h')) \geq 2 = \min\{2,d_H(h,h')\}$. Since for any $z \in N_G(g)$ there exists a path $(g,h),(z,h),(g,h')$, we conclude that $d_{G\circ H}((g,h),(g',h')) = 2 = \min\{2,d_H(h,h')\}$. □

Corollary 39. *For any connected non-trivial graph G and any graph H,*

$$D(G \circ H) = \max\{D(G), \min\{2, D(H)\}\}.$$

Observe that if H is not connected, then we are assuming $D(H) = \infty$.

Exercise 68. Let G and H be two connected graphs of order greater than or equal to 2. Find the diameter of the following graphs:

(a) $(G \odot H)\Box(H \odot G)$

(b) $(G \odot H) \boxtimes (H \odot G)$

(c) $(G \odot H)\Box G$

(d) $(G \odot H) \boxtimes G$

(e) $G \boxtimes (G \circ H)$

Solution:

(a) $D((G \odot H)\Box(H \odot G)) = D(G \odot H) + D(H \odot G) = D(G) + D(H) + 4.$

(b) $D((G \odot H) \boxtimes (H \odot G)) = \max\{D(G \odot H), D(H \odot G)\} = \max\{D(G) + 2, D(H) + 2\}.$

(c) $D((G \odot H)\Box G) = 2D(G) + 2.$

(d) $D((G \odot H) \boxtimes G) = D(G) + 2.$

(e) $D(G \boxtimes (G \circ H)) = \max\{D(G), \min\{2, D(H)\}\}.$

□

Notice that if G and H are not complete graphs, then

$$D((G \circ H) \boxtimes (H \circ G)) = \max\{D(G), D(H)\} = D(G \boxtimes H).$$

Definition 39. The *centre* of a graph G is the set

$$\mathcal{C}(G) = \{v \in V(G) : \varepsilon(v) = r(G)\}.$$

Theorem 40. *The center of a tree consists of either one single vertex or a pair of adjacent vertices.*

Proof. Let T be a tree. The result is trivial for $T = K_1$ and $T = K_2$. Suppose that $n(T) \geq 3$. Clearly, if $\varepsilon(v) = d_T(u, v)$, then $\delta(u) = 1$. Thus, if T' is the tree obtained from T by removing the vertices of degree one, then the eccentricity of each vertex of T' is exactly one less than the eccentricity of the same vertex in T. Thus, $\mathcal{C}(T) = \mathcal{C}(T')$. If the process of removing nodes is repeated, then we obtain trees having the same center as T. Since the order of T is finite, we obtain a subtree of T which is either K_1 or K_2. In either case, the vertices in this last tree constitute the center of T. So, the center of T consists of either one single vertex or a pair of adjacent vertices. □

Exercise 69. Give three examples of graphs $G = (V,E)$ such that $\mathcal{C}(G) = V$.

Solution: We take as examples the complete graphs $G = K_n$, the hypercubes $G = Q_k$, and the complete bipartite graphs $G = K_{r,s}$ with $r,s \geq 2$. $\square$

Proposition 41. *For any connected graph G and any graph H,*

$$\mathcal{C}(G \odot H) = \mathcal{C}(G).$$

Proof. As above, let $V = \{v_1,\ldots,v_n\}$ be the set of vertices of G and let $H_i = (V_i,E_i)$ be the i-th copy of H in $G \odot H$. Therefore, for each $i \in \{1,\ldots,n\}$, all the vertices of H_i are adjacent to v_i.

We know that for every $a \in V_i$ and $b \in V_j$,

$$d_{G\odot H}(a,b) = d_{G\odot H}(a,v_i) + d_{G\odot H}(v_i,v_j) + d_{G\odot H}(v_j,b) = 1 + d_G(v_i,v_j) + 1.$$

To obtain the result it is sufficient to observe that for every $v_i \in V$ we have $\varepsilon_{G\odot H}(v_i) = \varepsilon_G(v_i) + 1$ and for every $u_i \in V_i$ we have $\varepsilon_{G\odot H}(u_i) = \varepsilon_G(v_i) + 2$. Thus, $\mathcal{C}(G \odot H) = \mathcal{C}(G)$. $\square$

Exercise 70. Let G_k be a graph of the family defined by the following recurrence:

- $G_k = G_{k-1} \Box K_3$, for every integer $k \geq 2$,
- $G_1 = K_3$.

Find the order, size and diameter of G_k. Find the diameter of the following graphs for any $k \geq 2$.

- $G_k \circ G_k$
- $G_k \odot G_k$
- $G_k \boxtimes (G_{k-1} \Box C_4)$.

Solution: For G_k we have

- $n(G_k) = 3n(G_{k-1}) = 3^k$.
- $m(G_k) = \frac{3^k \cdot 2k}{2} = k \cdot 3^k$.
- $D(G_k) = D(G_{k-1}) + D(K_3) = k$.

Furthermore,

- $D(G_k \circ G_k) = D(G_k) = k$.
- $D(G_k \odot G_k) = D(G_k) + 2 = k + 2$.
- $D(G_k \boxtimes (G_{k-1} \Box C_4)) = \max\{D(G_k), D(G_{k-1}) + 2\} = k + 1$.

$\square$

4.2 Distances and bipartite graphs

It is possible to characterise bipartite graphs from the length of the cycles.

Theorem 42. *A graph is bipartite if and only if it does not have cycles of odd length.*

Proof. If G is bipartite, then each cycle $v_1, v_2, v_3, \dots, v_k, v_1$ has all the vertices of odd-numbered subscript in one of the sets of the bipartition and the vertices of even-numbered subscript in the another set. Therefore, the length k is even.

Now our hypothesis is that G does not have cycles of odd length. We consider, without loss of generality, that G is connected (if G it is not connected, we analyse each component separately). For $u_0 \in V$ we define the sets

$$V_1 = \{u_0\} \cup \{v \in V : d(u_0, v) \text{ is even}\}, \quad V_2 = V - V_1.$$

These sets determine a partition of the set of vertices of the graph.

Let $a = \{u, v\} \in E$ and we assume that $u, v \in V_1$ or $u, v \in V_2$. Let R_1 be an $u_0 - u$ path of minimum length and R_2 an $u_0 - v$ path of minimum length, both with even or odd lengths, by definition of V_1 and V_2. Travelling the previous paths from u_0, a vertex w_0 will be reached that is the last one that shares paths R_1 and R_2. For the subpaths $S_1 : w_0 - u$ on R_1 and $S_2 : w_0 - v$ on R_2 we have that $\ell(R_1) = d(u_0, w_0) + \ell(S_1)$ and $\ell(R_2) = d(u_0, w_0) + \ell(S_2)$. Thus $\ell(S_2) - \ell(S_1)$ is even and, in consequence, $\ell(S_1), \ell(S_2)$ have to be of the same parity. Therefore, with S_1, S_2 and the edge $\{u, v\}$ we form a cycle of odd length $\ell(S_1) + 1 + \ell(S_2)$, which contradicts the hypothesis. □

Corollary 43. *All trees are bipartite graphs.*

Exercise 71. Let G and H be two graphs. Prove that $G \square H$ is bipartite if and only if G and H are bipartite.

Solution: By Theorem 42 we know that a graph is bipartite if and only if it does not have cycles of odd length. First we will assume that $G \square H$ is bipartite. In such a case $G \square H$ does not have cycles of odd length and, in consequence, for every vertex $y \in V(H)$ the subgraph of $G \square H$ induced by $V(G) \times \{y\}$ does not have cycles of odd length. As the subgraph of $G \square H$ induced by $V(G) \times \{y\}$ is an isomorph to G, we can conclude that G is bipartite. By analogy we deduce that H is bipartite.

Now we will assume that G and H are bipartite. We will label the vertices of G with zeros and ones so that adjacent vertices have different labels. Now we consider a copy of H, denoted by $H_{0,1}$, in which we will label the vertices with zeros and ones so that adjacent vertices have different labels. We also label with zeros and ones another copy of H, denoted by $H_{1,0}$, in which a vertex has label one if and only if it has label zero in $H_{0,1}$. Hence, we can label the vertices of $G \square H$ with zeros and ones in the following way. If $x \in V(G)$ has label zero, then the labelling of the vertices in $\{x\} \times V(H)$ correspond to the labelling assigned to $H_{0,1}$, while if $x \in V(G)$ has label one, then the labelling of the vertices in $\{x\} \times V(H)$ correspond to the labelling assigned to $H_{1,0}$. In this manner, if two vertices are adjacent in $G \square H$, then they have different label. Therefore, $G \square H$ is bipartite. □

Exercise 72. Let G and H be two graphs. Determine a necessary and sufficient condition so that the following graphs are bipartite.

(a) $G \odot H$

(b) $G + H$

(c) $G \boxtimes H$

(d) $G \circ H$.

Solution: From Theorem 42 the following consequences can be derived.

(a) $G \odot H$ is bipartite if and only if G is bipartite and H is empty.

(b) $G + H$ is bipartite if and only if G and H are empty.

(c) $G \boxtimes H$ is bipartite if and only if G is bipartite and H is empty or vice versa.

(d) $G \circ H$ is bipartite if and only if G is bipartite and H is empty.

□

4.3 The shortest path problem

Consider a network of roads that connect a set of cities. The most natural algorithmic problem in this case is to find a route of minimum length between a pair of cities. The problem is solved by considering a graph whose vertices are the cities and whose edges are the pairs of cities with direct connection by road. In this case we need to use an additional attribute for the edges; the distance in kilometres between the different cities with a direct connection.

The distance between two vertices in an unweighted graph can be found, in an efficient way, if one applies the Breadth First Search (BFS) algorithm. If the graph is weighted, then the distance between two vertices cannot be calculated directly and it will be necessary to apply other algorithms.

The *Dijkstra algorithm* finds the distance between two vertices by constructing a tree from the initial vertex u_0 to each one of the other vertices of the graph.

The *Floyd algorithm* finds the distance between all the pairs of vertices of a graph.

Definition 40. A *weighted* graph is a pair (G, w) where $G = (V, E)$ is a graph and w is a map $w : E \to \mathbb{R}$ that assigns weights to the edges of the graph.

In a communications network, the allocation to each edge of a weight that is indicative of the time that it costs to visit the edge in question can be of interest; or kilometres, or corresponding economic costs in other respects that are related to the problem and the model that has been built. Usually the weights are positive or zero, but situations in which they are negative can also be considered.

Definition 41. Given a weighted graph (G, w) and a path $C : v_0, v_1, \ldots, v_k$ the *weight of the path C* is defined as

$$w(C) = \sum_{i=1}^{k} w(v_{i-1}, v_i).$$

And the *distance* between two vertices $u, v \in G$ as

$$d_G(u,v) = \min\{w(C) : \ C \text{ is a } u - v \text{ path } \}.$$

If the graph G is not weighted (or all the weights are equal to 1), then this definition coincides with the one given previously (Definition 37).

As we will see below, in the case of mutually inaccessible vertices we will assign the conventional value ∞ to this distance.

The shortest path problem admits several variants that can be solved by using adaptations of the same algorithms:

(1) Shortest path from an initial vertex (*Single Source Shortest Path*): given (G,w) and $s \in V$, look for $d(s,v)$ for all $v \in V$.

(2) Shortest path to a destination vertex (*Single Destination Shortest Path*): given (G,w) and $t \in V$, look for $d(v,t)$ for all $v \in V$.

(3) Shortest path between a pair of vertices (*Single Pair Shortest Path*): given (G,w) and $s,t \in V$, look for $d(s,t)$.

(4) Shortest path between all the pairs of vertices (*All Pairs Shortest Path*): given (G,w), look for $d(u,v)$ for all $u,v \in V$.

Next, we will study the specific algorithms to solve variants (1) and (4). The solution of variants (2) and (3) will be left as an exercise.

Dijkstra's Algorithm

Dijkstra's algorithm can be applied on a weighted graph (or digraph) and it will calculate the distance from an initial vertex s to the rest of vertices of the graph. At each step, it will label the vertices with $(dist(u), v)$ where $dist(u)$ is the current minimum distance of the vertices s to the vertices u, and v is the predecessor of u on the shortest path that joins s and u.

Necessary structures for the formulation of the algorithm:

- A weighted graph (G,w) represented by means of a list of adjacencies.
- A set U of vertices that have been visited, in the order in which it has been done.
- A table of distances, $dist(\cdot)$, indexed by the vertices of G, that registers the distance from the initial vertex to the visited vertices.
- At the end, a table $dist(\cdot)$ registering the distance from the initial vertex to the rest of the vertices.

Imput: (G,w) of order n and an initial vertex s.
Output: The distance, $dist(\cdot)$, from s to the rest of vertices.
Dijkstra algorithm(G,s)
    Start
        $U \leftarrow \varnothing$

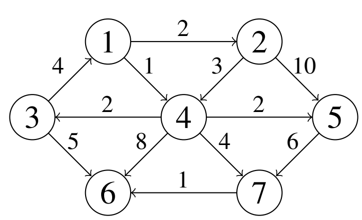
for $v \in V \backslash \{s\}$
$\quad dist(v) \leftarrow \infty$
$\quad$ Label v with $(dist(v), s)$
endfor
$dist(s) \leftarrow 0$
Label s with $(dist(s), s)$
For $i \leftarrow 0$ until $\leftarrow n-1$
$\quad u_i$ Vertex reached $\min_{v \in V-U}\{dist(v)\}$
$\quad U \leftarrow U \cup \{u_i\}$
$\quad$ For $v \in V-U$ adjacent to u_i
$\quad\quad$ If $dist(u_i) + w(u_i, v) < dist(v)$
$\quad\quad\quad$ then $dist(v) \leftarrow dist(u_i) + w(u_i, v)$
$\quad\quad\quad$ Label v with $(dist(v), u_i)$
$\quad\quad$ endif
$\quad$ endfor
endfor
Return $(dist)$
End

When it is possible to visit more than one vertex, always choose the one with the minimum index in the ordering of the available vertices.

At each step it establishes the distance of one of the vertices of the graph. Thus, after n steps it will have calculated the distance to all vertices of the graph.

The algorithm can be used to obtain a path of minimum length between the initial vertex and any other vertex, since after applying the algorithm, all vertices v have a label $(dist(v), u_i)$ associated, which is indicative of the distance to vertex v from the starting vertex, $dist(v) = d(s, v)$, and of the path followed to calculate the distance. If $v \neq s$ obtains a path $s - v$ of minimum length with $s = q_0, q_1, \ldots, q_k = v$, where the q_i are labelled with $(dist(q_i), q_{i-1})$ for $i = 1, \ldots, k$.

Example 62. Consider the digraph defined by the following chart:



We will do a simulation of the algorithm by means of a table where the columns correspond to the vertices of the graph. The start of the algorithm will be expressed in the following way:

1	2	3	4	5	6	7
(0,1)	(∞,1)	(∞,1)	(∞,1)	(∞,1)	(∞,1)	(∞,1)

In this case, in the Dijkstra algorithm table for the digraph the visited vertex is denoted with an asterisk. In the pairs $(dist(v), 1)$ we have

$$dist(v) = \min\{dist(1) + w(1, v), dist(v)\} = \min\{0 + w(1, v), dist(v)\} :$$

1	2	3	4	5	6	7
(0,1)	(∞,1)	(∞,1)	(∞,1)	(∞,1)	(∞,1)	(∞,1)
(0,1)*	(2,1)	(∞,1)	(1,1)	(∞,1)	(∞,1)	(∞,1)

For the non-visited vertices, v, we have $\min\{dist(v)\} = 1$, and this minimum is reached in vertex 4. Therefore, for each non-visited v we have

$$dist(v) = \min\{dist(4) + w(4,v), dist(v)\} = \min\{1 + w(4,v), dist(v)\} :$$

1	2	3	4	5	6	7
(0,1)	(∞,1)	(∞,1)	(∞,1)	(∞,1)	(∞,1)	(∞,1)
(0,1)*	(2,1)	(∞,1)	(1,1)	(∞,1)	(∞,1)	(∞,1)
(0,1)	(2,1)	(3,4)	(1,1)*	(3,4)	(9,4)	(5,4)

Now, for the non-visited vertices, v, we have $\min\{dist(v)\} = 2$, and this minimum is reached in vertex 2. Therefore, for each v not visited we have:

$$dist(v) = \min\{dist(2) + w(2,v), dist(v)\} = \min\{2 + w(2,v), dist(v)\}$$

1	2	3	4	5	6	7
(0,1)	(∞,1)	(∞,1)	(∞,1)	(∞,1)	(∞,1)	(∞,1)
(0,1)*	(2,1)	(∞,1)	(1,1)	(∞,1)	(∞,1)	(∞,1)
(0,1)	(2,1)	(3,4)	(1,1)*	(3,4)	(9,4)	(5,4)
(0,1)	(2,1)*	(3,4)	(1,1)	(3,4)	(9,4)	(5,4)

For the non-visited vertices, v, we have $\min\{dist(v)\} = 3$, and this minimum is reached in vertices 3 and 5. We take vertex 3 because we are following the order of the labels. Therefore, for each v not visited we have:

$$dist(v) = \min\{dist(3) + w(3,v), dist(v)\} = \min\{3 + w(3,v), dist(v)\}.$$

1	2	3	4	5	6	7
(0,1)	(∞,1)	(∞,1)	(∞,1)	(∞,1)	(∞,1)	(∞,1)
(0,1)*	(2,1)	(∞,1)	(1,1)	(∞,1)	(∞,1)	(∞,1)
(0,1)	(2,1)	(3,4)	(1,1)*	(3,4)	(9,4)	(5,4)
(0,1)	(2,1)*	(3,4)	(1,1)	(3,4)	(9,4)	(5,4)
(0,1)	(2,1)	(3,4)*	(1,1)	(3,4)	(8,3)	(5,4)

Now, for the non-visited vertices, v, we have $\min\{dist(v)\} = 3$, and this minimum is reached in vertex 5. Therefore, for each v not visited we have:

$$dist(v) = \min\{dist(5) + w(5,v), dist(v)\} = \min\{3 + w(5,v), dist(v)\}.$$

1	2	3	4	5	6	7
(0,1)	(∞,1)	(∞,1)	(∞,1)	(∞,1)	(∞,1)	(∞,1)
(0,1)*	(2,1)	(∞,1)	(1,1)	(∞,1)	(∞,1)	(∞,1)
(0,1)	(2,1)	(3,4)	(1,1)*	(3,4)	(9,4)	(5,4)
(0,1)	(2,1)*	(3,4)	(1,1)	(3,4)	(9,4)	(5,4)
(0,1)	(2,1)	(3,4)*	(1,1)	(3,4)	(8,3)	(5,4)
(0,1)	(2,1)	(3,4)	(1,1)	(3,4)*	(8,3)	(5,4)

In this case, for the non-visited vertices, v, we have $\min\{dist(v)\} = 5$, and this minimum is reached in vertex 7. Therefore, for each non-visited vertex v we have:

$$dist(v) = \min\{dist(7) + w(7,v), dist(v)\} = \min\{5 + w(7,v), dist(v)\}.$$

1	2	3	4	5	6	7
(0,1)	(∞,1)	(∞,1)	(∞,1)	(∞,1)	(∞,1)	(∞,1)
(0,1)*	(2,1)	(∞,1)	(1,1)	(∞,1)	(∞,1)	(∞,1)
(0,1)	(2,1)	(3,4)	(1,1)*	(3,4)	(9,4)	(5,4)
(0,1)	(2,1)*	(3,4)	(1,1)	(3,4)	(9,4)	(5,4)
(0,1)	(2,1)	(3,4)*	(1,1)	(3,4)	(8,3)	(5,4)
(0,1)	(2,1)	(3,4)	(1,1)	(3,4)*	(8,3)	(5,4)
(0,1)	(2,1)	(3,4)	(1,1)	(3,4)	(6,7)	(5,4)*

Finally, the only non-visited vertex is 6:

1	2	3	4	5	6	7
(0,1)	(∞,1)	(∞,1)	(∞,1)	(∞,1)	(∞,1)	(∞,1)
(0,1)*	(2,1)	(∞,1)	(1,1)	(∞,1)	(∞,1)	(∞,1)
(0,1)	(2,1)	(3,4)	(1,1)*	(3,4)	(9,4)	(5,4)
(0,1)	(2,1)*	(3,4)	(1,1)	(3,4)	(9,4)	(5,4)
(0,1)	(2,1)	(3,4)*	(1,1)	(3,4)	(8,3)	(5,4)
(0,1)	(2,1)	(3,4)	(1,1)	(3,4)*	(8,3)	(5,4)
(0,1)	(2,1)	(3,4)	(1,1)	(3,4)	(6,7)	(5,4)*
(0,1)	(2,1)	(3,4)	(1,1)	(3,4)	(6,7)*	(5,4)

From the last row of the table we can deduce that the distance of vertex 1 to the remaining vertices is $d(1,1) = 0, d(1,2) = 2, d(1,3) = 3, d(1,4) = 1, d(1,5) = 3, d(1,6) = 6, d(1,7) = 5$.

Analysis of the Dijkstra algorithm

To analyse this algorithm we will divide it into two parts:

1. A table of size n will be started with complexity of $O(n)$.
2. The main loop executes n times. In the i-th step it calculates the minimum of a list that contains $n-i$ elements. This can be done with $n-i$ comparisons.

   In the most internal loop the labels of the vertices which are adjacent to the analysed vertex are updated. The maximum number of adjacent vertices that are updated in step i-th is equal to $n-i-1$.

   In summary, in the main loop it performs

   $$\sum_{i=0}^{n-1}(n-i) + \sum_{i=0}^{n-1}(n-i-1) = n^2$$

   elementary operations. This gives a complexity of $O(n^2)$.

The entire algorithm will have a complexity of $\max\{O(n), O(n^2)\} = O(n^2)$, independently of the number of edges of the graph.

Example 63. The following matrix is the adjacency matrix of a weighted graph with vertices A, B, C, D, E and F.

$$\begin{pmatrix} 0 & 8 & 0 & 0 & 5 & 0 \\ 8 & 0 & 7 & 2 & 2 & 0 \\ 0 & 7 & 0 & 8 & 0 & 3 \\ 0 & 2 & 8 & 0 & 0 & 4 \\ 5 & 2 & 0 & 0 & 0 & 9 \\ 0 & 0 & 3 & 4 & 9 & 0 \end{pmatrix}$$

The following table shows the simulation of the algorithm starting from vertex A:

A	B	C	D	E	F
(0,A)	(∞,A)	(∞,A)	(∞,A)	(∞,A)	(∞,A)
(0,A)*	(8,A)	(∞,A)	(∞,A)	(5,A)	(∞,A)
(0,A)	(7,E)	(∞,A)	(∞,A)	(5,A)*	(14,E)
(0,A)	(7,E)*	(14,B)	(9,B)	(5,A)	(14,E)
(0,A)	(7,E)	(14,B)	(9,B)*	(5,A)	(13,D)
(0,A)	(7,E)	(14,B)	(9,B)	(5,A)	(13,D)*
(0,A)	(7,E)	(14,B)*	(9,B)	(5,A)	(13,D)

From the last row of the table we can deduce that the distance from vertex A to the rest of vertices is: $d(A,B) = 7$ and it arrives via vertex E; $d(A,C) = 14$ and a shortest path is A,E,B,C; $d(A,D) = 9$ and a shortest path is A,E,B,D; $d(A,E) = 5$ and it arrives directly from E; $d(A,F) = 13$ and a shortest path is A,E,B,D,F.

The simulation of the algorithm starting from vertex C is shown in the following table.

A	B	C	D	E	F
(∞,C)	(∞,C)	(0,C)	(∞,C)	(∞,C)	(∞,C)
(∞,C)	(7,C)	(0,C)*	(8,C)	(∞,C)	(3,C)
(∞,C)	(7,C)	(0,C)	(7,F)	(12,F)	(3,C)*
(15,B)	(7,C)*	(0,C)	(7,F)	(9,B)	(3,C)
(15,B)	(7,C)	(0,C)	(7,F)*	(9,B)	(3,C)
(14,E)	(7,C)	(0,C)	(7,F)	(9,B)*	(3,C)
(14,E)*	(7,C)	(0,C)	(7,F)	(9,B)	(3,C)

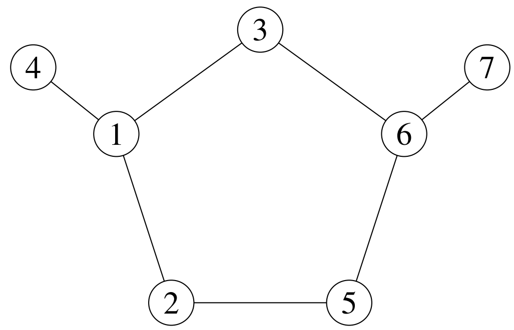
From the last row of the table we can deduce that the distance from vertex C to the rest of vertices is: $d(C,A) = 14$ and a shortest path is C,B,E,A; $d(C,B) = 7$ and it arrives directly from C; $d(C,D) = 7$ and a shortest path is C,F,D; $d(C,E) = 9$ and a shortest path is C,B,E; $d(C,F) = 3$ and it arrives directly from C.

Shortest path in a non-weighted graph

If the graph is not weighted (or all the edges have weight 1) then the BFS algorithm (Breadth First Search) can be used to calculate the distance between an initial vertex and the remaining vertices of the graph.

The data structures needed to formulate the algorithm are the same as in the BFS. In this case, we add a table $dist(\cdot)$ that stores the distances from the initial vertex to the remaining vertices.

Example 64. Consider the non-weighted graph represented in the figure.



The table registers the operation of the algorithm for this graph, with starting vertex $s = 1$.

Q	Added vertex	Deleted vertex	*dist*
1	1	-	[0,∞,∞,∞,∞,∞,∞]
12	2	-	[0,1,∞,∞,∞,∞,∞]
123	3	-	[0,1,1,∞,∞,∞,∞]
1234	4	-	[0,1,1,1,∞,∞,∞]
234	-	1	[0,1,1,1,∞,∞,∞]
2345	5	-	[0,1,1,1,2,∞,∞]
345	-	2	[0,1,1,1,2,∞,∞]
3456	6	-	[0,1,1,1,2,2,∞]
456	-	3	[0,1,1,1,2,2,∞]
56	-	4	[0,1,1,1,2,2,∞]
6	-	5	[0,1,1,1,2,2,∞]
67	7	-	[0,1,1,1,2,2,3]
7	-	6	[0,1,1,1,2,2,3]
∅	-	7	[0,1,1,1,2,2,3]

If we compare this algorithm with the Dijkstra algorithm applied to a non-weighted graph, you can observe that while Dijkstra has a complexity of $O(n^2)$, the BFS has a complexity of $O(n+m)$. For low-density graphs (few edges), the BFS algorithm is more efficient than the Dijkstra algorithm.

Floyd's algorithm

The problem of looking for the shortest paths between all the pairs of vertices of a graph can be resolved if you apply the Dijkstra algorithm n times:

Input : (G, w) of order n

Output: The distance, $d(\cdot,\cdot)$, between all the pairs of vertices with a complexity of $O(n^3)$.

Another alternative is to use a specific algorithm, of comparable efficiency to the Dijkstra algorithm, but with better performance for dense graphs. This is the *Floyd algorithm*. Floyd's algorithm considers the ordered vertices and, in the k-th step, compares the weight of the path obtained up to the moment using the previous $k-1$ vertices, with the path obtained by adding the k-th vertex. It labels the vertices $V = \{1, 2, 3, \ldots, n\}$ and uses a matrix d_{ij} $(1 \leq i, j \leq n)$ to store the distances.

Imput: (G, w) of order n
Output: The distance, $d(\cdot,\cdot)$, between all the pairs of vertices.
Algorithm *Floyd*(G)
  Start
    For $i \leftarrow 1$ until n
      For $j \leftarrow 1$ until n
        If $i = j$ then $d^0_{ij} \leftarrow 0$ endif
        If $(i, j) \in E$ then $d^0_{ij} \leftarrow w(i, j)$ endif
        If $(i, j) \notin E$ then $d^0_{ij} \leftarrow \infty$ endif
      endfor
    endfor
    For $k \leftarrow 1$ until n
      For $i \leftarrow 1$ until n
        For $j \leftarrow 1$ until n
          $d^k_{ij} \leftarrow \min(d^{k-1}_{ij}, d^{k-1}_{ik} + d^{k-1}_{kj})$
        endfor
      endfor
    endfor
    Return (d^n_{ij})
  End

Example 65. The following matrix represents the distances between different cities that are directly connected by road.

$$\begin{pmatrix} 0 & 96 & - & 56 & 105 \\ 96 & 0 & - & 157 & - \\ - & - & 0 & 118 & 91 \\ 56 & 157 & 118 & 0 & - \\ 105 & - & 91 & - & 0 \end{pmatrix}$$

If the Floyd algorithm is applied to the corresponding graph we obtain the series of two-dimensional matrices:

$$d^0 = \begin{pmatrix} 0 & 96 & \infty & 56 & 105 \\ 96 & 0 & \infty & 157 & \infty \\ \infty & \infty & 0 & 118 & 91 \\ 56 & 157 & 118 & 0 & \infty \\ 105 & \infty & 91 & \infty & 0 \end{pmatrix} \quad d^1 = \begin{pmatrix} 0 & 96 & \infty & 56 & 105 \\ 96 & 0 & \infty & 152 & 201 \\ \infty & \infty & 0 & 118 & 91 \\ 56 & 152 & 118 & 0 & 161 \\ 105 & 201 & 91 & 161 & 0 \end{pmatrix}$$

Using the first row and the first column of d^0 we have calculated the matrix d^1 by the formula $d^1_{ij} = \min(d^0_{ij}, d^0_{i1} + d^0_{1j})$. Analogously, we use the second row and the second column of d^1 to calculate the matrix d^2 by the formula $d^2_{ij} = \min(d^1_{ij}, d^1_{i2} + d^1_{2j})$, and so on successively.

$$d^2 = \begin{pmatrix} 0 & 96 & \infty & 56 & 105 \\ 96 & 0 & \infty & 152 & 201 \\ \infty & \infty & 0 & 118 & 91 \\ 56 & 152 & 118 & 0 & 161 \\ 105 & 201 & 91 & 161 & 0 \end{pmatrix} \quad d^3 = \begin{pmatrix} 0 & 96 & \infty & 56 & 105 \\ 96 & 0 & \infty & 152 & 201 \\ \infty & \infty & 0 & 118 & 91 \\ 56 & 152 & 118 & 0 & 161 \\ 105 & 201 & 91 & 161 & 0 \end{pmatrix}$$

$$d^4 = \begin{pmatrix} 0 & 96 & 174 & 56 & 105 \\ 96 & 0 & 270 & 152 & 201 \\ 174 & 270 & 0 & 118 & 91 \\ 56 & 152 & 118 & 0 & 161 \\ 105 & 201 & 91 & 161 & 0 \end{pmatrix} \quad d^5 = \begin{pmatrix} 0 & 96 & 174 & 56 & 105 \\ 96 & 0 & 270 & 152 & 201 \\ 174 & 270 & 0 & 118 & 91 \\ 56 & 152 & 118 & 0 & 161 \\ 105 & 201 & 91 & 161 & 0 \end{pmatrix}$$

Example 66. If we apply the Floyd algorithm to the weighted graph in Example 63 we obtain the following matrices.

$$d^0 = \begin{pmatrix} 0 & 8 & \infty & \infty & 5 & \infty \\ 8 & 0 & 7 & 2 & 2 & \infty \\ \infty & 7 & 0 & 8 & \infty & 3 \\ \infty & 2 & 8 & 0 & \infty & 4 \\ 5 & 2 & \infty & \infty & 0 & 9 \\ \infty & \infty & 3 & 4 & 9 & 0 \end{pmatrix} \qquad d^1 = \begin{pmatrix} 0 & 8 & \infty & \infty & 5 & \infty \\ 8 & 0 & 7 & 2 & 2 & \infty \\ \infty & 7 & 0 & 8 & \infty & 3 \\ \infty & 2 & 8 & 0 & \infty & 4 \\ 5 & 2 & \infty & \infty & 0 & 9 \\ \infty & \infty & 3 & 4 & 9 & 0 \end{pmatrix}$$

$$d^2 = \begin{pmatrix} 0 & 8 & 15 & 10 & 5 & \infty \\ 8 & 0 & 7 & 2 & 2 & \infty \\ 15 & 7 & 0 & 8 & 9 & 3 \\ 10 & 2 & 8 & 0 & 4 & 4 \\ 5 & 2 & 9 & 4 & 0 & 9 \\ \infty & \infty & 3 & 4 & 9 & 0 \end{pmatrix} \qquad d^3 = \begin{pmatrix} 0 & 8 & 15 & 10 & 5 & 18 \\ 8 & 0 & 7 & 2 & 2 & 10 \\ 15 & 7 & 0 & 8 & 9 & 3 \\ 10 & 2 & 8 & 0 & 4 & 4 \\ 5 & 2 & 9 & 4 & 0 & 9 \\ 18 & 10 & 3 & 4 & 9 & 0 \end{pmatrix}$$

$$d^4 = \begin{pmatrix} 0 & 8 & 15 & 10 & 5 & 14 \\ 8 & 0 & 7 & 2 & 2 & 6 \\ 15 & 7 & 0 & 8 & 9 & 3 \\ 10 & 2 & 8 & 0 & 4 & 4 \\ 5 & 2 & 9 & 4 & 0 & 8 \\ 14 & 6 & 3 & 4 & 8 & 0 \end{pmatrix} \qquad d^5 = \begin{pmatrix} 0 & 7 & 14 & 9 & 5 & 13 \\ 7 & 0 & 7 & 2 & 2 & 6 \\ 14 & 7 & 0 & 8 & 9 & 3 \\ 9 & 2 & 8 & 0 & 4 & 4 \\ 5 & 2 & 9 & 4 & 0 & 8 \\ 13 & 6 & 3 & 4 & 8 & 0 \end{pmatrix}$$

$$d^6 = \begin{pmatrix} 0 & 7 & 14 & 9 & 5 & 13 \\ 7 & 0 & 7 & 2 & 2 & 6 \\ 14 & 7 & 0 & 7 & 9 & 3 \\ 9 & 2 & 7 & 0 & 4 & 4 \\ 5 & 2 & 9 & 4 & 0 & 8 \\ 13 & 6 & 3 & 4 & 8 & 0 \end{pmatrix}$$

Thus, for example, the distance from A to D is $d(A,D) = 9$, and the distance from C to E is $d(C,E) = 9$. Note that the diameter of the weighted graph is 14 and the radius is 7.

Analysis of the Floyd algorithm

Floyd's algorithm is very easy to analyse. Basically it has two parts. The start of the distance matrix and the calculation of distances.

The start has a complexity of $O(n^2)$. The calculation of distances has a complexity of $O(n^3)$. Thus, all entire algorithm will have a complexity of $O(n^3)$, independently of the number of edges, and is comparable to the efficiency of the Dijkstra algorithm applied n times.

Exercise 73. The following table represents the distance between several airports joined by an air route.

	A	B	C	D	E	F	G
A	0	5	3	2	-	-	-
B	5	0	2	-	3	-	1
C	3	2	0	7	7	-	-
D	2	-	7	0	2	6	-
E	-	3	7	2	0	1	1
F	-	-	-	6	1	0	-
G	-	1	-	-	1	-	0

(a) What is the minimum distance between airport A and the rest of the airports?

(b) What is the minimum number of aeroplane transfers that it will be necessary to go from airport A to the rest of the airports?

Solution:

(a) Dijkstra's algorithm must be applied on the graph obtained from the table of distances:

A	B	C	D	E	F	G
(0,A)	(∞,A)	(∞,A)	(∞,A)	(∞,A)	(∞,A)	(∞,A)
(0,A)*	(5,A)	(3,A)	(2,A)	(∞,A)	(∞,A)	(∞,A)
(0,A)	(5,A)	(3,A)	(2,A)*	(4,D)	(8,D)	(∞,A)
(0,A)	(5,A)	(3,A)*	(2,A)	(4,D)	(8,D)	(∞,A)
(0,A)	(5,A)	(3,A)	(2,A)	(4,D)*	(5,E)	(5,E)
(0,A)	(5,A)*	(3,A)	(2,A)	(4,D)	(5,E)	(5,E)
(0,A)	(5,A)	(3,A)	(2,A)	(4,D)	(5,E)*	(5,E)
(0,A)	(5,A)	(3,A)	(2,A)	(4,D)	(5,E)	(5,E)*

The last row of the table gives the minimum distances between the airport A and the remaining airports.

(b) Consider the same graph, but now without weights, that is, we are only interested in knowing the number of edges that join the airport A with the remaining airports. In this case, we can apply the BFS algorithm to calculate distances in the non-weighted case:

Q	Added vertex	Deleted vertex	*dist*
A		A -	[0,∞,∞,∞,∞,∞,∞]
AB		B -	[0,1,∞,∞,∞,∞,∞]
ABC	C	-	[0,1,1,∞,∞,∞,∞]
ABCD	D	-	[0,1,1,1,∞,∞,∞]
BCD	-	A	[0,1,1,1,∞,∞,∞]
BCDE	E	-	[0,1,1,1,2,∞,∞]
BCDEG	G	-	[0,1,1,1,2,2,∞]
CDEG	-	B	[0,1,1,1,2,2,∞]
DEG	-	C	[0,1,1,1,2,2,∞]
DEGF	F	-	[0,1,1,1,2,2,2]
EFG	-	D	[0,1,1,1,2,2,2]
FG	-	E	[0,1,1,1,2,2,2]
G	-	F	[0,1,1,1,2,2,2]
∅	-	-	[0,1,1,1,2,2,2]

The list [0,1,1,1,2,2,2] gives the number of transfers that will be necessary to connect airport A with the remaining airports. □

Exercise 74. The following table represents the time needed to connect directly several nodes of a network. The symbol "-" means that the two nodes are not accessible directly. Using the Floyd algorithm, find the minimum time necessary to connect each pair of nodes of the network

	1	2	3	4	5
1	0	3	8	-	4
2	-	0	-	1	7
3	-	4	0	-	-
4	2	-	5	0	-
5	-	-	-	6	0

Solution: The initial table for applying the Floyd algorithm is:

$$d^0 = \begin{pmatrix} 0 & 3 & 8 & \infty & 4 \\ \infty & 0 & \infty & 1 & 7 \\ \infty & 4 & 0 & \infty & \infty \\ 2 & \infty & 5 & 0 & \infty \\ \infty & \infty & \infty & 6 & 0 \end{pmatrix}$$

After applying the algorithm we obtain the table

$$d^5 = \begin{pmatrix} 0 & 3 & 8 & 4 & 4 \\ 3 & 0 & 6 & 1 & 7 \\ 7 & 4 & 0 & 5 & 11 \\ 2 & 5 & 5 & 0 & 6 \\ 8 & 11 & 11 & 6 & 0 \end{pmatrix},$$

where the element in position (i, j) represents the minimum time needed to connect node i with j. □

Exercise 75. The following table represents the distance between the nodes of a network.

	A	B	C	D	E	F	G	H
A	0	3	-	-	-	9	-	4
B	3	0	5	-	10	-	-	-
C	-	5	0	2	-	-	-	18
D	-	-	2	0	7	-	16	-
E	-	10	-	7	0	5	-	-
F	9	-	-	-	5	0	2	-
G	-	-	-	16	-	2	0	5
H	4	-	18	-	-	-	5	0

(a) Apply an algorithm to calculate the distance from C to the other nodes.

(b) Determine a shortest path to go from C to each one of the nodes of the network.

Solution:

(a) On applying the Dijkstra algorithm from node C we obtain the following table:

A	B	C	D	E	F	G	H
(∞,C)	(∞,C)	(0,C)	(∞,C)	(∞,C)	(∞,C)	(∞,C)	(∞,C)
(∞,C)	(5,C)	(0,C)*	(2,C)	(∞,C)	(∞,C)	(∞,C)	(18,C)
(∞,C)	(5,C)	(0,C)	(2,C)*	(9,D)	(∞,C)	(18,D)	(18,C)
(8,B)	(5,C)*	(0,C)	(2,C)	(9,D)	(∞,C)	(18,D)	(18,C)
(8,B)*	(5,C)	(0,C)	(2,C)	(9,D)	(17,A)	(18,D)	(12,A)
(8,B)	(5,C)	(0,C)	(2,C)	(9,D)*	(14,E)	(18,D)	(12,A)
(8,B)	(5,C)	(0,C)	(2,C)	(9,D)	(14,E)	(17,H)	(12,A)*
(8,B)	(5,C)	(0,C)	(2,C)	(9,D)	(14,E)*	(16,F)	(12,A)
(8,B)	(5,C)	(0,C)	(2,C)	(9,D)	(14,E)	(16,F)*	(12,A)

The distance from node C to the other nodes appears in the last row of the table.

(b) The paths are $C-B$, $C-D$, $C-B-A$, $C-B-A-H$, $C-D-E$, $C-D-E-F$ and $C-D-E-F-G$. □

Exercise 76. Consider a weighted network formed by the nodes of the set $\{0,1,2,3,4,5,6\}$. If we apply the Dijkstra algorithm from node 0, then we obtain the following table:

0	1	2	3	4	5	6
(0,0)	(∞,0)	(∞,0)	(∞,0)	(∞,0)	(∞,0)	(∞,0)
(0,0)*	(8,0)	(12,0)	(9,0)	(∞,0)	(∞,0)	(∞,0)
(0,0)	(8,0)*	(12,0)	(9,0)	(∞,0)	(17,1)	(∞,0)
(0,0)	(8,0)	(11,3)	(9,0)*	(21,3)	(16,3)	(∞,0)
(0,0)	(8,0)	(11,3)*	(9,0)	(21,3)	(16,3)	(19,2)
(0,0)	(8,0)	(11,3)	(9,0)	(21,3)	(16,3)*	(19,2)
(0,0)	(8,0)	(11,3)	(9,0)	(21,3)	(16,3)	(19,2)*
(0,0)	(8,0)	(11,3)	(9,0)	(21,3)*	(16,3)	(19,2)

(a) Determine the weight of the edges $\{3,4\}$, $\{3,5\}$ and $\{2,6\}$

(b) Determine three different paths to go from vertex 0 to vertex 4 and find the length of each one of them.

Solution:

(a) The weight of the edges $\{3,4\}$, $\{3,5\}$ and $\{2,6\}$ are 12, 7 and 8, respectively.

(b) The paths are: $0-3-4$, of length 21, $0-1-5-3-4$, of length 36, and $0-2-3-4$, and of length 26. □

Exercise 77. This table registers the operation of the Breadth First Search algorithm (BFS) for a connected graph G, with starting vertex $v = 1$.

Q	Added vertex	Deleted vertex	R
1	1	-	[1]
14	4	-	[1,4]
4	-	1	[1,4]
42	2	-	[1,4,2]
425	5	-	[1,4,2,5]
25	-	4	[1,4,2,5]
253	3	-	[1,4,2,5,3]
53	-	2	[1,4,2,5,3]
3	-	5	[1,4,2,5,3]
∅	-	3	[1,4,2,5,3]

Determine a spanning tree of G.

Solution: The edges of the spanning tree of G are $\{1,4\}$, $\{4,5\}$, $\{4,2\}$ and $\{2,3\}$. □

Exercise 78. To resolve the shortest path problem between two vertices of a network, modeled by a weighted graph G of order 7, we have applied the Dijkstra algorithm beginning with vertex a. We have obtained the following table as the result.

a	b	c	d	e	f	g
$(0,a)$	(∞,a)	(∞,a)	(∞,a)	(∞,a)	(∞,a)	(∞,a)
$(0,a)^*$	$(4,a)$	$(3,a)$	(∞,a)	(∞,a)	(∞,a)	(∞,a)
$(0,a)$	$(4,a)$	$(3,a)^*$	$(9,c)$	$(8,c)$	(∞,a)	(∞,a)
$(0,a)$	$(4,a)^*$	$(3,a)$	$(9,c)$	$(5,b)$	$(10,b)$	(∞,a)
$(0,a)$	$(4,a)$	$(3,a)$	$(9,c)$	$(5,b)^*$	$(10,b)$	$(15,e)$
$(0,a)$	$(4,a)$	$(3,a)$	$(9,c)^*$	$(5,b)$	$(10,b)$	$(15,e)$
$(0,a)$	$(4,a)$	$(3,a)$	$(9,c)$	$(5,b)$	$(10,b)^*$	$(12,f)$
$(0,a)$	$(4,a)$	$(3,a)$	$(9,c)$	$(5,b)$	$(10,b)$	$(12,f)^*$

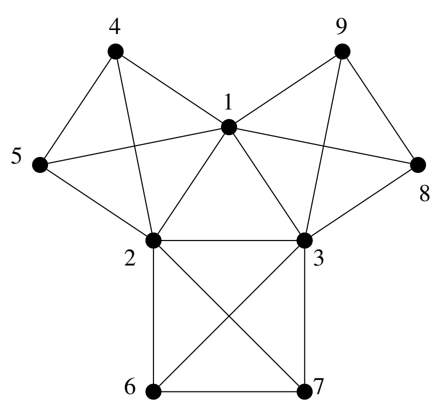
Knowing that the edge of the graph K_2 is denoted by $\{x,y\}$ and that it has weight equal to 5, determine the minimum distance that a robot could travel if it is moving from node (a,x) to the node (g,y) through the weighted network $G\Box K_2$. Determine, in addition, a cycle of minimum length among all the cycles of $G\Box K_2$ that contain vertices (a,x) and (g,y).

Solution: According to the table, in graph G a shortest path from a to g is $a-b-f-g$. Therefore, in $G\Box K_2$ a minimum cost route from (a,x) to (g,y) is $(a,x)-(b,x)-(f,x)-(g,x)-(g,y)$. Another route is $(a,x)-(a,y)-(b,y)-(f,y)-(g,y)$. As both routes have minimum length and only share the initial and final vertices, the cycle we are looking for is:

$$(a,x)-(b,x)-(f,x)-(g,x)-(g,y)-(f,y)-(b,y)-(a,y)-(a,x)$$

The distance from (a,x) to (g,y) is equal to $d_G(a,g)+d_{K_2}(x,y)=12+5=17$. □

Exercise 79. Consider graph G shown in the figure.



(a) Apply the BFS algorithm to determine a spanning tree for G. In the table of the algorithm only consider 3 columns: one for the visited vertices that reflects the data structure Q, one for the added edges, and another with the list of included edges, that at the end will contain the edges of the spanning tree.

(b) Find the minimum number of edges that have to be deleted from $(G\Box P_3)^c$ to obtain a connected subgraph which contains a single cycle.

(c) Determine the diameter of $H=(G\Box P_5)\odot(K_{100,23}+P_{99})$.

(d) How many edges have to be deleted from $G \boxtimes K_5$ to obtain a subgraph of a size equal to $5m(G)$?

(e) Determine if the following statement is true or false. There is a graph whose degree sequence is 6,3,3,3,3,3,3,3,3, but is not a subgraph of G.

(f) Determine the maximum value of n such that the line graph of G has a subgraph isomorph to K_n.

(g) Which edges would be deleted from G to obtain a bipartite graph of the same order as G?

Solution:

(a) The following table shows the steps of the BFS algorithm for determining the spanning tree of G.

Q	Added edges	$A(T)$
1	-	$\varnothing$
12	$\{1,2\}$	$\{\{1,2\}\}$
123	$\{1,3\}$	$\{\{1,2\},\{1,3\}\}$
1234	$\{1,4\}$	$\{\{1,2\},\{1,3\},\{1,4\}\}$
12345	$\{1,5\}$	$\{\{1,2\},\{1,3\},\{1,4\},\{1,5\}\}$
123458	$\{1,8\}$	$\{\{1,2\},\{1,3\},\{1,4\},\{1,5\},\{1,8\}\}$
1234589	$\{1,9\}$	$\{\{1,2\},\{1,3\},\{1,4\},\{1,5\},\{1,8\},\{1,9\}\}$
234589	-	$\{\{1,2\},\{1,3\},\{1,4\},\{1,5\},\{1,8\},\{1,9\}\}$
2345896	$\{2,6\}$	$\{\{1,2\},\{1,3\},\{1,4\},\{1,5\},\{1,8\},\{1,9\},\{2,6\}\}$
23458967	$\{2,7\}$	$\{\{1,2\},\{1,3\},\{1,4\},\{1,5\},\{1,8\},\{1,9\},\{2,6\},\{2,7\}\}$
3458967	-	$\{\{1,2\},\{1,3\},\{1,4\},\{1,5\},\{1,8\},\{1,9\},\{2,6\},\{2,7\}\}$
458967	-	$\{\{1,2\},\{1,3\},\{1,4\},\{1,5\},\{1,8\},\{1,9\},\{2,6\},\{2,7\}\}$
58967	-	$\{\{1,2\},\{1,3\},\{1,4\},\{1,5\},\{1,8\},\{1,9\},\{2,6\},\{2,7\}\}$
8967	-	$\{\{1,2\},\{1,3\},\{1,4\},\{1,5\},\{1,8\},\{1,9\},\{2,6\},\{2,7\}\}$
967	-	$\{\{1,2\},\{1,3\},\{1,4\},\{1,5\},\{1,8\},\{1,9\},\{2,6\},\{2,7\}\}$
67	-	$\{\{1,2\},\{1,3\},\{1,4\},\{1,5\},\{1,8\},\{1,9\},\{2,6\},\{2,7\}\}$
7	-	$\{\{1,2\},\{1,3\},\{1,4\},\{1,5\},\{1,8\},\{1,9\},\{2,6\},\{2,7\}\}$
$\varnothing$	-	$\{\{1,2\},\{1,3\},\{1,4\},\{1,5\},\{1,8\},\{1,9\},\{2,6\},\{2,7\}\}$

(b) The size of $G\square P_3$ is $m(G\square P_3) = 3m(G) + 2n(G) = 3 \cdot 18 + 2 \cdot 9 = 72$; thus, the size of the complement is $\binom{27}{2} - 72 = 279$. The entire spanning tree of $(G\square P_3)^c$ has 26 edges, so we can delete $279 - 27 = 252$ edges.

(c) The diameter of H is $D(H) = D(G\square P_5) + 2 = D(G) + 6 = 8$.

(d) We have to delete $2m(G)m(K_5) + n(G)m(K_5) = 2 \cdot 18 \cdot 10 + 9 \cdot 10 = 450$ edges.

(e) True. If we apply the Havel-Hakimi algorithm we can check that the sequence is graphic:

6,3,3,3,3,3,3,3,3

2,2,2,2,2,2,3,3

3,3,2,2,2,2,2,2

2,1,1,2,2,2,2

2,2,2,2,2,1,1

1,1,2,2,1,1

2,2,1,1,1,1

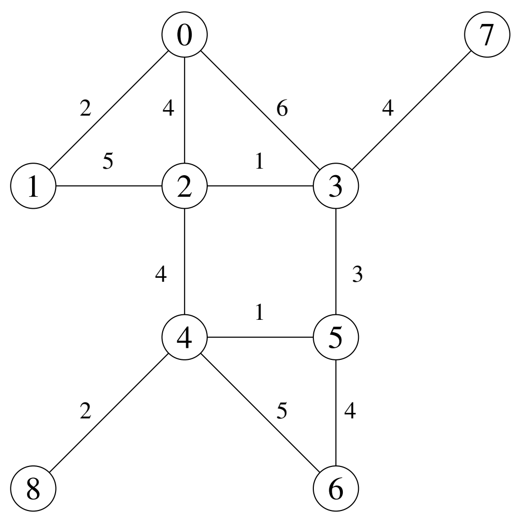
1,0,1,1,1

This last sequence is graphic, and corresponds to $K_2 \cup K_2 \cup K_1$. To see that there are no subgraphs of G with the degree sequence listed above, take a vertex of degree 6, which by symmetry could be the 1. The other vertices of the subgraph have to have degree 3, but this is impossible as with vertices 2 and 3 we can only delete one edge, $2-3$, if we erase another edge incident to 2 or to 3; then we obtain vertices of degree 2, which is a contradiction.

(f) The value of n is 6, the maximum degree of G. Note that the edges incidents to a vertex v of G determine a subgraph of $L(G)$ which is isomorph to $K_{\delta(v)}$.

(g) We have to delete all the cycles of odd length. For example, we can delete edges $2-3$, $6-7$, $1-5$, $2-4$, $1-8$ and $3-9$. □

Exercise 80. Consider a transport network defined by the following graph:



Supposing that the weight of each edge represents the distance between points of the network, apply an algorithm to determine the minimum distance between point 0 and the other points of the network. Determine, in addition, the shortest paths to travel.

Solution: The following table shows the steps of the Dijkstra algorithm:

0	1	2	3	4	5	6	7	8
(0,0)	(∞,0)	(∞,0)	(∞,0)	(∞,0)	(∞,0)	(∞,0)	(∞,0)	(∞,0)
(0,0)*	(2,0)	(4,0)	(6,0)	(∞,0)	(∞,0)	(∞,0)	(∞,0)	(∞,0)
(0,0)	(2,0)*	(4,0)	(6,0)	(∞,0)	(∞,0)	(∞,0)	(∞,0)	(∞,0)
(0,0)	(2,0)	(4,0)*	(5,2)	(8,2)	(∞,0)	(∞,0)	(∞,0)	(∞,0)
(0,0)	(2,0)	(4,0)	(5,2)*	(8,2)	(8,3)	(∞,0)	(9,3)	(∞,0)
(0,0)	(2,0)	(4,0)	(5,2)	(8,2)*	(8,3)	(13,4)	(9,3)	(10,4)
(0,0)	(2,0)	(4,0)	(5,2)	(8,2)	(8,3)*	(12,5)	(9,3)	(10,4)
(0,0)	(2,0)	(4,0)	(5,2)	(8,2)	(8,3)	(12,5)	(9,3)*	(10,4)
(0,0)	(2,0)	(4,0)	(5,2)	(8,2)	(8,3)	(12,5)	(9,3)	(10,4)*
(0,0)	(2,0)	(4,0)	(5,2)	(8,2)	(8,3)	(12,5)*	(9,3)	(10,4)

The distance from point 0 to the other points is in the last row of the table. The shortest paths are: $0-1$, $0-2$, $0-2-3$, $0-2-3-7$, $0-2-4$, $0-2-4-8$, $0-2-3-5$ and $0-2-3-5-6$ □

Exercise 81. Determine the length of the longest route among all the routes of minimum length of $G\square K_2$. It is known that in an intermediate step of the Floyd algorithm (applied to graph G) we have obtained the matrix

$$d^5 = \begin{pmatrix} 0 & 4 & 3 & 9 & 5 & 10 & 15 \\ 4 & 0 & 4 & 5 & 1 & 6 & 11 \\ 3 & 4 & 0 & 6 & 5 & 10 & 12 \\ 9 & 5 & 6 & 0 & 4 & 4 & 6 \\ 5 & 1 & 5 & 4 & 0 & 6 & 10 \\ 10 & 6 & 10 & 4 & 6 & 0 & 2 \\ 15 & 11 & 12 & 6 & 10 & 2 & 0 \end{pmatrix}.$$

Consider that the edge of the weighted graph K_2 is denoted by $\{x,y\}$ and has equal weight to 5.

Solution: It involves determining the diameter of the weighted graph $G\square K_2$. Continuing the Floyd algorithm applied to the graph G obtains the following matrices:

$$d^6 = \begin{pmatrix} 0 & 4 & 3 & 9 & 5 & 10 & 12 \\ 4 & 0 & 4 & 5 & 1 & 6 & 8 \\ 3 & 4 & 0 & 6 & 5 & 10 & 12 \\ 9 & 5 & 6 & 0 & 4 & 4 & 6 \\ 5 & 1 & 5 & 4 & 0 & 6 & 8 \\ 10 & 6 & 10 & 4 & 6 & 0 & 2 \\ 12 & 8 & 12 & 6 & 8 & 2 & 0 \end{pmatrix} \quad d^7 = \begin{pmatrix} 0 & 4 & 3 & 9 & 5 & 10 & 12 \\ 4 & 0 & 4 & 5 & 1 & 6 & 8 \\ 3 & 4 & 0 & 6 & 5 & 10 & 12 \\ 9 & 5 & 6 & 0 & 4 & 4 & 6 \\ 5 & 1 & 5 & 4 & 0 & 6 & 8 \\ 10 & 6 & 10 & 4 & 6 & 0 & 2 \\ 12 & 8 & 12 & 6 & 8 & 2 & 0 \end{pmatrix}$$

From the last matrix we deduce that the diameter of G is 12. Therefore, the diameter of $G\square K_2$ $D(G\square K_2) = D(G) + 5 = 17$. □

Exercise 82. The following matrix is the adjacency matrix of a weighted graph G whose vertices are labelled in alphabetical order starting from vertex A, which corresponds to row 1 and column 1.

$$\begin{pmatrix} 0 & 3 & 7 & 12 & 0 & 0 \\ 3 & 0 & 2 & 0 & 3 & 0 \\ 7 & 2 & 0 & 7 & 0 & 4 \\ 12 & 0 & 7 & 0 & 0 & 2 \\ 0 & 3 & 0 & 0 & 0 & 3 \\ 0 & 0 & 4 & 2 & 3 & 0 \end{pmatrix}.$$

(a) Apply the Dijkstra algorithm starting from vertex A and determine a path of minimum length to go from vertex A to vertex D.

(b) Apply the Floyd algorithm to determine the distance between any pair of nodes of the network.

(c) Determine the eccentricity of each vertex and determine, in addition, the radius and diameter of the network.

(d) Determine the diameter of the weighted graph $G\square H$, where H is a weighted path of order 4 whose edges have weight 3.

Solution:

(a) The following table shows the steps of the Dijkstra algorithm:

A	B	C	D	E	F
(0,A)	(∞,A)	(∞,A)	(∞,A)	(∞,A)	(∞,A)
(0,A)*	(3,A)	(7,A)	(12,A)	(∞,A)	(∞, A)
(0,A)	(3,A)*	(5,B)	(12,A)	(6,B)	(∞,A)
(0,A)	(3,A)	(5,B)*	(12,A)	(6,B)	(9,C)
(0,A)	(3,A)	(5,B)	(12,A)	(6,B)*	(9,C)
(0,A)	(3,A)	(5,B)	(11,F)	(6,B)	(9,C)*
(0,A)	(3,A)	(5,B)	(11,F)*	(6,B)	(9,C)

The distance from A to D is 11 and the shortest path is $A-B-C-F-D$ the weights of the stretches are 3, 2, 4 and 2, respectively.

(b) Applying the Floyd algorithm we obtain the matrix

$$d^6 = \begin{pmatrix} 0 & 3 & 5 & 11 & 6 & 9 \\ 3 & 0 & 2 & 8 & 3 & 6 \\ 5 & 2 & 0 & 6 & 5 & 4 \\ 11 & 8 & 6 & 0 & 5 & 2 \\ 6 & 3 & 5 & 5 & 0 & 3 \\ 9 & 6 & 4 & 2 & 3 & 0 \end{pmatrix}.$$

(c) The eccentricities are: $\varepsilon(A) = 11$, $\varepsilon(B) = 8$, $\varepsilon(C) = 6$, $\varepsilon(D) = 11$, $\varepsilon(E) = 6$, $\varepsilon(F) = 9$. The radius of the network is $r(G) = 6$ and the diameter is $D(G) = 11$.

(d) $D(G\Box H) = D(G) + D(H) = 11 + 9 = 20.$ $\Box$

Exercise 83. Suppose that in a communication network the weights (w) of the edges reflect the quality of the information transmitted or the trust between nodes. For instance:

(I) The probability that a bit transmitted by node i is erroneously received by node j is a simple way to quantify the quality of transmission between node i and node j. This probability can be viewed as the weight $w(i,j)$ of edge (i,j).

(II) The level of trust between node i and node j in a social network can be quantified as a weight $w(i,j) \in [0,1]$ proportional to how much node i trusts node j. Extreme cases are no trust ($w(i,j) = 0$) and maximum trust ($w(i,j) = 1$).

In general, we are interested in the cases where the weight of a path can be computed as the product of the weights of the edges belonging to it. The weight of the path is called its *reliability*. More formally, let $G = (V,E,w)$ be a weighted digraph with vertex set V and edge set E, where $w : E \mapsto (0,1]$ is the weight function of G. For a path $P : u = v_i, v_{i+1}, \ldots, v_k = v$ in G we define the *reliability of P* as

$$w(P) = \prod_{l=i}^{k-1} w(v_l, v_{l+1}).$$

Let us assume that several routes exist from one node to another. The maximum reliability between two nodes (in terms of transmission quality, trust, etc.) is reached using the path with maximum reliability among those connecting both nodes.

For two vertices $u,v \in V$, we denote by $P_{\vec{uv}}$ the set of all directed paths from u to v. We denote by $F_{\vec{uv}}$ the weight of the most reliable path from u to v:

$$F_{\vec{uv}} = \max_{P \in P_{\vec{uv}}} \{w(P)\}. \tag{4.1}$$

(a) Find a method to compute the weights of the most reliable path from u to v for every pair of different nodes of $u,v \in V$.

(b) Let $G = (V,E,w)$ be the weighted digraph whose adjacency matrix is given by

$$\begin{pmatrix}
0.0 & 0.5 & 0.3 & 0.0 & 0.2 & 0.0 & 0.8 & 0.8 \\
0.2 & 0.0 & 0.2 & 0.0 & 0.2 & 0.2 & 0.2 & 0.2 \\
0.0 & 0.0 & 0.0 & 0.0 & 0.5 & 0.9 & 0.0 & 0.0 \\
0.1 & 0.0 & 0.4 & 0.0 & 0.4 & 0.2 & 0.1 & 0.2 \\
0.0 & 0.5 & 0.0 & 0.0 & 0.0 & 0.6 & 0.0 & 0.6 \\
0.3 & 0.2 & 0.0 & 0.0 & 0.0 & 0.0 & 0.3 & 0.1 \\
0.6 & 0.6 & 0.4 & 0.6 & 0.6 & 0.6 & 0.0 & 0.6 \\
0.0 & 0.7 & 0.6 & 0.0 & 0.7 & 0.0 & 0.6 & 0.0
\end{pmatrix}.$$

Compute the values of $F_{\vec{uv}}$ for every pair of different nodes $u,v \in V$.

Solution:

(a) For a path $P: u = v_i, v_{i+1}, \ldots, v_k = v$ in G we have

$$w(P) = \prod_{l=i}^{k-1} w(v_l, v_{l+1}) = \exp\left(\sum_{l=i}^{k-1} \ln\left(w(v_l, v_{l+1})\right)\right) = \exp\left(-\sum_{l=i}^{k-1} \ln\left(\frac{1}{w(v_l, v_{l+1})}\right)\right).$$

Hence, the problem of finding $F_{\vec{uv}}$ can be solved as follows. From $G = (V, E, w)$ we define the weighted digraph $G' = (V, E, w')$ where the weight function $w' : E \to \mathbb{R}_+$ is defined by

$$w'(i, j) = \ln\left(\frac{1}{w(v_i, v_{i+1})}\right).$$

In such a case we can take

$$w'(P) = \sum_{l=i}^{k-1} w'(v_l, v_{l+1}),$$

and, as a result,

$$w(P) = \exp(-w'(P)).$$

Notice that $w(P)$ is maximum whenever $w'(P)$ is minimum. Therefore, the reliability $F_{\vec{uv}}$ in $G = (V, E, w)$ is

$$F_{\vec{uv}} = \exp(-F'_{\vec{uv}}),$$

where

$$F'_{\vec{uv}} = \min_{P \in P_{\vec{uv}}} \{w'(P)\}.$$

Obviously, $F'_{\vec{uv}}$ can be calculated by Dijkstra's algorithm on the weighted digraph $G' = (V, E, w')$.

(b) The values of $F_{\vec{uv}}$ are given in the following table (row u).

	1	2	3	4	5	6	7	8
1	0	0.5600	0.4800	0.4800	0.5600	0.4800	0.8000	0.8000
2	0.2000	0	0.2000	0.1200	0.2000	0.2000	0.2000	0.2000
3	0.2700	0.2500	0	0.1620	0.5000	0.9000	0.2700	0.3000
4	0.1080	0.2000	0.4000	0	0.4000	0.3600	0.1440	0.2400
5	0.2160	0.5000	0.3600	0.2160	0	0.6000	0.3600	0.6000
6	0.3000	0.2000	0.1440	0.1800	0.1800	0	0.3000	0.2400
7	0.6000	0.6000	0.4000	0.6000	0.6000	0.6000	0	0.6000
8	0.3600	0.7000	0.6000	0.3600	0.7000	0.5400	0.6000	0

Another way to reach the same result: We can modify Dijkstra's algorithm, or Floyd's algorithm, to the problem statement for $G = (V, E, w)$. For example, in the case of Floyd's algorithm, we change $d_{ij}^k = \min(d_{ij}^{k-1}, d_{ik}^{k-1} + d_{kj}^{k-1})$ to $d_{ij}^k = \max(d_{ij}^{k-1}, d_{ik}^{k-1} \cdot d_{kj}^{k-1})$ whenever $i \neq j$. □

4.4 The metric dimension of metric spaces and graphs

The notion of metric dimension of a general metric space was introduced for the first time in 1953 by Blumenthal [1] in his book *Theory and Applications of Distance Geometry*. This theory attracted little attention until, more than twenty years later, it was applied to the particular case of graphs. Since then, it has been frequently applied and investigated in graph theory and many other disciplines.

Let $M = (X,d)$ be a metric space. If X is an infinite set, we put $|X| = +\infty$. In fact, it is possible to develop the theory with $|X|$ any cardinal number, but we shall not do this. A subset $A \subseteq X$ is said to *resolve* M if for any pair of different points $x,y \in X$, there exists a point $a \in A$ such that $d(x,a) \neq d(y,a)$. Informally, if an object in M knows its distance from each point of A, then it knows exactly where it is located in M. The class $\mathcal{R}(M)$ of subsets of X that resolve M is non-empty since X resolves M. The *metric dimension* $\dim_m(M)$ of $M = (X,d)$ is defined as

$$\dim_m(M) = \inf\{|A| : A \in \mathcal{R}(M).\}$$

The sets in $\mathcal{R}(M)$ are called the *metric generators* of M, and A is a *metric basis* of M if $A \in \mathcal{R}(M)$ and $|A| = \dim_m(M)$.

This terminology comes from the fact that a metric generator of a metric space $M = (X,d)$ induces a *global co-ordinate system* on M. For instance, if $(x_1,\dots,x_r)$ is an ordered metric generator of M, then the map $\psi : X \to \mathbb{R}^r$ given by

$$\psi(x) = \Big(d(x,x_1),\dots,d(x,x_r)\Big) \tag{4.2}$$

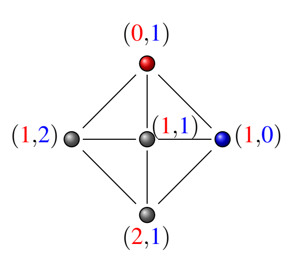
is injective, which implies that the map ψ is a bijection from X to a subset of $\mathbb{R}^r$, and the metric space inherits its co-ordinates from this subset.

Remark 44. *For any Euclidean affine space $\mathcal{A}$ of dimension n,*

$$\dim_m(\mathcal{A}) = n+1.$$

Proof. Exercise. □

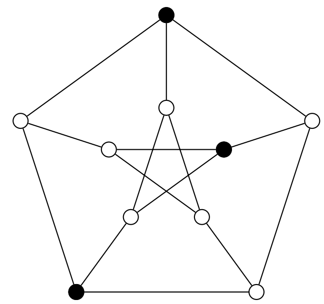
Example 67. $\dim_m(K_1 + C_4) = 2$. The following figure shows a metric basis of $K_1 + C_4$ and the corresponding co-ordinate system.



Example 68. $\dim_m(C_n) = 2$ and $\dim_m(Q_3) = 3$.

Exercise 84. Find a metric basis for the Petersen graph.

Solution: The set of black-coloured vertices forms a metric basis.



$\square$

Exercise 85. Let G be a connected graph of order $n \geq 2$. Show that $\dim_m(G) = 1$ if and only if $G \cong P_n$.

Solution: Since every vertex of degree one in P_n is a metric generator, $\dim_m(P_n) = 1$. Now, if $\dim_m(G) = 1$, the for any metric basis $\{w\}$ of G, we have that $d(w,x) \neq d(w,y)$ for every pair of different vertices $x, y \in V(G)$. Hence, the set $\{d(w,x) : x \in V(G) \setminus \{w\}\}$ has cardinality $n-1$, which implies that G has diameter $D = n-1$, and so $G \cong P_n$. $\square$

Exercise 86. Let G be a connected graph of order $n \geq 2$ and diameter D. Show that

$$\dim_m(G) \leq n - D.$$

Solution: Let $x_0, x_1, \ldots, x_D$ be a diametral path of G. Since $d(x_0, x_i) = i$ for every $i \in \{1, \ldots, D\}$, the set $X = V(G) \setminus \{x_1, \ldots, x_D\}$ is a metric generator of G, and so $\dim_m(G) \leq |X| = n - D$. $\square$

Exercise 87. Let G be a connected graph of order $n \geq 2$. Show that $\dim_m(G) = n-1$ if and only if $G \cong K_n$.

Solution: It is readily seen that $\dim_m(K_n) = n-1$. Now, if G has diameter $D \geq 2$, then Exercise 86 leads to $\dim_m(G) \leq n - D \leq n-2$. $\square$

Exercise 88. Find the value of $\dim_m(P_n \Box P_{n'})$ for every pair of integers $n \geq 2$ and $n' \geq 2$.

Solution: Let $V(P_n) = \{0, 1, \ldots, n-1\}$ and $V(P_{n'}) = \{0, 1, \ldots, n'-1\}$, where consecutive vertices are adjacent. Since $P_n \Box P_{n'}$ is not a path, $\dim_m(P_n \Box P_{n'}) \geq 2$. We claim that $\dim_m(P_n \Box P_{n'}) = 2$. To see this, suppose that $B = \{(0,0), (n-1,0)\}$ is not a metric generator. In such a case, for every pair of different vertices (x,y) and (x',y')

$$x + y = d_{P_n \Box P_{n'}}((0,0),(x,y)) = d_{P_n \Box P_{n'}}((0,0),(x',y')) = x' + y'$$

and

$$n - 1 - x + y = d_{P_n \Box P_{n'}}((n-1,0),(x,y)) = d_{P_n \Box P_{n'}}((n-1,0),(x',y')) = n - 1 - x' + y'.$$

Hence, $x = x'$ and $y = y'$, which is a contradiction. Thus, B is a metric generator, and so $\dim_m(P_n \Box P_{n'}) \leq |B| = 2$. Therefore, $\dim_m(P_n \Box P_{n'}) = 2$. $\square$

Exercise 89. Show that for any graph G of order n and diameter D,

$$D^{\dim_m(G)} + \dim_m(G) \geq n.$$

Solution: For any metric basis $B = \{x_1, \dots, x_r\}$, the function

$$\begin{aligned} \psi : V \setminus B &\longrightarrow \{1, \dots, D\}^r \\ \psi(x) &= \Big(d(x,x_1), \dots, d(x,x_r)\Big) \end{aligned}$$

is injective. Hence, $n - r = |V \setminus B| \leq D^r$. $\square$.

Exercise 90. Show that for any graph G of order n and diameter $D \geq 2$,

$$\dim_m(G) \geq \frac{\ln\left(\frac{nD}{D+1}\right)}{\ln(D)}.$$

Solution: From the previous exercise we know that $D^{\dim_m(G)} + \dim_m(G) \geq n$. Now, since $D \geq 2$ and $\dim_m(G) \geq 1$, we have that $D^{\dim_m(G)} \geq D \cdot \dim_m(G)$. Hence,

$$\begin{aligned} D^{\dim_m(G)} + \dim_m(G) &\geq n \\ D^{\dim_m(G)+1} + D \cdot \dim_m(G) &\geq n \cdot D \\ D^{\dim_m(G)+1} + D^{\dim_m(G)} &\geq n \cdot D \\ D^{\dim_m(G)}(D+1) &\geq n \cdot D \\ D^{\dim_m(G)} &\geq \frac{n \cdot D}{D+1} \\ \dim_m(G) &\geq \log_D\left(\frac{n \cdot D}{D+1}\right) \\ \dim_m(G) &\geq \frac{\ln\left(\frac{nD}{D+1}\right)}{\ln(D)}. \end{aligned}$$

$\square$

Exercise 91. Let G and H be two connected graphs of order $n \geq 2$ and $n' \geq 2$, respectively. Prove the following statements.

(a) $\dim_m(G \odot H) \geq n \cdot \dim_m(H)$.

(b) If $D(H) \leq 2$, then $\dim_m(G \odot H) = n \cdot \dim_m(H)$.

Solution: Let M be a metric basis of $G \odot H$. Let $H_i = (V_i, E_i)$ be the subgraph of $G \odot H$ corresponding to the i^{th}-copy of H.

Since no vertex outside V_i distinguishes vertices in V_i, we have that $M_i = M \cap V_i$ is a metric generator of H_i for every $v_i \in V(G)$. Thus, by the minimality of $|M|$ we can conclude that

$M \cap V(G) = \varnothing$, as every pair of vertices with at least one vertex in $V(G)$ is distinguished by some vertex in $\bigcup M_i$. Hence,

$$\dim_m(G \odot H) = |M| = \sum_{i=1}^{n} |M_i| \geq \sum_{i=1}^{n} \dim_m(H) = n \cdot \dim_m(H).$$

Therefore, (a) follows.

In order to prove (b), let $S_i \subseteq V_i$ be a resolving set for H_i and let $S = \bigcup_{i=1}^{n} S_i$. We will show that S is a metric generator for $G \odot H$. Let us consider two different vertices x, y of $G \odot H$. We have the following cases.

Case 1: $x, y \in V_i$. Let $z \in S_i$ be a vertex which distinguishes x and y in H_i. Since $D(H_i) \leq 2$, we have that z distinguishes x and y in $G \odot H$.

Case 2: $x \in V_i$ and $y \in V_j$, $i \neq j$. Let $v \in S_i$. Hence we have $d(x,v) \leq 2 < 3 \leq d(y,v)$.

Case 3: $x, y \in V$. Let $x = v_i$. Then for every vertex $v \in S_i$ we have $d(x,v) = 1 < d(y,x) + 1 = d(y,v)$.

Case 4: $x \in V_i$ and $y \in V$. If $x \sim y$, then let $v \in S_j$, for some $j \neq i$. So we have $d(x,v) = 1 + d(y,v) > d(y,v)$. Moreover, if $x \not\sim y = v_j$, for $v \in S_j$ we have $d(x,v) = d(x,y) + d(y,v) > d(y,v)$.

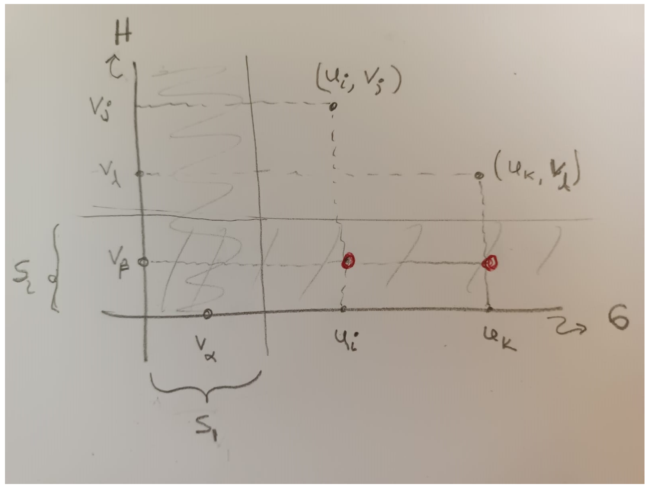
According to the four cases above, S is a metric generator for $G \odot H$ and, as a consequence, $\dim_m(G \odot H) \leq n \cdot \dim_m(H)$. Therefore, by (a) we conclude the proof of (b). □

Exercise 92. Let G and H be two connected graphs of order $n_1 \geq 2$ and n_2, respectively. Prove that

$$\dim_m(G \boxtimes H) \leq n_1 \cdot \dim_m(H) + n_2 \cdot \dim_m(G) - \dim_m(G) \cdot \dim_m(H).$$

Solution: Let $V_1 = \{u_1, u_2, \ldots, u_{n_1}\}$ and $V_2 = \{v_1, v_2, \ldots, v_{n_2}\}$ be the set of vertices of G and H, respectively. Let $S = (V_1 \times S_2) \cup (S_1 \times V_2)$, where S_1 and S_2 are metric basis for G and H, respectively. Let (u_i, v_j) and (u_k, v_l) be two different vertices of $G \boxtimes H$. Let $u_\alpha \in S_1$ such that u_i, u_k are distinguished by u_α and let $v_\beta \in S_2$ such that v_j, v_l are distinguished by v_β.

If $i = k$, then (u_i, v_j) and (u_k, v_l) are distinguished by $(u_i, v_\beta) \in (V_1 \times S_2) \subseteq S$. Analogously, if $j = l$, then (u_i, v_j) and (u_k, v_l) are distinguished by $(u_\alpha, v_j) \in (S_1 \times V_2) \subseteq S$.



If $i \neq k$ and $j \neq l$, then we suppose that neither (u_i, v_β) nor (u_k, v_β) distinguishes the pair $(u_i, v_j), (u_k, v_l)$, *i.e.*,

$$d_{G \boxtimes H}((u_i, v_j), (u_i, v_\beta)) = d_{G \boxtimes H}((u_k, v_l), (u_i, v_\beta)) \tag{4.3}$$

and

$$d_{G \boxtimes H}((u_i, v_j), (u_k, v_\beta)) = d_{G \boxtimes H}((u_k, v_l), (u_k, v_\beta)). \tag{4.4}$$

Hence, $d_H(v_j, v_\beta) = \max\{d_G(u_k, u_i), d_H(v_l, v_\beta)\}$ and since $d_H(v_j, v_\beta) \neq d_H(v_l, v_\beta)$, we obtain that

$$d_H(v_j, v_\beta) = d_G(u_k, u_i). \tag{4.5}$$

Also, by (4.4) we have $d_H(v_l, v_\beta) = \max\{d_G(u_i, u_k), d_H(v_j, v_\beta)\}$ and since $d_H(v_j, v_\beta) \neq d_H(v_l, v_\beta)$, we obtain that

$$d_H(v_l, v_\beta) = d_G(u_i, u_k). \tag{4.6}$$

From (4.5) and (4.6) we have that $d_H(v_j, v_\beta) = d_H(v_l, v_\beta)$, which is a contradiction with the statement that v_j, v_l are distinguished by v_β in H. □

Exercise 93. Prove that the bound given in Exercise 92 is tight.

Solution: Since $K_{n_1} \boxtimes K_{n_2} \cong K_{n_1 \cdot n_2}$ and for any complete graph K_n, $\dim_m(K_n) = n - 1$, we deduce

$$\begin{aligned}
\dim_m(K_{n_1} \boxtimes K_{n_2}) &= n_1 \cdot n_2 - 1 \\
&= n_1(n_2 - 1) + n_2(n_1 - 1) - (n_1 - 1)(n_2 - 1) \\
&= n_1 \cdot \dim_m(K_{n_2}) + n_2 \cdot \dim_m(K_{n_1}) - \dim_m(K_{n_1}) \cdot \dim_m(K_{n_2}).
\end{aligned}$$

Therefore, the above bound is tight. □

Exercise 94. Let G be a connected graph of order n and let $r \geq 2$ be an integer. Find the value of $\dim_m(G \circ K_r)$ under the assumption that $N[x] \neq N[x']$ for every pair of different vertices $x, x' \in V(G)$.

Solution: We proceed to show that $\dim_m(G \circ K_r) = n(r-1)$. To this end, for every vertex $u \in V(G)$, the subgraph of $G \circ K_r$ induced by $V_u = \{u\} \times V(K_r)$ will be denoted by H_u. Obviously, $H_u \cong K_r$.

Let B be a metric basis of $G \circ K_r$. Since for every $u \in V(G)$, no vertex outside V_u is able to distinguish two vertices in V_u, the set $B \cap V_u$ is a metric generator of H_u, and so

$$\dim_m(G \circ K_r) = |B| = \sum_{u \in V(G)} |B \cap V_u| \geq \sum_{u \in V(G)} \dim_m(H_u) = n(r-1).$$

From now on, let X_u be a metric basis of H_u for every $u \in V(G)$. We claim that $X = \bigcup_{u \in V(G)} X_u$ is a metric generator of $G \circ K_r$. To see this, let $(x,y),(x',y') \in V(G \circ K_r) \setminus X$ be two different vertices. Notice that, since $H_u \cong K_r$ and X_u is a metric basis of H_u, we can conclude that $x \neq x'$, as $|X_u| = r-1$. Notice that if $x \not\sim x'$, then every vertex in X_x distinguishes the pair $(x,y),(x',y')$. Assume $x \sim x'$. Since $N[x] \neq N[x']$, without loss of generality, we can assume that there exists $x'' \in N(x') \setminus N(x)$, and so any vertex in $X_{x''}$ distinguishes the pair $(x,y),(x',y')$. Therefore, X is a metric generator of $G \circ K_r$ and, as a result,

$$\dim_m(G \circ K_r) \leq |X| = \sum_{u \in V(G)} |X_u| = \sum_{u \in V(G)} \dim_m(H_u) = n(r-1).$$

In summary, we have shown that $\dim_m(G \circ K_r) = n(r-1)$. □

Definition 42. A set S of vertices in a graph G is an *adjacency generator* for a graph G if for every pair of different vertices $x,y \in V(G) \setminus S$ there exists $s \in S$ such that $|N_G(s) \cap \{x,y\}| = 1$, where $N_G(s)$ denotes the neighbourhood of s in G.

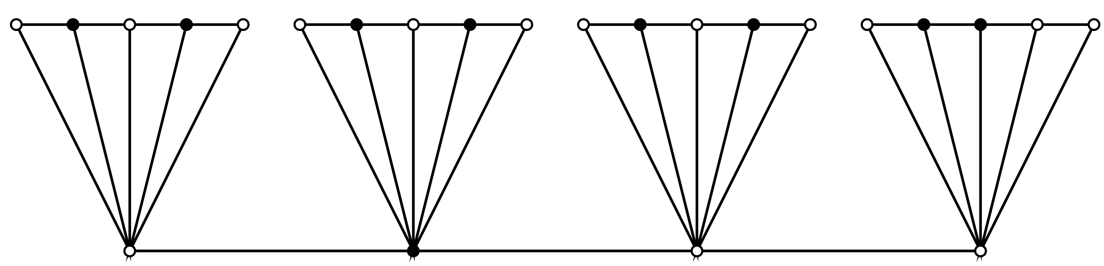
A set of minimum cardinality among all adjacency generators is called an *adjacency basis* for G and its cardinality, the *adjacency dimension* of G, is denoted by $\dim_A(G)$.

Observe that an adjacency generator of a graph $G = (V,E)$ is also a generator in a suitably chosen metric space, namely by considering $(V, d_{G,2})$, with

$$d_{G,2}(x,y) = \min\{d_G(x,y), 2\}.$$

Exercise 95. Find an adjacency basis for $P_4 \odot P_5$.

Solution: The bold type indicates an adjacency basis for $P_4 \odot P_5$.



□

Exercise 96. Show that for any connected graph G of order $n \geq 2$ and any non-trivial graph H,

$$\dim_m(G \odot H) = n \cdot \dim_A(H).$$

Solution: We first need to prove that $\dim_m(G \odot H) \leq n \cdot \dim_A(H)$. For any $i \in \{1, \ldots, n\}$, let S_i be an adjacency basis of $H_i = (V_i, E_i)$, the i^{th}-copy of H. In order to show that $X = \bigcup_{i=1}^{n} S_i$ is a metric generator for $G \odot H$, we differentiate the following four cases for two vertices $x, y \in V(G \odot H) \setminus X$.

Case 1. $x, y \in V_i$. Since S_i is an adjacency basis of H_i, there exists a vertex $u \in S_i$ such that $|N_{H_i}(u) \cap \{x, y\}| = 1$. Hence,

$$d_{G \odot H}(x, u) = d_{\langle v_i \rangle + H_i}(x, u) \neq d_{\langle v_i \rangle + H_i}(y, u) = d_{G \odot H}(y, u).$$

Case 2. $x \in V_i$ and $y \in V$. If $y = v_i$, then for $u \in S_j$, $j \neq i$, we have

$$d_{G \odot H}(x, u) = d_{G \odot H}(x, y) + d_{G \odot H}(y, u) > d_{G \odot H}(y, u).$$

Now, if $y = v_j$, $j \neq i$, then we also take $u \in S_j$ and we proceed as above.

Case 3. $x = v_i$ and $y = v_j$. For $u \in S_j$, we find that

$$d_{G \odot H}(x, u) = d_{G \odot H}(x, y) + d_{G \odot H}(y, u) > d_{G \odot H}(y, u).$$

Case 4. $x \in V_i$ and $y \in V_j$, $j \neq i$. In this case, for $u \in S_i$ we have

$$d_{G \odot H}(x, u) \leq 2 < 3 \leq d_{G \odot H}(u, y).$$

Hence, X is a metric generator for $G \odot H$ and, as a consequence,

$$\dim_m(G \odot H) \leq \sum_{i=1}^{n} |S_i| = n \cdot \dim_A(H).$$

It remains to prove that $\dim_m(G \odot H) \geq n \cdot \dim_A(H)$. To do this, let W be a metric basis for $G \odot H$ and, for any $i \in \{1, \ldots, n\}$, let $W_i = V_i \cap W$. Let us show that W_i is an adjacency generator for H_i. To do this, consider two different vertices $x, y \in V_i - W_i$. Since no vertex $a \in V(G \odot H) - V_i$ distinguishes the pair x, y, there exists some $u \in W_i$ such that $d_{G \odot H}(x, u) \neq d_{G \odot H}(y, u)$. Now, since $d_{G \odot H}(x, u) \in \{1, 2\}$ and $d_{G \odot H}(y, u) \in \{1, 2\}$, we conclude that $|N_{H_i}(u) \cap \{x, y\}| = 1$ and consequently, W_i must be an adjacency generator for H_i. Hence, for any $i \in \{1, \ldots, n\}$, $|W_i| \geq \dim_A(H_i)$. Therefore,

$$\dim_m(G \odot H) = |W| \geq \sum_{i=1}^{n} |W_i| \geq \sum_{i=1}^{n} \dim_A(H_i) = n \cdot \dim_A(H).$$

This completes the proof. $\square$

Obviously, $\dim_m(G) \leq \dim_A(G)$. Now, if G has diameter at most two, then $\dim_m(G) = \dim_A(G)$.

Exercise 97. Find a family of graphs G with diameter $D(G) \geq 3$ and $\dim_m(G) = \dim_A(G)$.

Solution: Let G be a connected graph of order n, diameter $DG) \geq 3$, and let $r \geq 2$ be an integer. As we have shown in Exercise 94, under the assumption that $N[x] \neq N[x']$ for every pair of different vertices $x, x' \in V(G)$, we have $\dim_m(G \circ K_r) = n(r-1) = n \dim_A(K_r)$. It is readily seen that in this case $D(G \circ K_r) \geq 3$ and $\dim_m(G \circ K_r) = \dim_A(G \circ K_r)$. $\square$

Chapter 5

Trees

Trees play a central role in the design and analysis of connected networks, and they are very important to the structural understanding of them. In computer science, for instance, they are an important tool to study compiles, data structure and algorithms. In this chapter, we present some basic results and algorithms related to trees.

5.1 Trees: basic concepts

In accordance with Definition 35, a *tree* is a connected graph without cycles. If we delete the condition of connectivity, we obtain a *forest*, that is, a forest is an acyclic graph.

Theorem 45. *If $T = (V,E)$ is a graph of order n and size m, then the following properties are equivalent:*

(1) *T is a tree, i.e., T is a connected graph and does not have cycles.*

(2) *Between each pair of vertices of T there exists exactly one path.*

(3) *T is connected and $m = n-1$.*

(4) *T is acyclic and $m = n-1$.*

Proof. **(1)** $\Leftrightarrow$ **(2)**. First, assume that T is a tree. Then between each pair of vertices there is a path. Since T does not contain any cycles, this path has to be unique, since if there were two different paths l_1 and l_2 between u and v then the walk $l_1 \cup l_2$ would be closed and, therefore, would contain a cycle. Reciprocally, if between each pair of vertices of T there exists exactly one path, T is connected and does not contain cycles.

(1) $\Leftrightarrow$ **(3)**. First, assume that T is a tree. T is connected by the definition of a tree. We will show by induction that $m = n-1$. For $n = 1$ the result is trivially true. We suppose that the result is true for all trees of order $k < n$ and are going to show it for n. Let n be the order of T and let $e = \{u,v\}$ be an edge of T. Since we have already proven that **(1)** $\Leftrightarrow$ **(2)** we can affirm that this edge is the only path that joins vertices u and v; and, therefore, the graph $T-e$ is formed exactly by two components T_u, which contain u, and T_v, which contains v. Each one of these components is a tree, as it is a subgraph of T. Since T_u and T_v have order less than n, we can apply the hypothesis of induction to obtain $m(T_u) = n(T_u) - 1$ and $m(T_v) = n(T_v) - 1$. Therefore,

$$m(T) = m(T_u) + m(T_v) + 1 = n(T_u) - 1 + n(T_v) - 1 + 1 = n - 1.$$

Conversely, assume that T is a connected graph of order n and size $m(T) = n - 1$. It is necessary to show that T is acyclic. We already know that a connected graph of order n has to have a minimum of $n - 1$ edges. So, if T has size $m(T) = n - 1$ and it contains a cycle, then we can deleted an edge of the cycle, and the resulting graph would continue being connected but would have size $n - 2$, which is not possible. Therefore, T is acyclic.

(1) $\Leftrightarrow$ **(4)**. The implication **(1)** $\Rightarrow$ **(4)** is deduced directly from the definition of a tree and from the previous equivalence. Conversely, we have to show that if T is an acyclic graph of order n and size $n - 1$ then T is connected. Let $T_1, \ldots, T_k$ ($k \geq 1$) be the components of T. Since each T_i does not contain cycles and is connected, it will be a tree and $n(T_i) = m(T_i) - 1$. Therefore,

$$n - 1 = m = \sum_{i=1}^{k} m(T_i) = \sum_{i=1}^{k} (n(T_i) - 1) = n - k$$

From this last equality we can deduce that $k = 1$ and, therefore, T is connected. □

Definition 43. A *leaf* of a tree is a vertex of degree 1.

Proposition 46. *All trees with a minimum of two vertices have a minimum of two leaves.*

Proof. Let $T = (V, E)$ be a tree of order n and let H be the set of the leaves. Since $|E| = n - 1$, by the degree sum formula we have:

$$\begin{aligned} 2(n-1) &= 2|E| = \sum_{v \in V} \delta(v) = \sum_{v \in H} \delta(v) + \sum_{v \notin H} \delta(v) = \\ &= \sum_{v \in H} 1 + \sum_{v \notin H} \delta(v) \geq |H| + \sum_{v \notin H} 2 = |H| + 2(n - |H|) \end{aligned}$$

Thus, from the previous inequality we can derive $|H| \geq 2$. □

Exercise 98. Show that a forest of order n formed by k trees has size $n - k$.

Solution: In effect, the forest $G = (V, E)$ will be the union of components $T_1, \ldots, T_k$ ($k \geq 1$), which are also trees. The result $m(T_i) = n(T_i) - 1$ can be applied to each one of them and, therefore, we can write

$$|E| = \sum_{i=1}^{k} m(T_i) = \sum_{i=1}^{k} (n(T_i) - 1) = \left(\sum_{i=1}^{k} n(T_i) \right) - k = n - k.$$

□

Exercise 99. Find the number of leaves of a tree that have one vertex of degree 3, three vertices of degree 2 and the remaining vertices of degree 1.

Solution: Remember that if $T=(V,E)$ is a tree, then $m=n-1$, with n being the order and m the size of T. If x is the number of leaves, it means that $n=x+3+1$, and if the degree sum formula is applied we have :

$$x+3\cdot 2+3=\sum_{v\in V}\delta(v)=2(n-1)=2(x+3+1-1)=2x+6,$$

Thus, $x=3$. The sequence of degrees is $3,2,2,2,2,1,1,1$. □

Example 69. Let $T=(V,E)$ be a tree of order $n=9$ that has three vertices of degree 3. What is the complete sequence of degrees?

Solution: Let us suppose that $V=\{v_1,\dots,v_9\}$ and that $x_i=\delta(v_i)$, $i=1,\dots,9$, are the degrees of the vertices of the tree. The sequence of degrees is $x_1,x_2,x_3,x_4,x_5,x_6,3,3,3$.

By the degree sum formula we have,

$$16=2(n-1)=2|E|=\sum_{v\in V}\delta(v)=x_1+x_2+x_3+x_4+x_5+x_6+3+3+3$$

Thus, $x_1+x_2+x_3+x_4+x_5+x_6=7$. Since any tree is connected, it cannot have vertices of degree 0, which means that $x_1,x_2,x_3,x_4,x_5,x_6\geq 1$ and, in consequence, the value of these unknowns degrees has to be 1, except one that has to be 2. Therefore, the complete sequence is 3, 3, 3, 2, 1, 1, 1, 1, 1. □

Exercise 100. Let $T=(V,E)$ be a tree of order $n\geq 2$.

(a) Prove that the number of leaves is $2+\sum\limits_{\delta(v)\geq 3}(\delta(v)-2)$.

(b) If every vertex of T that is not a leaf has degree 4, prove that the number of leaves is $2x_4+2$, where x_4 is the number of vertices of degree 4.

Solution:

(a) Let x_1 be the number of leaves of T, x_2 the number of vertices of degree 2, and let x_3 be the number of vertices with degree greater than or equal to 3. According to the degree sum formula we have,

$$\begin{aligned}
2m &= \sum_{v\in V}\delta(v)\\
2(n-1) &= \sum_{v\in V}\delta(v)\\
2(x_1+x_2+x_3-1) &= x_1+2x_2+\sum_{\delta(v)\geq 3}\delta(v)\\
x_1 &= 2-2x_3+\sum_{\delta(v)\geq 3}\delta(v)\\
x_1 &= 2+\sum_{\delta(v)\geq 3}(\delta(v)-2).
\end{aligned}$$

(b) As before, let x_1 be the number of leaves of T. According to the degree sum formula $2(n-1) = x_1 + 4x_4$. As $n = x_1 + x_4$, we obtain $x_1 = 2x_4 + 2$. □

Exercise 101. Consider a connected graph G of size 20 that has one vertex of degree 5, three vertices of degree 4, two of degree 3, two of degree 2 and the other vertices with degree 1.

(a) Find the order of G.

(b) Find the number of cycles of G.

(c) Is G bipartite? Justify your answer.

(d) Find the size of the complement of G.

Solution:

(a) Let x be the number of vertices of degree 1 of G. The order of G is $n = 1+3+2+2+x$. Applying the degree sum formula we have

$$20 = \frac{1\cdot 5 + 3\cdot 4 + 2\cdot 3 + 2\cdot 2 + x\cdot 1}{2}.$$

Thus, $x = 13$ and $n = 21$.

(b) G is connected and has size $m = n-1$. Therefore, G is a tree, and does not have cycles.

(c) G is bipartite since it is a tree.

(d) The size of the complement of G is $\binom{21}{2} - 20 = 190$. □

Exercise 102. Let T be a tree such that the degree of every vertex belongs to the set $\{1,2,4,5\}$. Let $x_i(T)$ be the number of vertices of degree i, where $x_4(T) = 8$ and $x_5(T) = 5$.

(a) Find $x_1(T)$.

(b) If there is no adjacency between vertices of degree two, what is the maximum number of vertices of T?

Solution: Since $m(T) = n(T) - 1$ and $2m(T) = \sum_{v\in V(T)} \delta(v)$,

$$2(x_1(T) + x_2(T) + x_4(T) + x_5(T) - 1) = x_1(T) + 2x_2(T) + 4x_4(T) + 5x_5(T).$$

Thus, $x_1(T) = 2 + 2x_4(T) + 3x_5(T) = 33$.

Now, let T_0 be the tree satisfying the restrictions above and $x_2(T_0) = 0$, and let T_{max} be the tree with maximum number of vertices of degree two under the restriction that there is no adjacency between vertices of degree two. Observe that T_{max} can be obtained from T_0 by inserting an additional vertex of degree two per each edge of T_0. That is, each edge of T_0 becomes a path of length two in T_{max}, and so $x_1(T_{max}) = x_1(T_0) = 33$, $x_2(T_{max}) = m(T_0) = n(T_0) - 1 = 45$, $x_4(T_{max}) = x_4(T_0) = 8$ and $x_5(T_{max}) = x_5(T_0) = 5$. Therefore, $n(T_{max}) = 91$. □

5.2 Minimum spanning trees

Definition 44. Given a weighted graph (G,w) and a spanning tree T of G, we define the *weight of the tree* T as $w(T)=\sum_{e\in E(T)} w(e)$. A *minimum spanning tree* of G is a spanning tree T of G of minimum weight among all the spanning trees of G.

If G is not connected, then we can speak of the *minimum spanning forest* of G as that which has the minimum weight among all the spanning forests of G.

Kruskal algorithm

To construct a minimum spanning tree we begin with an "empty tree"; throughout the process a subgraph will grow with new edges in the following way: in each stage of the process an edge is added that does not form a cycle with those previously chosen. To guarantee the minimality, we choose, among the possibilities, the edge of minimum weight (not necessarily adjacent to some previously incorporated edge). The process ends when $n-1$ edges have been incorporated.

Formulation of the Kruskal algorithm

Imput: a weighted connected graph (G,w) of order n.
Output: a minimum spanning tree T of G.
Algorithm:

Start
  $k \leftarrow 1$, $T=(V,E')$, $E'=\varnothing$
  while $k \leq n-1$
    Choose the edge $e\in E$ of minimum weight, not chosen previously
    so that the subgraph $T=(V,E'\cup e)$ is acyclic.
    Add e to E'.
    $k \leftarrow k+1$
  endwhile
  Return(T)
end.

Implementation of the Kruskal algorithm

To implement the Kruskal algorithm we need to keep an orderly list by weight of the edges of the graph and a structure that allows us to check, in an efficient way, that cycles are not formed.

Intuitively, the algorithm keeps a forest. At the beginning, there exist $|V|$ trees of a single vertex. When we add an edge two trees are combined into one. When the algorithm finishes only one tree exists, the minimum spanning tree.

Structures needed for the implementation of the algorithm:

- A weighted and connected graph (G,w) represented by means of a list of adjacencies.

- A list of edges, X, ordered according to weight.
- A structure designated *union-search* of sets. In principle, each vertex forms a set (that only contains this vertex). When an edge $\{u,v\}$ is analysed, a *search* is done to locate the set of u and the one of v. If u and v are in the same set, the edge is not accepted because u and v already are connected and, adding the edge $\{u,v\}$ would form a cycle. In the opposite case, the edge is accepted and a *union* is made of the two groups that contain u and v to form a new tree.

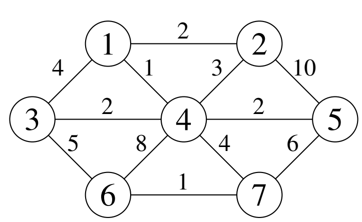
  Thus, the two operations that have to be done are:

  - *search* (U,u), gives back the representative of the tree that contains u.
  - *union*(U,x,y), merges the trees that have as representatives x, y to form a new tree.
- A tree T that will contain a minimum spanning tree.

Input: (G,w) connected and weighted, of order n
Output: a minimum spanning tree T of G
Algorithm: Kruskal (G)

```
Start
    U=structure union-search
    X=list of edges ordered by weight in ascending order
    T ← (V,E'), E' ← ∅
    k ← 1
    i ← 1
    While (k < n)
        Let {u,v} the edge X[i]
        x ← Search (U,u)
        y ← Search (U,v)
        If (x ≠ y)
            E' ← E' ∪ {u,v}
            k ← k+1
            Union (U,x,y)
        endif
        i ← i+1
    endwhile
    Return (T)
End
```

Exercise 103. Apply the Kruskal algorithm to find a minimum spanning tree of the graph of the figure:



Solution: In this case, it is only necessary to make a list of the edges from less weight to more weight (in case of equality, enter first those that are formed by vertices of lower weight).

Edges	Weights
$\{1,4\}$	1
$\{6,7\}$	1
$\{1,2\}$	2
$\{3,4\}$	2
$\{4,5\}$	2
$\{2,4\}$	3
$\{1,3\}$	4
$\{4,7\}$	4
$\{3,6\}$	5
$\{5,7\}$	6
$\{4,6\}$	8
$\{2,5\}$	10

Since there are seven vertices, we have to choose the first six edges that do not form any cycle. We will mark them with an asterisk; the rejected edges because they form cycles will be marked in bold.

Edges	Weights
$\{1,4\}*$	1
$\{6,7\}*$	1
$\{1,2\}*$	2
$\{3,4\}*$	2
$\{4,5\}*$	2
$\{\mathbf{2},\mathbf{4}\}$	
$\{\mathbf{1},\mathbf{3}\}$	
$\{4,7\}*$	4
$\{3,6\}$	
$\{5,7\}$	
$\{4,6\}$	
$\{2,5\}$	

Therefore, the minimum spanning tree will be formed by the edges: $\{1,4\}$, $\{6,7\}$, $\{1,2\}$, $\{3,4\}$, $\{4,5\}$, $\{4,7\}$ with a total weight of 12. □

Formulation of the Prim algorithm

Input: a weighted connected graph (G,w) of order n.
Output: a minimum spanning tree T of G.
Algorithm:

Start
$k \leftarrow 1$, $T = (V,E')$, $E' = \varnothing$

while $k \leq n-1$
Choose the edge $a \in E$ of minimum weight, not chosen previously,
adjacent to some edge of E' and
such that the subgraph $T = (V, E' \cup a)$ is acyclic.
Add a to E'.
$k \leftarrow k+1$
endwhile
Return(T)
end.

Implementation of the Prim algorithm

The Prim algorithm builds the spanning tree making a single tree grow in successive steps. It begins by choosing any vertex as a starting vertex. In each step we add the edge of minimum weight that connects a vertex of the tree with one from outside.

The implementation of the Prim algorithm is essentially identical to the Dijkstra algorithm to find the distance between two vertices. Only the update rule has to be modified.

Structures needed for the implementation of the algorithm:

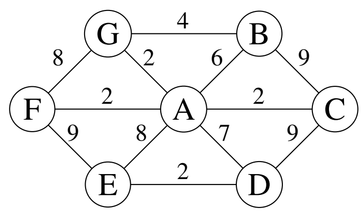
- A weighted graph (G, w) represented by means of an adjacency list.
- A set U of vertices that have been visited, in the order done.
- A table of weights, $X(\cdot)$, indexed by the vertices of G, that registers the weight of the edge of minimum weight that connects a vertex v with an already visited vertex.
- At the end, the table $X(\cdot)$ registers the weights of the edges that form a minimum spanning tree.

In each step, a label of one vertex of the graph is attached. Thus, after n steps we will have calculated the minimum spanning tree.

Prim Algorithm of
Input a weighted connected graph (G, w) of order n
T, a minimum spanning tree of G
Prim Algorithm (G)
Start
We select a starting vertex $u_0 \in V$
$U \leftarrow \varnothing$
for $v \in V \setminus \{u_0\}$
$X(v) \leftarrow \infty$
Label v with $(X(v), u_0)$
endfor
$X(u_0) \leftarrow 0$
Label u_0 with $(0, u_0)$
$T \leftarrow (V, E')$, $E' \leftarrow \varnothing$
For $i \leftarrow 1$ until n

u_i vertex reached by $\min\{X(v) \mid v \in V - U\}$
$U \leftarrow U \cup \{u_i\}$
$E' \leftarrow E' \cup \{x, u_i\}$ /*where $(X(u_i), x)$ is the label of u_i*/
for $v \in V - U$ adjacent to u_i
If $w(u_i, v) < X(v)$
Then $X(v) \leftarrow w(u_i, v)$
Label v con $(X(v), u_i)$
endif
endfor
endfor
Return (T)
End

Exercise 104. Apply the Prim algorithm to determine a minimum spanning tree of the graph



Solution: If vertex B is chosen as the starting vertex, the following table is obtained:

A	B	C	D	E	F	G
(∞,B)	(0,B)	(∞,B)	(∞,B)	(∞,B)	(∞,B)	(∞,B)
(6,B)	(0,B)*	(9,B)	(∞,B)	(∞,B)	(∞,B)	(4,B)
(2,G)	(0,B)	(9,B)	(∞,B)	(∞,B)	(8,G)	(4,B)*
(2,G)*	(0,B)	(2,A)	(7,A)	(8,A)	(2,A)	(4,B)
(2,G)	(0,B)	(2,A)*	(7,A)	(8,A)	(2,A)	(4,B)
(2,G)	(0,B)	(2,A)	(7,A)	(8,A)	(2,A)*	(4,B)
(2,G)	(0,B)	(2,A)	(7,A)*	(2,D)	(2,A)	(4,B)
(2,G)	(0,B)	(2,A)	(7,A)	(2,D)*	(2,A)	(4,B)

Express the spanning tree in the following format edge=weight:
$\{G,A\} = 2$, $\{A,C\} = 2$, $\{A,D\} = 7$, $\{D,E\} = 2$, $\{A,F\} = 2$ and $\{B,G\} = 4$. Therefore, the spanning tree has weight 19. □

Exercise 105. Apply the Prim algorithm, starting from vertex B, to the weighted graph of vertices A, B, C, D, E, and F, whose adjacency matrix is

$$\begin{pmatrix} 0 & 8 & 4 & 0 & 2 & 0 \\ 8 & 0 & 3 & 0 & 6 & 0 \\ 4 & 3 & 0 & 7 & 0 & 6 \\ 0 & 0 & 7 & 0 & 9 & 3 \\ 2 & 6 & 0 & 9 & 0 & 4 \\ 0 & 0 & 6 & 3 & 4 & 0 \end{pmatrix}$$

Solution: The following table shows the steps of the algorithm.

A	B	C	D	E	F
(∞,B)	(0,B)	(∞,B)	(∞,B)	(∞,B)	(∞,B)
(8,B)	(0,B)*	(3,B)	(∞,B)	(6,B)	(∞,B)
(4,C)	(0,B)	(3,B)*	(7,C)	(6,B)	(6,C)
(4,C)*	(0,B)	(3,B)	(7,C)	(2,A)	(6,C)
(4,C)	(0,B)	(3,B)	(7,C)	(2,A)*	(4,E)
(4,C)	(0,B)	(3,B)	(3,F)	(2,A)	(4,E)*
(4,C)	(0,B)	(3,B)	(3,F)*	(2,A)	(4,E)

We express the spanning tree in the following format edge=weight:
$\{C,A\} = 4$, $\{B,C\} = 3$, $\{F,D\} = 3$, $\{A,E\} = 2$, $\{E,F\} = 4$. Therefore, the spanning tree has weight 16. □

Exercise 106. Apply the Prim algorithm, starting from vertex A, to the weighted graph of vertices A, B, C, D, E, and F, whose adjacency matrix is

$$\begin{pmatrix} 0 & 8 & 0 & 0 & 5 & 0 \\ 8 & 0 & 7 & 2 & 2 & 0 \\ 0 & 7 & 0 & 8 & 0 & 3 \\ 0 & 2 & 8 & 0 & 0 & 4 \\ 5 & 2 & 0 & 0 & 0 & 9 \\ 0 & 0 & 3 & 4 & 9 & 0 \end{pmatrix}$$

Solution: The following table shows the steps of the algorithm.

A	B	C	D	E	F
(0,A)	(∞,A)	(∞,A)	(∞,A)	(∞,A)	(∞,A)
(0,A)*	(8,A)	(∞,A)	(∞,A)	(5,A)	(∞,A)
(0,A)	(2,E)	(∞,A)	(∞,A)	(5,A)*	(9,E)
(0,A)	(2,E)*	(7,B)	(2,B)	(5,A)	(9,E)
(0,A)	(2,E)	(7,B)	(2,B)*	(5,A)	(4,D)
(0,A)	(2,E)	(3,F)	(2,B)	(5,A)	(4,D)*
(0,A)	(2,E)	(3,F)*	(2,B)	(5,A)	(4,D)

We express the spanning tree in the following format edge=weight:
$\{E,B\} = 2$, $\{F,C\} = 3$, $\{B,D\} = 2$, $\{A,E\} = 5$, $\{D,F\} = 4$. Therefore, the spanning tree has weight 16. □

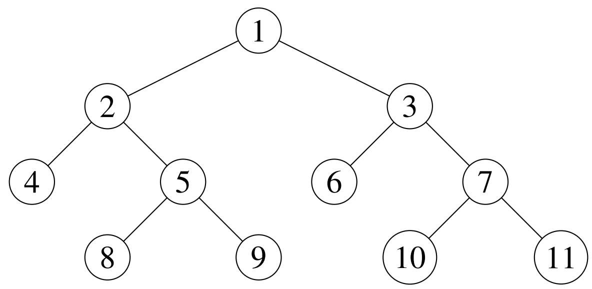
5.3 Exploration of binary trees

Definition 45. Let $T = (V,E)$ be a tree.

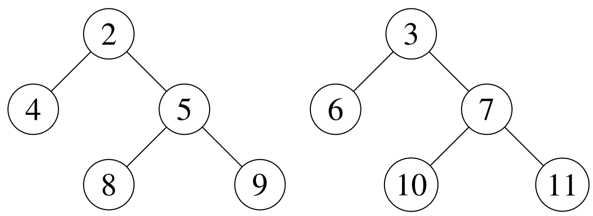
- T is a *rooted tree* if one of its vertices has been designated as the root.
- The *level* of a vertex of a rooted tree is the length of the only path that goes from the root to it. The level of the root is 0.
- Any vertex of a rooted tree that is not a leaf is an *internal vertex*.

- If in a rooted tree, v is the vertex that immediately precedes w on the path from the root to w, then v is the *parent* of w and w is a *child* of v.
- An *m-ary* tree is a rooted tree in which each internal vertex has at most m children, and in which at least one vertex has exactly m children. A 2-ary tree is usually called a *binary tree*.
- A *complete m-ary* tree is an m-ary tree in which each internal vertex has exactly m children and all the leaves have the same level.

Example 70. The tree in the figure is a binary tree.



The root is the vertex labelled with 1. The root has two children with labels 2 and 3, that are roots of two subtrees:



Recursively, we also could consider the subtrees of roots 4, 5, 6,...

Exercise 107. Given a complete m-ary tree T of order 2396745, find the value of m knowing that T has 2097152 leaves.

Solution: Let r be the level of the leaves of T. The order of T is

$$1+m+m^2+m^3+\cdots+m^r=\frac{m^{r+1}-1}{m-1}=2396745.$$

Thus, $m^{r+1}=2396745m-2396744$ and dividing by m obtains

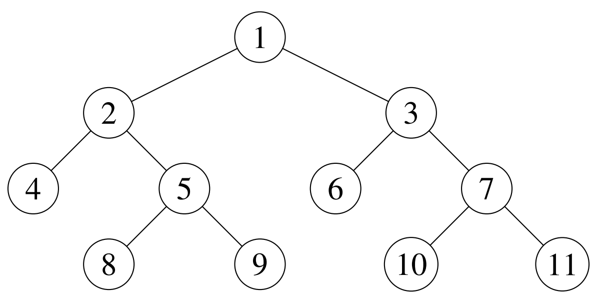
$$m^r=2396745-\frac{2396744}{m}.$$

As the number of leaves of T is $m^r=2097152$, we obtain $m=8$. □

A systematic visit of each vertex of a tree is called a tree traversal. Thus, a *traversal* of a binary tree is a global ordering of its vertices. In the specific case of binary trees, there are three basic tree traversals: *preorder*, *inorder* and *postorder*. The three traversals are distinguished basically in the way they explore the two children of each vertex. Before studying them one has to enter the following notation: T is the binary tree of root r, T_1 and T_2 are the subtrees of roots v_1 and v_2 induced by the children of root r, where v_1 is on the left and v_2 on the right in a plane drawing of the tree. These traversals are defined recursively in the following way.

- **Preorder traversal** of T:
  1. List the root r.
  2. Perform a preorder traversal of the left subtree T_1.
  3. Perform a preorder traversal of the right subtree T_2.
- **Inorder traversal** of T:
  1. Perform an inorder traversal of the left subtree T_1.
  2. List the root r.
  3. Perform an inorder traversal of the right subtree T_2.
- **Postorder traversal** of T:
  1. Perform a postorder traversal of the left subtree T_1.
  2. Perform a postorder traversal of the right subtree T_2.
  3. List the root r.

Exercise 108. Give the preorder, inorder and postorder traversals of the tree



Solution: The traversals are:

Preorder: $1,2,4,5,8,9,3,6,7,10,11$;

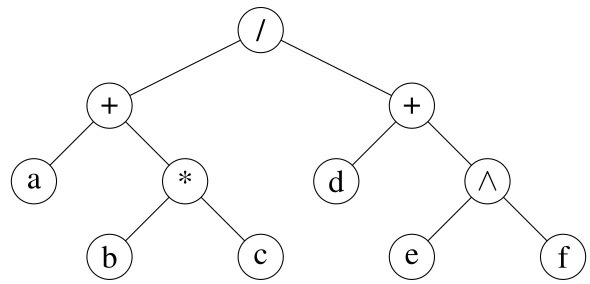
Inorder: $4,2,8,5,9,1,6,3,10,7,11$;

Postorder: $4,8,9,5,2,6,10,11,7,3,1$.

□

Exercise 109. Find the preorder, inorder and postorder traversals of the tree with the arithmetic expression $(a+b*c)/(d+e\hat{}f)$.

Solution: The tree of the expression is



The three traversals are:

Preorder: $/ \, + \, a \, * \, b \; c \, + \, d \, \hat{} \; e \; f$.

Inorder: $a \, + \, b \, * \, c \, / \, d \, + \, e \, \hat{} \, f$

Postorder: $a \; b \; c \, * \, + d \; e \; f \, \hat{} \, + \, /$

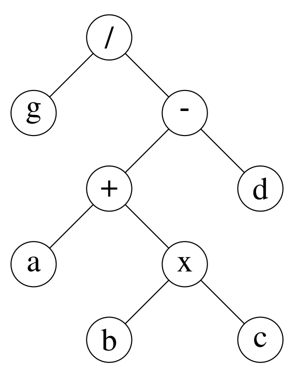
□

For the arithmetic expressions the preorder traversal gives place to a type of representation of the expression designated *prefix notation* (or *Polish notation*, originated by the logician Jan Lukasiewicz). The norder traversal gives place to *unfix notation* (which coincides with the conventional without bracket). Finally, the postorder traversal gives place to *the postfix notation* (or also Polish *reverse*).

The prefix and postfix notations are especially adapted to evaluate arithmetic expressions. Compilers are the attendants to transform the conventional expressions to the prefix or postfix notation, according to the case.

Exercise 110. Represent the arithmetic expression $g/((a+(b\times c))-d)$ by means of a binary tree. Determine the preorder and postorder traversals of the binary tree.

Solution: The binary tree of the arithmetic expression is



Preorder traversal: $/ \, g \, - \, + a \times b \; c \; d$.

Postorder traversal: $g \; a \; b \; c \times + d \, - /$.

□

Exercise 111. Determine the preorder, inorder and postorder traversals of the tree of the arithmetic expression that corresponds to the number of words of maximum length r, in an alphabet of n elements.

Solution: The number of words of maximum length r, in an alphabet of n elements is $\frac{n^{r+1}-1}{n-1}$. The traversals are:

Preorder: $/ \; - \; \wedge \, n \; + \; r \; 1 \; 1 \; - \; n \; 1$

Inorder: $n \wedge r \; + \; 1 \; - \; 1 \, / \, n \; - \; 1$

Postorder: $n \; r \; 1 \; + \; \wedge \; 1 \; - \; n \; 1 \; - \; /$

□

Exercise 112. Determine the preorder, inorder, and postorder traversals of the tree of the arithmetic expression (to+b/c)∗(d∗ (and+f)+g).

Solution: The traversals are:

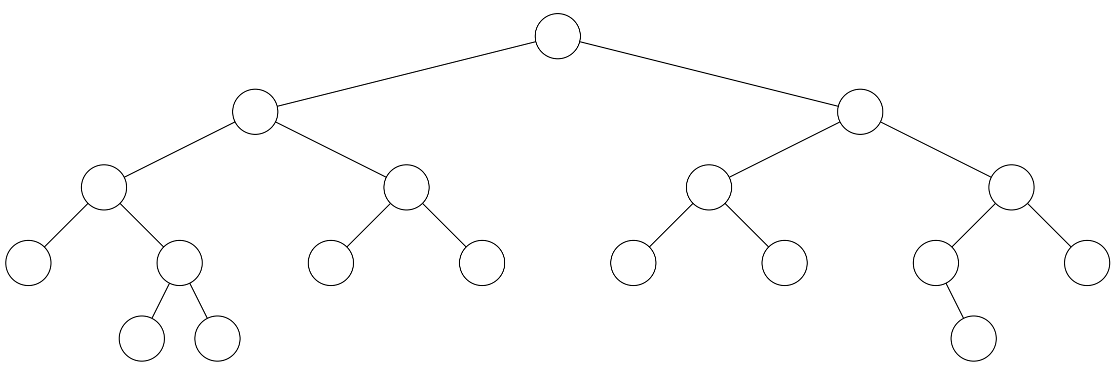
Preorder, ∗ + to / b c + ∗ d + and f g

Inorder, to + b / c ∗ d ∗ and + f + g

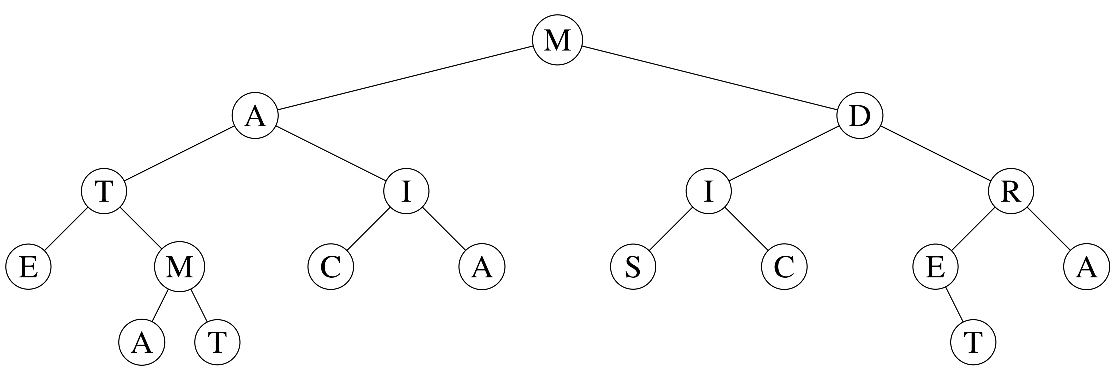
Postorder, to b c / + d and f + ∗ g + ∗

□

Exercise 113. Label the vertices of the following tree so that the preorder traversal is MATEMATICA DISCRETA.

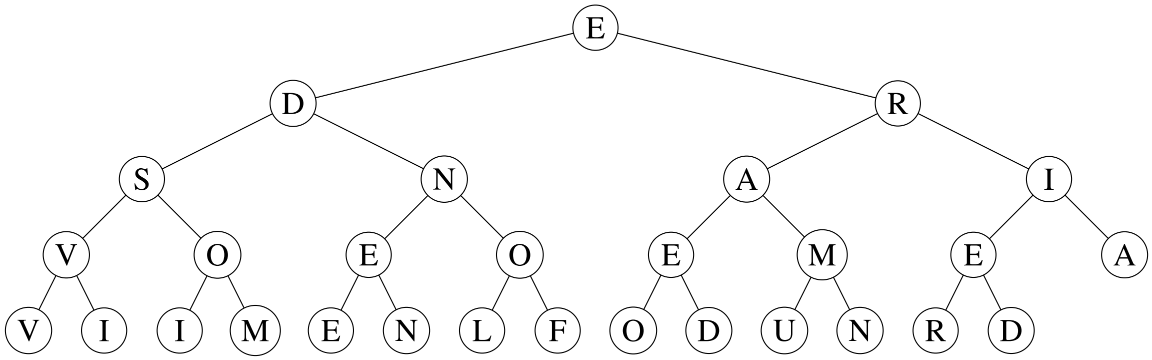


Solution: The disposition of the labels in the tree is:



□

Exercise 114. Determine the sentence that results when visiting the nodes of following tree in postorder traversal.



Solution: The sentence is: VIVIMOS EN EL FONDO DE UN MAR DE AIRE, which in English means WE LIVE AT THE BOTTOM OF A SEA OF AIR. □

Chapter 6

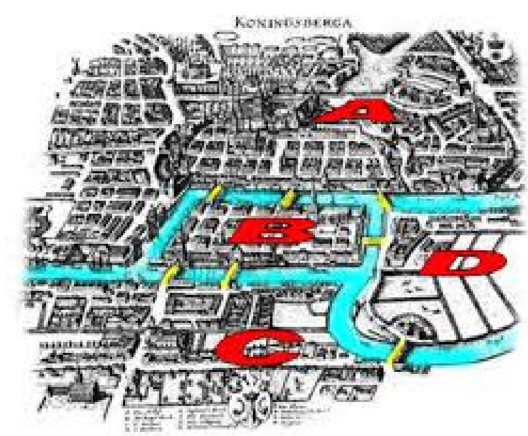
Eulerian Graphs and Hamiltonian Graphs

In this section we will study two classical problems of graph exploration:

- Visit all the edges of the graph returning to the starting vertex without having repeated edges (Eulerian circuit).
- Visit all the vertices of the graph returning to the starting vertex without having repeated vertices (Hamiltonian cycle).

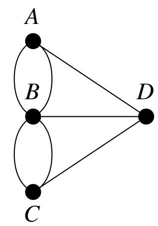
6.1 Eulerian Graphs

In the XVIII century the ancient city of Königsberg (currently Kaliningrado) was divided into four zones by the river Pregel. Seven bridges communicated these zones as shown in the following chart:



City of Königsberg

One of the entertainments of the citizens of Königsberg consisted in looking for a route that crossed each bridge once, and going back to the exact starting point. In terms of graphs this



entertainment is equivalent to looking for a walk that goes through each edge of the multigraph G exactly once.

Leonard Euler proved that it did not exist a route that fulfilled these conditions and characterised the graphs (multigraphs) that do contain it: these are the *Eulerian graphs*.

Definition 46. In a graph (multigraph) $G = (V, E)$:

- An Eulerian *circuit* is a circuit that goes through all the edges of the graph. If a graph admits a circuit of these characteristics, it is designated an *Eulerian graph*.

- An *Eulerian trail* is an open walk that contains all the edges of the graph without repetition. If a graph admits a trail of these characteristics, it is designated a *semi-Eulerian graph*.

Theorem 47 (Leonhard Euler, 1736)**.** *A graph is Eulerian if and only if every vertex has an even degree.*

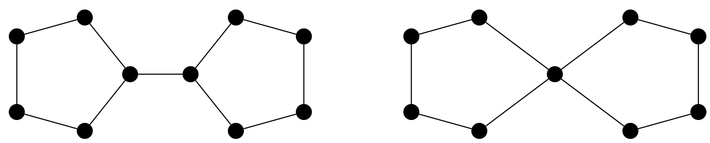
Proof. Let us suppose that the graph is Eulerian and check that the degree of any vertex is even. Let Q be an Eulerian circuit that, in the measure that it contains all the edges, goes through all the vertices; let v be any vertex, that has to belong to Q, and that eventually can be repeated. Each appearance of the vertex in the circuit contributes twice to the degree of the vertex (we have to explain the edge by which we "arrive", that is used for the first time as they cannot repeat edges, and have to "go out" by an edge still not used); the contribution is double because the adjacent edges that appear in the sequence have not been used previously; if k is the number of appearances of the vertex, then the degree is $\delta(v) = 2k$, that is, even.

Conversely, we assume that the degree of every vertex is even and go to prove that the graph is Eulerian. We have to find an Eulerian circuit in the graph G.

We select an arbitrary vertex v and form a circuit Q that starts in v and continues as indicated below. If we begin at v, every time that it reaches a vertex it selects an edge that is incident and that has not been used previously; we continue with another endpoint vertex. Since the degree is even, in each vertex we have an edge of exit for each edge of arrival; the only exception to this rule is that of the initial vertex v, which uses an edge of exit without having used any of entrance and, therefore, if it arrives at a vertex that no longer has more edges with which to continue the route Q, this vertex is the initial v.

When this happens there are two possibilities. One is that Q already contains all the edges of the graph, in which case we have already finished and this is the Eulerian circuit that we wanted to build. Another possibility is that it does not contain all the edges. In this case we have built a circuit in the graph and then, by connection, some vertex of the circuit is incident with another non-included edge in Q; this vertex can be used as a starting point for the construction of a new circuit Q' that does not contain edges of Q and that returns to the starting vertex, since in $G-Q$ each vertex is still of even degree. The union $Q \cup Q'$ is a circuit; if it contains all the edges, we have finished. If not, it repeats the process until it obtains an Eulerian circuit (the process finishes due to the fact that the graph is finite). □

Example 71. The right hand side graph corresponds to an Eulerian graph and the other one to a semi-Eulerian graph.



Corollary 48. *A graph is semi-Eulerian if and only if it has exactly two vertices of odd degree.*

Proof. If G contains an Eulerian trail $u-v$ then the degree of every vertex of G, which is different from u and v, will be an even number (with the same reasoning used for the Eulerian circuits) and the degree of u and v will be odd.

If G contains exactly two vertices u and v of odd degree, then the multigraph $G+uv$ is connected and has all the vertices of even degree. Therefore, if we apply the previous theorem, $G+uv$ contains an Eulerian circuit C. If we delete from this circuit C the edge $\{u,v\}$ we will obtain an Eulerian $u-v$ trail in G. □

Theorem 49. *A connected non-trivial graph G is Eulerian if and only if the set of edges can be partitioned into cycles.*

Proof. Let G be Eulerian. As G is not trivial, each vertex has an even degree greater than or equal to 2, so G contains some cycle C. When erasing all the edges of C, we obtain a subgraph G_1 of G which does not have vertices of odd degree. If G_1 is empty, then the set of edges of G form the cycle C. If G_1 is not empty, then applying the above reasoning, we obtain a cycle C_1, and erasing the edges of C_1 we obtain a subgraph G_2 of G_1 which does not have vertices of odd degree. Continuing this process until we obtain an empty graph G_k, we obtain a partition of the edges of G in cycles.

Let us now assume that the set of edges of G can be partitioned into cycles. Let C be a cycle of the partition. If $G \cong C$, then G is Eulerian. In the opposite case, take a cycle C_1 that has a vertex v_1 in common with C. The route that splits off from v_1, visits C and afterwards visits C_1 is a circuit that contains all the edges of C and C_1. By continuing this process we obtain an Eulerian circuit of G. □

The resultant algorithm of the previous theorem (first attributed to Hierholzer, 1873) builds an Eulerian circuit from the concatenation of disjoint circuits, that is, without edges in common.

Formulation of the Hierholzer algorithm

Input: a connected and Eulerian graph $G=(V,A)$ of order n and an initial vertex $s \in V$.
Output: an Eulerian circuit C of G represented as a list of vertices.

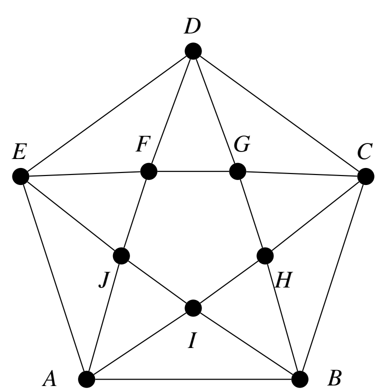
Algorithm <u>EulerianCircuit</u> (G,s)
Start
$C \leftarrow \{s\}$
While $A \neq \varnothing$
$v \leftarrow$ PositiveDegreeVertex(G,C)
$C' \leftarrow Circuit(G,v)$
$C \leftarrow Concatenate(C,C',v)$
$G \leftarrow G - C'$
endwhile
Return (C)
End

In the algorithm we have used the following functions:

- *PositiveDegreeVertex*(G,C) gives back the first vertex C that has positive degree in G.
- *Circuit*(G,v) gives back a circuit C' built in the graph G from vertex v.
- *Concatenate*(C,C',v) gives back the circuit that obtained by substituting the first appearance of the vertex v in the circuit C by the whole of the circuit C'.
- The instruction $G \leftarrow G - C'$ deletes from G the edges of the circuit C'.

Simulation of the Hierholzer algorithm

Exercise 115. Considers the Eulerian graph defined by the following chart:



Solution: When applying the Hierholzer algorithm we obtain the following table:

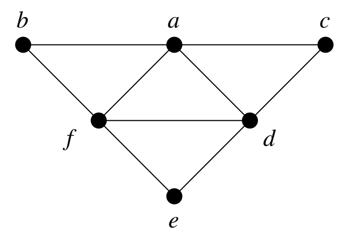
Iteration	v	C'	C
0	A		$\{A\}$
1	A	$\{A,B,I,A\}$	$\{A,B,I,A\}$
2	A	$\{A,E,J,A\}$	$\{A,E,J,A,B,I,A\}$
3	B	$\{B,C,H,B\}$	$\{A,E,J,A,B,C,H,B,I,A\}$
4	C	$\{C,D,G,C\}$	$\{A,E,J,A,B,C,D,G,C,H,B,I,A\}$
5	D	$\{D,E,F,D\}$	$\{A,E,J,A,B,C,D,E,F,D,G,C,H,B,I,A\}$
6	F	$\{F,G,H,I,J,F\}$	$\{A,E,J,A,B,C,D,E,F,G,H,I,J,F,D,G,C,H,B,I,A\}$

Thus, the Eulerian circuit obtained is

$$C = \{A, E, J, A, B, C, D, E, F, G, H, I, J, F, D, G, C, H, B, I, A\}.$$

□

Exercise 116. Apply the Hierholzer algorithm to determine an Eulerian circuit of the graph in the figure.



Solution: When applying the Hierholzer algorithm we obtain the following table:

Iteration	v	C'	C
0	a		$\{a\}$
1	a	$\{a,b,f,a\}$	$\{a,b,f,a\}$
2	a	$\{a,c,d,a\}$	$\{a,c,d,a,b,f,a\}$
3	d	$\{d,e,f,d\}$	$\{a,c,d,e,f,d,a,b,f,a\}$

Thus, the Eulerian circuit obtained is $\{a,c,d,e,f,d,a,b,f,a\}$ □

Analysis of the Hierholzer algorithm

In the Hierholzer algorithm the following operations can be distinguished:

1. Choose a vertex v of positive degree in the list C. This is a linear operation in the size of C. C will contain, at most, n different vertices. Therefore, it will be an operation of complexity $O(n)$.

2. Build a circuit C' in G from vertex v. This is an operation that depends on the number of edges chosen. In the worst of cases it will be very like the size m of the graph. It will be, then, a function of complexity $O(m)$.

3. Concatenate the circuits C and C'. This is a linear operation in the size of C, that is, of complexity $O(n)$.

4. Delete from G the edges of the circuit C'. Trivially, this an operation that, in the worst of cases, will have a complexity $O(m)$.

In summary, since the graph is connected we can conclude that the complexity of the entire algorithm will be of

$$\max\{O(n), O(m), O(n), O(m)\} = O(m).$$

6.2 Hamiltonian Graphs

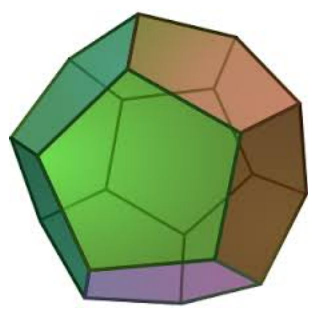
The origin of Hamiltonian Graphs is in the Hamilton's game, in which one has to find a closed walk without repetition of vertices, through the edges of a regular dodecahedron. Other problems like that of the chessboard, that of the travelling salesman, and that of the welding robot controlled by computer have motivated the study of these graphs.

Definition 47. In a graph $G = (V,A)$:

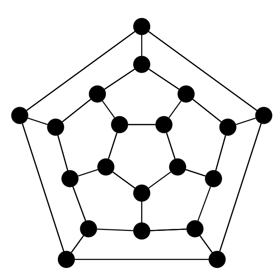
- A walk is a Hamiltonian *path* if it goes through all the vertices without repetition.
- A Hamiltonian *cycle* is a cycle that goes through all the vertices of the graph. If the graph admits a cycle of these characteristics, it is called a *Hamiltonian Graph*.

These definitions only make sense in the case of a connected graph; in an opposite case, it applies to each one of the components.

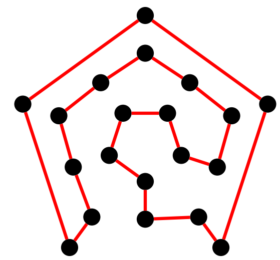
The mathematician W.R. Hamilton invented in 1856 a game designated "*The traveller's dodecahedron*", that consisted of a dodecahedron whose vertices represented the main cities of the world of that period; the goal was to find a closed walk along the edges of the polyhedron that went through all the vertices without repetition.



We can consider a model in terms of graph theory for this entertainment. As can be seen in the following figure, the vertices of the graph correspond to the vertices of the dodecahedron, and the edges of the graph correspond to the edges of the polyhedron. In this case, it involves looking for a closed walk on the graph that goes through all the vertices exactly once.



In this case, the problem has a solution as we can see in the following chart.



Since we can think that a Hamiltonian cycle is "analogous" to an Eulerian circuit, one might hope for a characterization of Hamiltonian graphs, as occurs in the case of Eulerian graphs. No such characterization is known, nor is there a quick way of determining whether a given graph is Hamiltonian. In fact, the problem is NP-complete, which makes unlikely the existence of a polynomial algorithm that determines if an arbitrary graph is Hamiltonian. Fortunately, there exist some necessary conditions of hamiltonicity and also, independently, sufficient conditions. Some of these conditions have little practical applicability.

Definition 48. A graph $G = (V,A)$ is *2-connected* if each pair of vertices u and v of G is connected by a minimum of two disjoint paths, that is, two paths in which the only vertices that are in common are u and v.

Theorem 50. *If $G = (V,A)$ is a Hamiltonian graph then it fulfils the following properties:*

(a) *G is connected and all its vertices have degree greater than or equal to 2.*

(b) *G is 2-connected.*

(c) *For all $S \subset V$, $S \neq \varnothing$ verifies $c(G-S) \leq |S|$, where $c(G-S)$ represents the number of components of the graph obtained from G after deleting the vertices (and the incident edges) in S.*

(d) *If G is (V_1,V_2)-bipartite, then $|V_1| = |V_2|$.*

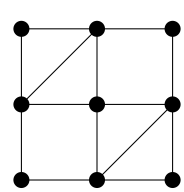
Proof. If G is Hamiltonian, then there is a cycle $C : v_1, v_2, \ldots, v_n, v_1$ containing all the vertices of G. Thus each vertex v_i connects, at least, with v_{i-1} and v_{i+1}. Therefore, $g(v_i) \geq 2$.

In the same way, between v_i and v_j $(i < j)$ there will be, at least, two disjointed paths: the path $v_i, v_{i+1}, \ldots, v_j$ and the path $v_j, v_{j+1}, \ldots, v_n, v_1, \ldots, v_i$. Therefore, it will be 2-connected.

If in a cycle we delete r vertices, then it will produce at most r components. Therefore, since G is Hamiltonian, whenever a set of vertices S is deleted, the number of components produced can not exceed the cardinality of S.

The last condition is a consequence of the distribution of the vertices of the graph G in a cycle. Necessarily, the vertices have to belong alternatively to V_1 and V_2. And this implies that $|V_1| = |V_2|$. □

Exercise 117. Study if the following graph is Eulerian; analyse also if it is Hamiltonian.



Solution: The graph is Eulerian because all the vertices have even degree. However, it is not Hamiltonian. If we consider the set S formed by the four vertices situated in the mid-points of the four sides, then $|S| = 4$, whereas $c(G-S) = 5$ contradicts condition 3 of the previous theorem. □

Exercise 118. Determine which of the following graphs are Hamiltonian and which are Eulerian.

(a) $G = K_n$ $(n \geq 3)$

(b) $G = N_1 + (K_4 \cup K_4)$

(c) $G = (K_n \cup K_m)^c$

Solution: Eulerian:

(a) K_n $(n \geq 3)$ is connected and for n odd it is Eulerian since the degree, $n-1$, is even.

(b) $G = N_1 + (K_4 \cup K_4)$ is Eulerian because it is connected and all the vertices have even degree: there is a vertex of degree 8 and the others have degree 4.

(c) Note that $G = (K_n \cup K_m)^c = K_{n,m}$ (is a complete bipartite graph). G is Eulerian if and only if n and m are even numbers.

Hamiltonian:

(a) We denote the vertices of K_n by $v_1, v_2, \ldots, v_n$. To see that $G = K_n$ s Hamiltonian it is sufficient to consider the cycle $v_1 - v_2 - \cdots - v_n - v_1$.

(b) $G = N_1 + (K_4 \cup K_4)$ is not Hamiltonian, since by deleting the vertex of N_1 we obtain two connected components.

(c) $G = (K_n \cup K_m)^c = K_{n,m}$. If $n \neq m$, by the Theorem 50 (d) we conclude that G is not Hamiltonian.

In the case $m = n \leq 1$, G does not have cycles and, therefore, is not Hamiltonian. In the case $m = n \geq 2$, denoting the vertices of a copy of K_n by $v_1, v_2, \ldots, v_n$, and the ones of the another copy by $u_1, u_2, \ldots, u_n$ we see that a Hamiltonian cycle is $v_1 - u_1 - v_2 - u_2 \cdots - v_n - u_n - v_1$.

□

Exercise 119. Let $G = C_4 \square P_2$. Is $G \square P_2$ Eulerian? And the complement?

Solution: $G \square P_2$ is Eulerian, as it is 4-regular. The complement is 11-regular, therefore, it is not Eulerian. □

Exercise 120. Consider the graph $G = C_3 \square C_3$

(a) Draw G.

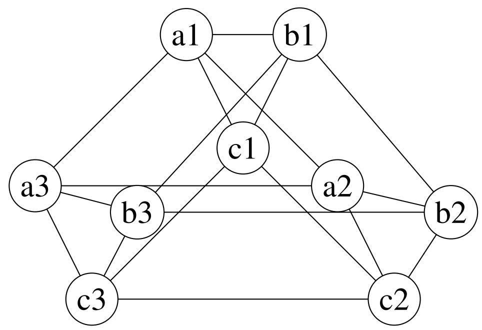
(b) Is the complement of G Eulerian?

(c) Without doing calculations, explain the relation that exists between the size of G and the size of its complement.

(d) Is the complement of G bipartite?

Solution:

(a) $G = C_3 \square C_3$ is



(b) The complement of G is Eulerian, as it is 4-regular.

(c) G and its complement have the same size since both are regular of degree 4.

(d) The complement of G is not bipartite as it has cycles of odd length. For example, the vertices $1b$, $2a$ and $3c$ form a cycle of length 3 in the complement of G.

□

Exercise 121. Let G be a Hamiltonian graph. Given the sequence of graphs $G_0 = G$, $G_1 = G\square K_2, \ldots,$ $G_r = G\square \underbrace{K_2 \square K_2 \square \cdots \square K_2}_{r}$. show that for every non-negative integer r, the graph G_r is Hamiltonian.

Solution: We will apply the induction method.

Base case: $G_0 = G$ is Hamiltonian.

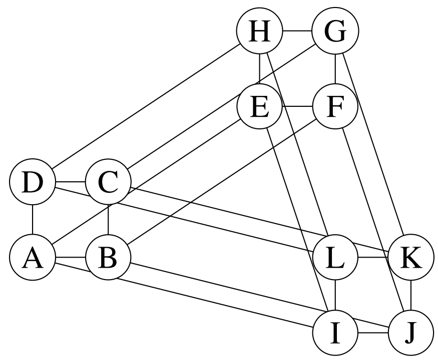
Hypothesis: we assume that G_r is Hamiltonian.

Let $v_1, v_2, \ldots, v_n, v_1$ be a Hamiltonian cycle of G_r and let a and b be the vertices of K_2. A Hamiltonian cycle of $G_r \square K_2$ is

$$(v_1,a),(v_2,a),\ldots,(v_n,a),(v_n,b),(v_{n-1},b),\ldots,(v_1,b),(v_1,a)$$

Therefore, $G_{r+1} = G_r \square K_2$ is also Hamiltonian. □

Exercise 122. Consider the graph G in the figure and answer in a reasoned way:



(a) Is the graph G Eulerian? In the affirmative case, build an Eulerian circuit with the help of the corresponding algorithm.

(b) Is G Hamiltonian?

(c) Express the graph G as a product of known graphs.

Solution:

(a) Yes, since every vertex has even degree. When applying the Hierholzer algorithm we obtain the following table:

Iteration	v	C'	C
0	A		$\{A\}$
	A	$\{A,B,C,D,A\}$	$\{A,B,C,D,A\}$
1	A	$\{A,E,F,G,H,D,L,H,E,I,A\}$	$\{A,E,F,G,H,D,L,H,E,I,A,B,C,D,A\}$
2	B	$\{B,F,J,B\}$	$\{A,E,F,G,H,D,L,H,E,I,A,B,$ $F,J,B,C,D,A\}$
3	C	$\{C,G,K,C\}$	$\{A,E,F,G,H,D,L,H,E,I,A,B,$ $F,J,B,C,G,K,C,D,A\}$
4	I	$\{I,J,K,L,I\}$	$\{A,E,F,G,H,D,L,H,E,I,J,K,L,$ $I,A,B,F,J,B,C,G,K,C,D,A\}$

Thus, the Eulerian circuit obtained is

$$C = \{A,E,F,G,H,D,L,H,E,I,J,K,L,I,A,B,F,J,B,C,G,K,C,D,A\}.$$

(b) G It is Hamiltonian. A Hamiltonian cycle is $A,B,C,G,F,E,I,J,K,L,H,D,A$.

(c) $G = C_4 \square C_3$.

$\square$

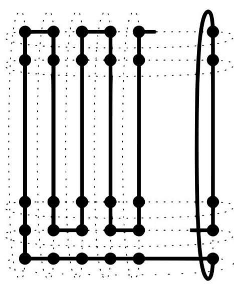
Exercise 123. Show that if G and H are Hamiltonian graphs, then $G \square H$ is Hamiltonian.

Solution: Let $n = |V(G)|$ and let C_n be a Hamiltonian cycle of G. Analogously, let $n' = |V(H)|$ and let $C_{n'}$ be Hamiltonian cycle of H. Since $C_n \Box C_{n'}$ is a spanning subgraph of $G \Box H$, we only need to observe that $C_n \Box C_{n'}$ is Hamiltonian.

We can label the vertices of G so that the cycle C_n is expressed by the sequence $u_1, \ldots, u_n, u_1$ of adjacent vertices. Now, we can label the vertices of H so that the cycle $C_{n'}$ is expressed by the sequence $v_1, \ldots, v_{n'}, v_1$ of adjacent vertices.

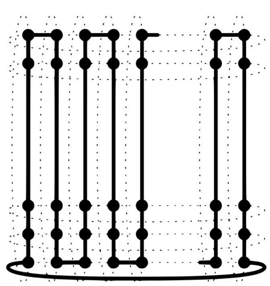
Thus, if n is odd, then a Hamiltonian cycle of $C_n \Box C_{n'}$ is obtained by concatenating the sequences

$$\begin{array}{l}
(u_1,v_1),(u_1,v_2),\ldots(u_1,v_{n'}),\\
(u_2,v_{n'}),(u_2,v_{n'-1}),\ldots,(u_2,v_2),\\
(u_3,v_2),(u_3,v_3),\ldots,(u_3,v_{n'}),\\
\quad\vdots\\
(u_n,v_2),(u_n,v_3),\ldots,(u_n,v_{n'}),(u_n,v_1),\\
(u_{n-1},v_1),\ldots,(u_1,v_1).
\end{array}$$



Analogously, if n is even, then a Hamiltonian cycle of $C_n \Box C_{n'}$ is obtained by concatenating the sequences

$$\begin{array}{l}
(u_1,v_1),(u_1,v_2),\ldots(u_1,v_{n'}),\\
(u_2,v_{n'}),(u_2,v_{n'-1}),\ldots,(u_2,v_1),\\
(u_3,v_1),(u_3,v_2),\ldots,(u_3,v_{n'}),\\
\quad\vdots\\
(u_n,v_{n'}),(u_n,v_{n'-1}),\ldots,(u_n,v_1),(u_1,v_1).
\end{array}$$



□

Chapter 7

Planar graphs

Planar graph theory traces back to the period of Euler, who in 1752 entered the known formula that relates the number m of edges of a convex polyhedron with the number f of faces and the number n of vertices: $m+2=f+n$. This subject remained practically without further development until 1930, when the Polish mathematician Kazimierz Kuratowski made a considerable advance while finding a characterisation of planar graphs. Precisely the formula of Euler and the theorem of Kuratowski are the main focus of this chapter in which, in addition, we analyse several examples and consequences of these two fundamental results.

7.1 Planar graphs and the Euler Formula

Definition 49. A *planar drawing* of a graph is a drawing in the plane without edge-crossings.

Definition 50. A graph is said to be *planar* if there there exists a planar drawing of it

In some engineering problems it is necessary to design interconnection networks so that the underlying graph is planar. To give a well-known example, the case of the integrated electronic circuits stands out. Another example of planar graphs are those designated fullerenes, which are the mathematical model of the family of chemical compounds called fullerenes.

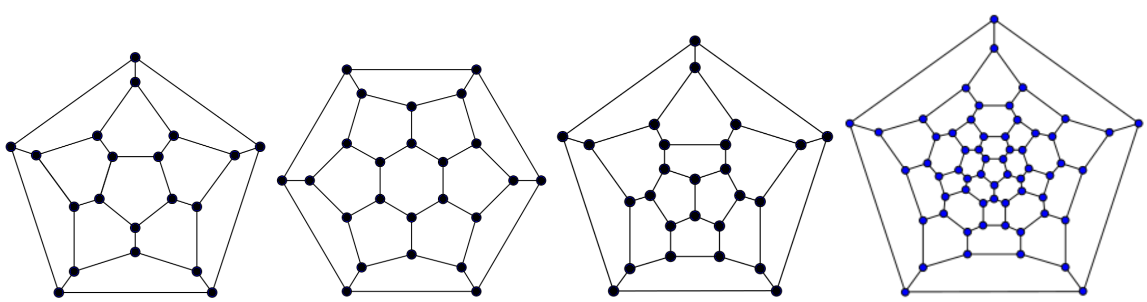


Figure 7.1: Examples of fullerenes

A particular well-known case of fullerenes is the one associate with the ball used in football/soccer (see Figure 7.2). This fullerene is known as a buckminsterfullerene and the formula

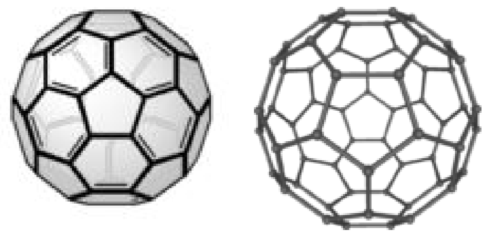


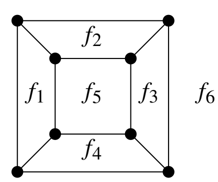
Figure 7.2: Buckminsterfullerene

of the chemical compound associated is C_{60}: it is a fullerene formed by 60 atoms of carbon, in which any of the pentagons that comprise it share an "edge". The structure of this fullerene is formed by 20 hexagons and 12 pentagons. The name of buckminsterfullerene comes from the architect Richard Buckminster Fuller due to the similarity of the molecule with one of his constructions.

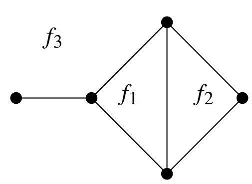
We will analyse below some of the mathematical properties of planar graphs. To this end, we will define some concepts.

Definition 51. The term *face* is used to describe each one of the regions of the plane determined by a planar drawing of a graph, likewise, *external face* describes the non-bounded region.

Example 72. The 3-cube $Q_3 = K_2 \square K_2 \square K_2$ is a planar graph with 6 faces. The degree[1] of each one of the faces is 4. A planar drawing of the cube is the following.



Example 73. The graph in the following figure has 3 faces: two interior faces of degree 3 and one exterior face with degree 5.



Example 74. All trees are planar and have a single, exterior, face.

[1]The degree of a face is the number of edges that delimit it.

Theorem 51. (*The Euler Formula*[2]) *For every planar drawing of a connected graph with n vertices and m edges, the number f of faces satisfies*

$$n + f = m + 2.$$

Proof. Induction with regard to m. If $m = 1$, then the graph is isomorph to the complete graph K_2. In this case $n = 2$ and $f = 1$, the result is true.

We assume that the result is true for any connected planar graph of size $m \geq 2$ arbitrary. That is, for G obeys $n + f = m + 2$.

There are two ways to obtain a connected planar graph G' of size $m+1$ from G. Case 1. We add an edge incident to a single vertex of G, and so it is necessary to add a new vertex to G. As G' has $n' = n+1$, $m' = m+1$ and $f' = f$, therefore, from the hypothesis ($n + f = m+2$) we obtain $n' + f' = m' + 2$.
Case 2. We add an edge incident to two vertices of G, and in this way a face is divided into two faces. Thus, in G' we have $n' = n$, $m' = m+1$ and $f' = f+1$. As above, from the hypothesis we obtain $n' + f' = m' + 2$. □

Example 75. The buckminsterfullerene, represented in Figure 7.2, has order $n = 60$ and $f = 32$ faces (20 hexagons y 12 pentagons); as such, according to the Euler Formula, the number of edges is $m = 90$.

Exercise 124. The degree sequence of a planar connected graph G is $6, 5, 5, 5, 4, 4, 4, 4, 3,$ $3, 3, 3, 3,$ $3, 3, 2, 2, 2, 2, 2$. Knowing that a face of G is a cycle of 14 vertices and that the other faces are triangles and squares, determine the number of triangles and squares of G.

Solution: The order of the graph is $n = 19$ and the size is

$$m = (1 \cdot 6 + 3 \cdot 5 + 4 \cdot 4 + 7 \cdot 3 + 4 \cdot 2)/2 = 33.$$

According to the Euler Formula ($m + 2 = n + f$) we can deduce that the graph has $f = 16$ faces. Let X_3 and X_4 be, respectively, the number of triangles and squares of the graph: $X_3 + X_4 + 1 = 16$. As each edge lies on exactly two faces, we have $3X_3 + 4X_4 + 14 = 2m = 66$. Thus, when resolving the system

$$\begin{aligned} X_3 + X_4 &= 15 \\ 3X_3 + 4X_4 &= 52, \end{aligned}$$

we obtain $X_3 = 8$ and $X_4 = 7$. □

Remark 52. *For the case of trees it is already known that the Euler Formula is expressed as* $m = n - 1$.

The following result, which we present here as a consequence of the previous theorem, in fact is a generalisation of the Euler Formula to the case of disconnected graphs.

[2]This formula was also discovered independently by Rene Descartes (1596-1650), and therefore on occasion is called the Euler-Descartes Formula. In addition, in topology it is known as the Euler-Poincare Formula.

Corollary 53. *For every planar drawing of a graph with n vertices, m edges and k components, the number f of faces satisfies*

$$n+f=m+k+1.$$

Proof. If G is a disconnected planar graph, the Euler Formula is not valid. If G has k components, then it is necessary to add $k-1$ new edges to connect them. In this way we obtain a connected planar graph G' that has the same number of faces as G. But when applying the Euler Formula to G' we have $n'+f'=m'+2$. As $c'=c$, $n'=n$ and $m'=m+k-1$, we obtain $n+f=m+k+1$. □

Another important consequence of the Euler Formula is the following necessary condition for the planarity of a graph.

Proposition 54. *For any planar graph of girt*[3] *g, order n, size m, and k components,*

$$m \leq \frac{g}{g-2}(n-k-1).$$

Proof. As each face has at least g edges and each edge lies on, at most, two faces, we have $fg \leq 2m$. From $f \leq \frac{2m}{g}$ and the Euler Formula for planar graphs of k components (Corollary 53) we derive the result. □

Corollary 55. *For any planar graph G of order $n \geq 3$,*

$$m \leq 3n-6.$$

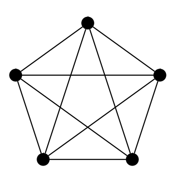
In addition, if G is free of triangles, then

$$m \leq 2n-4.$$

Proof. If G is a forest made up of k trees, then the number of edges is $m=n-k$ and $n-k \leq 2(n-2) \leq 3(n-2)$ for $n \geq 3$. Therefore, the result is true. If G is not a forest, then the result is deduced from Proposition 54. □

Below, we will see two important examples of applications of the previous result.

Example 76. Show that the complete graph K_5 is not planar.



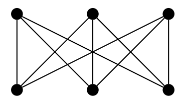
$G = K_5$

If K_5 was planar, it would have to be true that $m \leq 3n-6$, but we know that $m=\binom{5}{2}=10>9=3n-6$. Therefore, K_5 is not planar.

[3]The girth of a graph G is the minimum of the lengths of the cycles of G

Exercise 125. We suppose that in a small town there are three houses and three wells. All the neighbours from the houses have the right to use any of the three wells. As they don't get on well with each other, they never want to cross paths. Is it possible to find nine paths that connect the three houses with the three wells without crossing each other?

Solution: The problem can be reduced to verifying if a complete bipartite graph $K_{3,3}$ is planar.

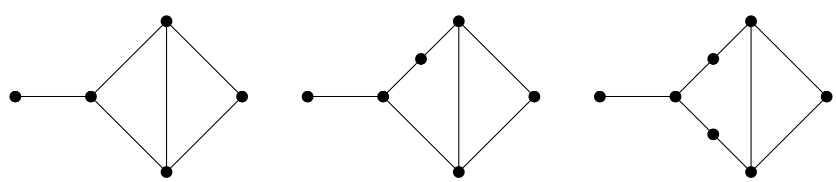


As $K_{3,3}$ is bipartite, it is free of triangles. Thus, if it was planar, it would have to obey $m \leq 2n-4$, but we know that $m = 9 > 8 = 2n-4$. Therefore, it is impossible to find the nine paths that join the three houses with the three wells without crossing. □

7.2 Characterisation of the planar graphs. Kuratowski's Theorem

Definition 52. An *elementary subdivision* of a simple graph G is produced when an edge $\{x,y\}$ of G is deleted, and afterwards the edges $\{x,v\}$ and $\{v,y\}$ are added to the graph $G - \{x,y\}$.

Example 77. In the following figure, the graph G_2 is obtained by means of an elementary subdivision of G_1, and G_3 is obtained by means of an elementary subdivision of G_2.



G_1 G_2 G_3

Definition 53. Two graphs G_1 and G_2 are *homeomorphic* if they obey any of the following conditions:

- They are isomorphic.
- Both can be obtained by means of a series of elementary subdivisions of the same graph.

Using the definition of homeomorphic graphs the Polish mathematician Kazimierz Kuratowski established in 1930 the following characterisation of planar graphs.

Theorem 56. (**Kuratowski's Theorem**) *A graph is planar if and only if it does not contain subgraphs which are homeomorphic to K_5 or $K_{3,3}$.*

We will omit the demonstration of the previous result that would occupy several pages. We refer those interested in a detailed demonstration to the book [5]. For examples of applications of this result we recommend the exercises 126, 131, 132 and 138.

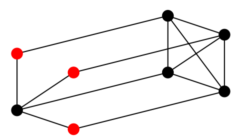
7.3 Exercises

Exercise 126. Study the planarity of the following graphs.

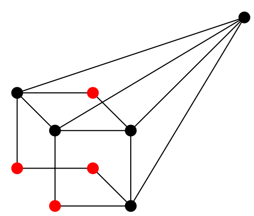
(a) $K_4 \square K_2$

(b) $Q_3 + K_1$

Solution: (a) The graph in the following figure is a subgraph of $K_4 \square K_2$ and it is a homeomorphic to K_5. Therefore, $K_4 \square K_2$ is not planar.



(b) The graph in the following figure is a subgraph of $Q_3 + K_1$ and it is homeomorphic to K_5. Therefore, $Q_3 + K_1$ is not planar.



$\square$

Exercise 127. Let G be a 3-regular connected planar graph of order 20. Determine in how many regions the plane will remain divided from any planar drawing of G. And if the graph has 3 components?

Solution: G has $m = \frac{3\times 20}{2} = 30$ edges. Applying the Euler Formula we can conclude that G has $f = m + 2 - n = 30 + 2 - 20 = 12$ faces. If the graph has 3 components, then according to Corollary 53 we have $f = m + k + 1 - n = 30 + 3 + 1 - 20 = 14$. $\square$

Exercise 128. Let G be a connected planar graph of order n and size m. Find the values of n and m if it is known that G has 6 faces, one vertex of degree 5, one vertex of degree 3, three vertices of degree 2, two vertices of degree 1, and the remaining vertices have degree 4.

Solution: According to the Euler Formula: $m + 2 = n + f$. It follows that $m = n + 4$. On the other hand, when applying the degree sum formula we obtain $m = 2n - 6$. Therefore, $n = 10$ and $m = 14$. $\square$

Exercise 129. Prove that any planar graph has at least one vertex of degree less than or equal to 5.

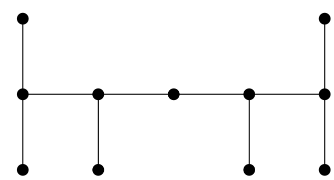
Solution: We suppose that every vertex has degree greater than or equal to 6. In this case the number of edges of the graph is bounded by $m \geq \frac{6n}{2} = 3n$. In addition, according to Corollary 55, $m \leq 3n - 6$, thus $0 \leq -6$, which is a contradiction. $\square$

Exercise 130. Prove that for any connected planar graph whose faces are cycles of length g,

$$m = \frac{g}{g-2}(n-2).$$

Solution: As each face has g edges and each edge edge lies on exactly two faces, we have $fg = 2m$. Hence, from $f = \frac{2m}{g}$ and the Euler Formula we derive the result. □

Exercise 131. Determine if the complement of the graph in the figure is planar.



Solution: The vertices of degree 1 will all be adjacent between them in the complement of G. Then, G^c contains a subgraph which is isomorph to K_6, by Kuratowski's Theorem we can conclude that G^c is not planar. □

Exercise 132. Study the planarity of the complement of $G = K_3 \square P_4$.

Solution: Let X be the set of vertices belonging to the extreme copies of K_3 (corresponding to the leaves of the path P_4). In G^c the subgraph induced by the vertices in X is isomorph to $K_{3,3}$. Thus, by Kuratowski's Theorem we can conclude that G^c is not planar. □

Exercise 133. Determine the values of k such that the hypercubes Q_k are planar graphs.

Solution: We know that $Q_1 \cong K_2$, $Q_2 \cong C_4$ and Q_3 (standard cube) are planar graphs. If Q_k was planar for $k \geq 4$, it would have to obey $m \leq 2(n-2)$, as the hypercubes are free of triangles. Since $n = 2^k$ and $m = k2^{k-1}$, we obtain $k2^{k-1} \leq 2(2^k - 2)$. But notice that $k2^{k-1} - 2^{k+2} \leq -4$ leads to $2^{k-1}(k-4) \leq -4$, and so $k < 4$. Therefore, for $k \geq 4$, the hypercubes are not planar. □

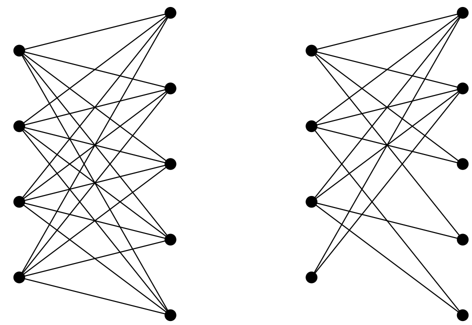
Exercise 134. Let G be a connected planar graph with an equal number of faces and vertices. There exist planar drawing of G where all the faces are triangles or squares so that the number of triangles and squares differs by one. Find the order of G.

Solution: By the Euler Formula we find that $m = 2n-2$. On the other hand, as each edge lies on exactly two faces, we have two possibilities $4\frac{n-1}{2} + 3\frac{n+1}{2} = 2m$ or $3\frac{n-1}{2} + 4\frac{n+1}{2} = 2m$. Therefore, $n = 9$ or $n = 7$. □

Exercise 135. Show that $K_{4,5}$ has a subgraph which is homeomorphic to K_5.

Solution: The graph on the left in the figure is $K_{4,5}$ and the graph on the right is a subgraph of $K_{4,5}$ which is homeomorphic to K_5. The vertices of degree 2 indicate the elementary subdivisions.

□



Exercise 136. Let G be a planar, 3-regular, connected graph with all the faces being 5-cycles or 6-cycles. Determine the number of 5-cycles.

Solution: The size of G is $m = \frac{3n}{2}$. By the Euler Formula, $m+2 = n+f$, we deduce that $f = \frac{1}{2}n+2$. Let f_r be the number of faces of G with size r. Then

$$f_5 + f_6 = f = \frac{1}{2}n + 2.$$

Since each edge lies in exactly two faces,

$$5f_5 + 6f_6 = 2m = 3n.$$

Solving these equations we find that $f_5 = 12$ and $f_6 = \frac{n-20}{2}$. □

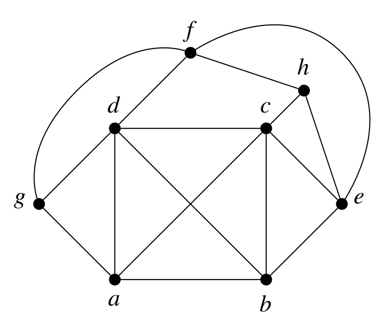
Exercise 137. The degree sequence of a connected planar graph G is $4,4,4,4,3,3,3,3,3,3,3,3$. Knowing that a face of G is a cycle of 8 vertices and that the other faces are triangles and squares, find the number of triangles and squares of G.

Solution: The size of the graph is $m = \frac{4\cdot 4+3\cdot 8}{2} = 20$. According to the Euler Formula we can deduce that the graph has $f = 10$ faces. Let X_3 and X_4 be, respectively, the number of triangles and squares of the graph. Then we have $X_3 + X_4 + 1 = 10$ and, as each edge lies on exactly two faces, we have $3X_3 + 4X_4 + 8 = 2m = 40$. From the system

$$\begin{aligned} X_3 + X_4 &= 9 \\ 3X_3 + 4X_4 &= 32, \end{aligned}$$

we have $X_3 = 4$ and $X_4 = 5$. □

Exercise 138. Determine if the line graph of the graph G represented in the following figure is planar.



Solution: As the graph G has vertices of degree 5, the graph $L(G)$ has subgraphs isomorphic to K_5. Therefore, by Kuratowski's Theorem we can conclude that $L(G)$ is not planar. $\square$

Exercise 139. Prove that if G is a planar graph of order $n > 10$, then its complement is a nonplanar graph.

Solution: For any planar graph G of order n,

$$m(G) \leq 3n - 6. \tag{7.1}$$

The size of G^c is

$$m(G^c) = \frac{n(n-1)}{2} - m(G). \tag{7.2}$$

Now, if G^c is also planar, then

$$\frac{n(n-1)}{2} - m(G) \leq 3n - 6. \tag{7.3}$$

Adding equations (7.1) and (7.3) we have

$$n(n-1) \leq 4(3n-6),$$

which implies that $n \leq 10$. Therefore, if $n \geq 11$, then G^c is not planar. $\square$

Exercise 140. Let G be a connected planar graph of order $n = 15$. If two faces of G are triangles and the remaining faces are squares, compute the number of edges of $G \square C_4$.

Solution: The size of $G \square C_4$ is $15 \cdot 4 + 4m$, where m is the size of G. Let x be the number of squares of G. By the Euler Formula ($m + 2 = n + f$) we deduce that $m - x = 15$. Since each edge is in the boundary of two faces, $6 + 4x = 2m$. Then, by solving the system

$$\begin{aligned} m - x &= 15 \\ m - 2x &= 3, \end{aligned}$$

we find $x = 12$ and $m = 27$. Therefore, the size of $G \square C_4$ is 168. $\square$

Exercise 141. Let G be a connected planar graph of order n and size m having 10 vertices of degree two and the remaining vertices of degree 3. If 8 faces of G are hexagons, 4 faces are pentagons and the remaining faces cycles of length 18, compute the values of n and m.

Solution: Since $2m = \sum \delta(v)$, we deduce that $2m = 2 \cdot 10 + 3(n - 10)$. Thus,

$$2m = 3n - 10. \tag{7.4}$$

On the other hand, since 8 faces of G are hexagons, 4 faces are pentagons and the remaining faces are cycles of length 18, we have that $2m = 6 \cdot 8 + 5 \cdot 4 + 18x$, where x denotes the number of cycles of length 18. Thus,

$$m = 34 + 9x. \tag{7.5}$$

Now, since the number of faces is $x + 12 = m + 2 - n$, equations (7.4) and (7.5) lead to $x = 1$, $m = 43$ and $n = 32$. $\square$

Exercise 142. Let G be a connected and planar graph of order n and size $m = 73$, which have six vertices of degree 5, eight vertices of degree 4, and the remaining vertices of degree $n-22$. If all faces of G are triangles or cycles of length 8, compute the number of triangles.

Solution: Since $2m = 6\cdot 5 + 8\cdot 4 + (n-14)(n-22)$, we deduce that $n = 28$. Now, since G is planar and connected, and all faces are cycles of length 8 or 3, we have that $3x_3 + 8x_8 = 2m$ and $m+2 = n + x_3 + x_8$, where x_3 denotes the number of triangles and x_8 denotes the number of cycles of length 8. Therefore, $3x_3 + 8x_8 = 146$ and $x_3 + x_8 = 47$, which implies that G has $x_3 = 46$ triangles. $\square$

Chapter 8

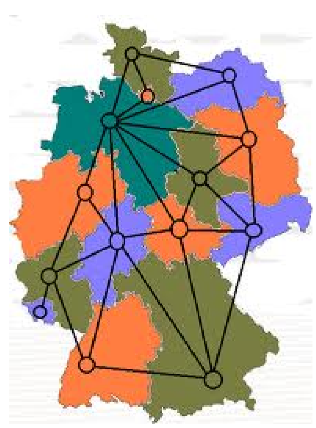
Vertex colouring

In this chapter, we give a brief introduction to the theory of vertex-colouring. In particular, we present some basic results on the chromatic number and we give some tools which are useful to compute the chromatic polynomial of a graph.

8.1 Introduction and basic results

The first problem of colouring of graphs that we know of was posed in London in 1852 by Francis Guthrie, a Ph.D student supervised by the mathematician Augustus of Morgan. Guthrie observed that it was possible to draw a map of England with four colours so that countries with a common border each had different colours. The question that Guthrie formulated to his supervisor was whether four colours would be sufficient to draw all the possible partitions of the map into regions so that neighbouring regions each had different colours. This problem was unresolved until 1976, when Wolfgang Haken and Kenneth Apple answered the question affirmatively with the demonstration of the famous four colour theorem.

We assume that we have the following "map" where the regions represent countries.



The problem of colouring the regions so that two countries with a common border have different colours can be approached as a problem of colouring the vertices of a graph so that adjacent vertices have different colours.

Definition 54. A *vertex-colouring* of a graph $G = (V,E)$ is a function $f : V \to \mathbb{N}$ with the property that $f(u) \neq f(v)$ whenever $\{u,v\} \in E$. Let $\mathcal{F}(G)$ be the set of vertex-colourings of G. The *chromatic number* of G is defined as

$$\chi(G) = \min_{f \in \mathcal{F}(G)} |Im(f)|,$$

Where $Im(f)$ denotes the image of f.

In other words the chromatic number is the minimum number of necessary colours to draw the vertices of a graph so that adjacent vertices have different colours. The following result is deduced directly from the definition of chromatic number.

Proposition 57. *Let G be a graph of order n. Then the following affirmations hold:*

(a) $\chi(G) = 1$ *If and only if G is empty.*

(b) $\chi(G) = 2$ *If and only if G is bipartite and not empty.*

(c) $\chi(K_n) = n$ *for every n.*

Corollary 58. *For all non-bipartite graphs of order $n \geq 2$,*

$$\chi(G) \geq 3.$$

Definition 55. Let G and H be two graphs. A map $f : V(G) \longrightarrow V(H)$ is a *homomorphism* from G to H if $f(x)$ and $f(y)$ are adjacent in H whenever x and y are adjacent in G.

Any isomorphism between graphs is a homomorphism. However, there are many homomorphisms that are not isomorphisms, as the following example illustrates.

Example 78. If $G = (V_1 \cup V_2, E)$ is a bipartite graph, then the map $f : V_1 \cup V_2 \longrightarrow V(K_2)$ that sends all the vertices of V_i to vertex i is a homomorphism from G to K_2.

Proposition 59. *The chromatic number of a graph G is the smallest integer r such that there exists a homomorphism from G to K_r.*

Proof. Suppose that f is a homomorphism from the graph G to a graph H. For every $y \in V(H)$ we define

$$f^{-1}(y) = \{x \in V(G) : \ f(x) = y\}.$$

Note that, if $f^{-1}(y) \neq \varnothing$, then the set $f^{-1}(y)$ is independent[1]. Hence, if there are r elements $y \in V(H)$ such that $f^{-1}(y) \neq \varnothing$, then these r sets induces a vertex-colouring of G with r colour classes, and so $\chi(G) \leq r$.

Now, if $\chi(G) = r$, since G can be coloured with r colours $\{1, \ldots, r\}$, the map that matches each vertex to its colour is a homomorphism from G to K_r. $\square$

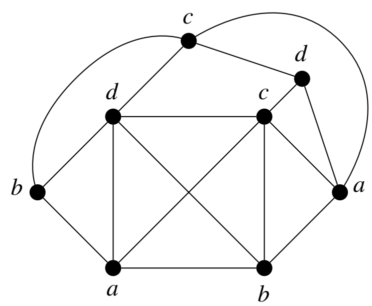
As all homomorphisms of graphs have to transform a complete subgraph into a complete subgraph of the same order, we obtain the following corollary.

[1] The subgraph induced by $f^{-1}(y)$ is empty.

Corollary 60. *If a graph G contains a subgraph which is isomorphic to K_r, then*

$$\chi(G) \geq r.$$

Example 79. The labelling of the vertices of the graph G in the figure shows that $\chi(G) \leq 4$. On the other hand, according to Corollary 60 we have that $\chi(G) \geq 4$. Therefore, $\chi(G) = 4$.



Proposition 61. *For all graph G of size m,*

$$m \geq \frac{\chi(G)(\chi(G)-1)}{2}.$$

Proof. For each pair of colours that comprise a vertex-colouring of minimum cardinality, there exists at least one edge of the graph; therefore, the number of edges is bounded below by $\binom{\chi(G)}{2} = \frac{\chi(G)(\chi(G)-1)}{2}$. □

A set $S \subseteq V(G)$ is said to be an independent set of G if the subgraph induced by S is empty. The independence number of G is defined to be

$$\alpha(G) = \max\{|S| : \ S \text{ is an independent set of } G\}.$$

Proposition 62. *For any graph G of order n,*

$$\chi(G) \geq \frac{n}{\alpha(G)}.$$

Proof. Let $f : V(G) \longrightarrow \{1, 2, \ldots, \chi(G)\}$ be a vertex-colouring of G. Let us denote the colour classes associated to f by

$$V_i = \{x \in V(G) : \ f(x) = i\} \text{ for every } i \in \{1, 2, \ldots, \chi(G)\}.$$

Since these colour classes are independent sets, and $V_1, \ldots V_{\chi(G)}$ is a partition of $V(G)$,

$$n = \sum_{i=1}^{\chi(G)} |V_i| \leq \sum_{i=1}^{\chi(G)} \alpha(G) = \chi(G)\alpha(G).$$

Therefore, the result follows. □

In some cases of composite graphs we can get the chromatic number of the resulting graph from the graphs that it originates from. We see, for example, the join of graphs. As in the join of graphs G_1+G_2 each vertex of G_1 is adjacent to all the vertices of G_2, then in all vertex-colourings of G_1+G_2 each vertex of G_2 has to be coloured in a different colour from all the colours used in the vertices of G_1. Therefore, it obeys the following result.

Proposition 63. *For all pairs of graphs G_1 and G_2,*

$$\chi(G_1+G_2)=\chi(G_1)+\chi(G_2).$$

Example 80. Can we affirm that $\chi(P_3+(C_3\square C_4))\geq 5$?

Solution: Yes, it is sufficient to observe that

$$\chi(P_3+(C_3\square C_4))=\chi(P_3)+\chi(C_3\square C_4)=2+\chi(C_3\square C_4).$$

And as $C_3\square C_4$ contains triangles, $\chi(C_3\square C_4)\geq 3$. □

Proposition 64. *For any two graphs G and H,*

$$\chi(G\square H)=\max\{\chi(G),\chi(H)\}.$$

Proof. Since $G\square H$ have subgraphs isomorphic to G and H,

$$\chi(G\square H)\geq\max\{\chi(G),\chi(H)\}.$$

It remains to show that $\chi(G\square H)\leq\max\{\chi(G),\chi(H)\}$. We may assume that $\chi(G)=k\geq\chi(H)$. Let $f_G:V(G)\longrightarrow\{1,2,\ldots,k\}$ and $f_H:V(H)\longrightarrow\{1,2,\ldots,\chi(H)\}$ be vertex-colourings. We construct a function $f:V(G)\times V(H)\longrightarrow\{1,2,\ldots,k\}$ defined by

$$f(g,h)=f_G(g)+f_H(h)\pmod{k}.$$

We proceed to show that f is a vertex-colouring of $G\square H$.

First, since $f_G(g_1)\neq f_G(g_2)$ for every $g_1g_2\in E(G)$, we have that for any $h\in V(H)$,

$$f_G(g_1)+f_H(h)\neq f_G(g_2)+f_H(h)\pmod{k}.$$

Thus, $f(g_1,h)\neq f(g_2,h)$. Now, since $\chi(H)\leq k$ and $f_H(h_1)\neq f_H(h_2)$ for every $h_1h_2\in E(H)$, we have that for any $g\in V(G)$,

$$f_G(g)+f_H(h_1)\neq f_G(g)+f_H(h_2)\pmod{k}.$$

Hence, $f(g,h_1)\neq f(g,h_2)$. Therefore, f is a vertex-colouring of $G\square H$, and so $\chi(G\square H)\leq k=\max\{\chi(G),\chi(H)\}$, which completes the proof. □

Exercise 143. Prove that for any non-empty graph G and any graph H,

$$2\chi(H)\leq\chi(G\circ H)\leq\chi(G)\chi(H).$$

Solution: Let $f_G : V(G) \longrightarrow \{1,2,\ldots,\chi(G)\}$ and $f_H : V(H) \longrightarrow \{1,2,\ldots,\chi(H)\}$ be vertex-colourings. We construct a function $f : V(G)\times V(H) \longrightarrow \{1,2,\ldots,\chi(G)\}\times\{1,2,\ldots,\chi(H)\}$ defined by

$$f(g,h) = (f_G(g), f_H(h)).$$

We proceed to show that f is a vertex-colouring of $G\circ H$. Let (g,h) and (g',h') be two adjacent vertices of $G\circ H$. If $g = g'$, then $h \sim h'$, and so $f_H(h) \neq f_H(h')$. Hence, $f(g,h) = (f_G(g), f_H(h)) \neq (f_G(g), f_H(h')) = f(g',h')$. Now, if $g \sim g'$, then $f_G(g) \neq f_G(G')$, which implies that $f(g,h) = (f_G(g), f_H(h)) \neq (f_G(g'), f_H(h')) = f(g',h')$. Therefore, the upper bound follows.

Let c be a vertex colouring of $G\circ H$. To prove the lower bound we only need to observe that for any $g\in V(G)$ the restriction of c to $\{g\}\times V(H)$ has at least $\chi(H)$ different colours, and for any pair $g,g'\in V(G)$ of adjacent vertices of G, no colours used in $\{g\}\times V(H)$ can be used in $\{g'\}\times V(H)$, and vice versa. Therefore, $\chi(G\circ H) \geq 2\chi(H)$. □

Corollary 65. *Let G be a non-empty graph. If G is bipartite, then for any graph H,*

$$\chi(G\circ H) = 2\chi(H).$$

Exercise 144. Find $\chi(C_7\circ H)$ for any graph H with $\chi(H) = 3$.

Solution: We already know that $\chi(C_7\circ H) \geq 2\chi(H) = 6$. Now if $\chi(C_7\circ H) = 6$, then any proper vertex-colouring of $C_7\circ H$ assigns a set of 3 colours for each copy of H in $C_7\circ H$, and for $u,u'\in N(v)$ the set of 3 colours assigned to $\{u\}\times V(H)$ is equal to the set of 3 colours assigned to $\{u'\}\times V(H)$, which is impossible, as C_7 has even order. Hence, $\chi(C_7\circ H) \geq 7$.

Now, if $\chi(H) = 3$, then it is easy to construct a proper vertex-colouring of $C_7\circ H$ with colours $0,1,\ldots,6$. Let $V(C_7) = \{u_0,\ldots,u_6\}$, where consecutive vertices are adjacent. The colours assigned to $\{u_i\}\times V(H)$ are $i, i+2$ and $i+4$, where the sum is taken modulo 7.

In summary, if $\chi(H) = 3$, then $\chi(C_7\circ H) = 7$. □

Theorem 66 (*The four colour Theorem: K. Appel y W. Haken, 1976*)**.** *For all planar graphs G,*

$$\chi(G) \leq 4.$$

We highlight some data that give an idea of the difficulty of the proof of the theorem of the four colours:

- 1852: Francis Guthrie posed the problem.
- 1879: Arthur Cayley published the conjecture [3].
- 1879: Sir Alfred Bray Kempe published his proof.
- 1890: Percy Heawood discovered an unresolvable error in the proof given by Kempe.
- 1976: Ken Appel and Wolfgang Haken demonstrated the theorem with help of a computer: 50 days of calculation, differentiating more than 1900 distinct configurations.
- 1996: Robertson, Sanders, Seymour and Thomas obtained a shorter proof [9].

8.2 Chromatic polynomial

Definition 56. Given a graph G, we will denote by $P_G(x)$ the number of vertex-colourings of G that use at most x colours. $P_G(x)$ is known by the name of *chromatic polynomial* of G.

The name "chromatic polynomial" is justified by the following proposition.

Proposition 67. *For any graph G of order n, $P_G(x)$ is a polynomial of degree n.*

Proof. For any vertex-colouring of G the nonempty colour classes constitute a partition of $V(G)$ where each part is an independent set. We may count those colourings that give a certain partition and add them up for all such partitions to find the total number of colourings. Since $V(G)$ is a finite set, it has a finite number of partitions, under which it is sufficient to show that the number of colourings for a single partition is a polynomial of x. Fix a partition with p parts, each of them being an independent set. By assigning a different colour to each part, we get all the colourings belonging to the partition. We may pick the first colour in x possible ways, the second in $x-1$ ways, etc. so there are $x(x-1)\cdots(x-p+1)$ colourings, which is obviously a polynomial. Notice that this also works when $x < p$.

Finally, there is no partition with more than n parts and only a single partition with exactly n parts. For this partition, the number of colourings is a polynomial of degree n while for all other partitions it has a degree less than n. The sum of this type of polynomials is one of degree n. □

Proposition 68. *The chromatic polynomial of all trees T of order n is*

$$P_T(x) = x(x-1)^{n-1}.$$

Proof. We will use the method of induction. The result is obvious for $n = 1$ and $n = 2$. We suppose that the chromatic polynomial of all trees of order $n-1$ is $x(x-1)^{n-2}$. Let v be a leaf of T and let u be the vertex of T adjacent to v. By hypothesis, the chromatic polynomial of $T' = T - \{v\}$ (tree that results when deleting v in T) is $P_{T'}(x) = x(x-1)^{n-2}$. In each vertex-colouring of T, any one of the colours which are different from the colour of u can be assigned to vertex v. Thus, vertex v can be coloured in $x-1$ ways. Therefore, $P_T(x) = (x-1)P_{T'}(x) = x(x-1)^{n-1}$. □

For all complete graphs of order n it is true that in all the vertex-colourings n colours are used and, therefore, $P_{K_n}(x) = V(x,n)$; is the number of variations of x elements taken of n in n. From here the following result can be deduced.

Proposition 69. *For all complete graphs of order n it is true that,*

$$P_{K_n}(x) = x(x-1)(x-2)\cdots(x-n+1).$$

On the other hand, if G it is not a complete graph, then the problem of colouring G can be reduced to colouring two graphs G' and G'' obtained from G in the following way. Let u and v be two non-adjacent vertices of G. Let G' be the graph obtained by adding to G the edge $\{u,v\}$, and let G'' be the simple graph obtained by identifying the vertices u and v. Thus, in G'' they are not the vertices u and v and in their place there is a new vertex uv that is adjacent to the neighbours of u and to the neighbours of v. In this way, the vertex-colourings of G where u and v have different colours are in a one to one correspondence with the vertex-colourings of G'. Similarly, the vertex-colourings of G where u and v have the same colour are in one to one correspondence with the vertex-colourings of G''. Therefore the following result is obtained:

Proposition 70. *For all graphs $G \ncong K_n$,*

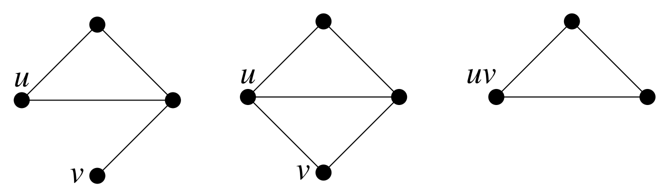
$$P_G(x) = P_{G'}(x) + P_{G''}(x).$$

By definition, $\chi(G)$ is the lower positive integer x that obeys $P_G(x) \geq 1$. Therefore the following consequence is obtained.

Corollary 71. *For all graph $G \ncong K_n$,*

$$\chi(G) = \min\{\chi(G'), \chi(G'')\}.$$

Example 81. For an example of application of the algorithm to calculate the chromatic polynomial consider the graph G in the following figure.



G G' G''

In this case we have $P_G(x) = P_{G'}(x) + P_{G''}(x)$, in addition, $P_{G'}(x) = P_{K_4}(x) + P_{K_3}(x)$. Therefore,

$$\begin{aligned} P_G(x) &= P_{G'}(x) + P_{G''}(x) \\ &= P_{K_4}(x) + 2P_{K_3}(x) \\ &= x(x-1)(x-2)(x-3) + 2x(x-1)(x-2) \\ &= x(x-1)^2(x-2). \end{aligned}$$

Although we have used the graph G in the previous figure to illustrate in a simple way the algorithm to determine the chromatic polynomial of a graph, in such a simple case like this we can calculate the polynomial directly. The vertex of degree 3 can be coloured in x different ways and, once coloured, there are $x-1$ possibilities for the vertex u and $x-1$ possibilities for the vertex of degree 1. Likewise, once coloured these three vertices have $x-2$ possibilities for colouring the another vertex of degree 2. Thus $P_G(x) = x(x-1)^2(x-2)$.

Proposition 72. *The chromatic polynomial of all cycles of order $n \geq 3$ is*

$$P_{C_n}(x) = (x-1)^n + (-1)^n(x-1).$$

Proof. We apply the method of induction. As $C_3 = K_3$, we have $P_{C_3}(x) = x(x-1)(x-2) = (x-1)^3 - (x-1)$. We assume that the result is true for $n-1$, with $n \geq 4$. Let $G' = C_n$ be the cycle obtained when joining with an edge the endpoints of the path P_n, and let $G'' = C_{n-1}$ be the cycle that results when identifying the endpoints of P_n. According to the Proposition 70 we have

$$P_{C_n}(x) = P_{P_n}(x) - P_{C_{n-1}}(x) = x(x-1)^{n-1} - P_{C_{n-1}}(x).$$

By hypothesis of induction we have $P_{C_{n-1}}(x) = (x-1)^{n-1} + (-1)^{n-1}(x-1)$, therefore

$$P_{C_n}(x) = x(x-1)^{n-1} - (x-1)^{n-1} - (-1)^{n-1}(x-1) = (x-1)^n + (-1)^n(x-1).$$

□

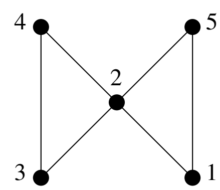
If a graph G contains a vertex u such that the subgraph induced by the open neighbourhood $N(u)$ is complete, then in all vertex-colourings of G, the colour of u is different from the $\delta(u)$ colours used to colour the neighbours of u. Thus, in all colouring of the neighbours of u, from x colours, there are $x - \delta(u)$ possibilities to choose the colour of u. Therefore, the following result follows.

Proposition 73. *If the subgraph induced by $N(u)$ is complete, then the chromatic polynomial of G is calculated as*

$$P_G(x) = P_{G-\{u\}}(x)(x - \delta(u)),$$

where $G - \{u\}$ is the subgraph of G obtains by deleting vertex u.

Example 82. Find the chromatic polynomial of the graph G in the figure.



We start from the complete graph of vertices $\{1,2,5\}$, whose chromatic polynomial is already known: $P_{K_3}(x) = x(x-1)(x-2)$, afterwards we consider the graph G_1 that results when we add vertex 3 to the triangle of the previous graph in such a way that 2 and 3 are adjacent in G_1. In this case

$$P_{G_1}(x) = P_{K_3}(x)(x-1) = x(x-1)^2(x-2).$$

Then, as the neighbours of 4 are adjacent to each other in G, we find that the chromatic polynomial of G is

$$P_G(x) = P_{G_1}(x)(x-2) = x(x-1)^2(x-2)^2.$$

Proposition 74. *Let $G = (V,E)$ be a graph and let r be a positive integer. If there exist two subgraphs of G, $G_1 = (V_1,E_1)$ and $G_2 = (V_2,E_2)$, such that $V = V_1 \cup V_2$, $\langle V_1 \cap V_2 \rangle \cong K_r$ and no edge of G connects $V_1 - (V_1 \cap V_2)$ with $V_2 - (V_1 \cap V_2)$, then*

$$P_G(x) = \frac{P_{G_1}(x)P_{G_2}(x)}{P_{K_r}(x)}.$$

Proof. As G contains $\langle V_1 \cap V_2 \rangle \cong K_r$ as a subgraph, G does not have colourings of vertices with less than r colours and therefore $P_{K_r}(x)$ is a factor of $P_G(x)$, i.e., there exists a polynomial $P^*(x)$ such that $P_G(x) = P^*(x)P_{K_r}(x)$. Thus, for all $x \geq r$, the number of ways to extend the colourings of $\langle V_1 \cap V_2 \rangle$ to the remaining vertices of the graph G, using at most x colours, is $P^*(x) = \frac{P_G(x)}{P_{K_r}(x)}$.

Analogously, since $P_{K_r}(x)$ is a factor of $P_{G_1}(x)$ and $P_{G_2}(x)$, there exist two polynomials $P_1(x)$ and $P_2(x)$ such that $P_{G_1}(x) = P_1(x)P_{K_r}(x)$ and $P_{G_2}(x) = P_2(x)P_{K_r}(x)$. Hence, for all $x \geq r$, the number of ways to extend the colourings of $\langle V_1 \cap V_2 \rangle$, using at most x colours, to the remaining vertices of the graph G_i is $P_i(x) = \frac{P_{G_i}(x)}{P_{K_r}(x)}$, $i \in \{1, 2\}$.

As there are no edges that connect vertices belonging to $V_1 - (V_1 \cap V_2)$ with vertices belonging to $V_2 - (V_1 \cap V_2)$, the extensions of the colourings of $\langle V_1 \cap V_2 \rangle$ to G_1 and G_2 are independent. Therefore, $P^*(x) = P_1(x)P_2(x)$. Therefore, $\frac{P_G(x)}{P_{K_r}(x)} = \frac{P_{G_1}(x)}{P_{K_r}(x)} \frac{P_{G_2}(x)}{P_{K_r}(x)}$. $\square$

Example 83. We will apply Proposition 74 to calculate the chromatic polynomial of the following graphs.

(a) $G = K_1 + P_3$.

(b) $G = K_1 + (K_2 \cup K_2)$.

In the first case we have $P_{K_1+P_3}(x) = \frac{P_{K_3}(x)P_{K_3}(x)}{P_{K_2}(x)} = x(x-1)(x-2)^2$, while in the second case $P_{K_1+(K_2 \cup K_2}(x) = \frac{P_{K_3}(x)P_{K_3}(x)}{P_{K_1}(x)} = x(x-1)^2(x-2)^2$.

Proposition 75. *For all graphs G and all non-negative integers r, the chromatic polynomial of the graph $K_r + G$ is*

$$P_{K_r+G}(x) = P_{K_r}(x)P_G(x-r).$$

Proof. We can colour the vertex of K_1 in x different ways and, once this is coloured, we have $x-1$ colours for colouring the vertices of G, thus $P_{K_1+G}(x) = xP_G(x-1)$. As $K_2 + G = K_1 + (K_1 + G)$, applying the previous reasoning we can deduce that $P_{K_2+G}(x) = xP_{K_1+G}(x-1) = x(x-1)P_G(x-2)$. Continuing with this process we find that $P_{K_r+G}(x) = x(x-1)\dots(x-r+1)P_G(x-r) = P_{K_r}(x)P_G(x-r)$. $\square$

In the disjoint union of graphs $G_1 \cup G_2$ all vertex-colourings of G_1 are independent of the vertex-colourings of G_2, and vice versa. Therefore, we can check the following proposition.

Proposition 76. *The chromatic polynomial of the disjoint union of two graphs G_1 and G_2 is*

$$P_{G_1 \cup G_2}(x) = P_{G_1}(x)P_{G_2}(x).$$

In the following example we will apply Propositions 75 and 76.

Example 84. The chromatic polynomial of $K_2 + (C_n \cup C_n)$ is

$$\begin{aligned} P_{K_2+(C_n \cup C_n)}(x) &= P_{K_2}(x)P_{C_n}(x-2)P_{C_n}(x-2) \\ &= x(x-1)\left((x-3)^n + (-1)^n(x-3)\right)^2. \end{aligned}$$

Proposition 77. *For any graph G of order n and any graph H,*

$$P_{G\odot H}(x) = P_G(x)\,(P_H(x-1))^n\,.$$

Proof. For any vertex $v \in V(G)$, let H_v be the copy of H associated to v in $G \odot H$. Let x be the number of colours. Now, for every vertex-colouring of the subgraph G of $G \odot H$, there are $x-1$ available colours to be used in the subgraph H_v, and so there are $P_{H_v}(x-1) = P_H(x-1)$ vertex-colouring of H_v with the $x-1$ colours. Hence,

$$P_{G\odot H}(x) = P_G(x) \prod_{v\in V(G)} P_{H_v}(x-1) = P_G(x)\,(P_H(x-1))^n\,.$$

$\square$

8.3 Exercises

Exercise 145. In a summer course we have programmed 9 talks, $v_1, v_2, \ldots, v_9$, of one hour each. The organising committee has sufficient classrooms to do simultaneous talks. When asking the participants to select the talks that interest them, they received the following applications $\{v_1, v_2, v_3\}$, $\{v_1, v_4, v_5\}$, $\{v_3, v_4, v_7\}$, $\{v_4, v_5, v_8\}$, $\{v_2, v_5, v_9\}$ and $\{v_2, v_3, v_6\}$.

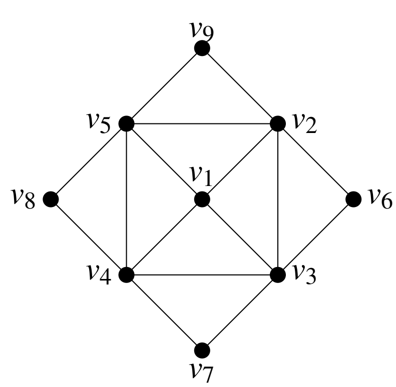
(a) Find the minimum number of hours necessary to distribute the talks so that all the participants can attend all the talks that interest them.

(b) Find the number of classrooms needed to give the talks in the minimum number of hours so that all participants can attend all the talks that interest them.

(c) We suppose now that we have x hours to do the talks and want to create a schedule of talks to deliver to the students. How many different schedules can it be done so that each student can attend all the talks that interest them?

Solution:

(a) We can represent this situation by means of the graph G in the figure.



□

The vertices correspond to the 9 talks and the edges to the possible conflicts. Then it attempts to look for the chromatic number so that G is $\chi(G) = 3$. Thus, all the talks can be given in three hours.

(b) Notice that in all the vertex-colourings of G with three colours the central vertex will have the same colour as the vertices of degree two. Therefore, 5 classrooms are needed.

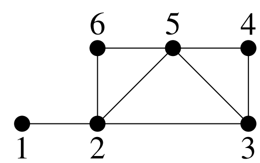
(c) In this case it attempts to calculate the chromatic polynomial of G, $P_G(x)$. Let G_1 be the graph that results when suppressing the vertices of degree 2 in G. As $G_1 \cong K_1 + C_4$, in accordance with Proposition 75 the chromatic polynomial of G_1 is

$$P_{G_1}(x) = x((x-2)^4 + (x-2)) = x(x-1)(x-2)(x^2 - 5x + 7).$$

On the other hand, in all vertex-colouring of G, the vertices of degree two can be coloured with $x-2$ colours. Therefore,

$$P_G(x) = P_{G_1}(x)(x-2)^4 = x(x-1)(x-2)^5(x^2 - 5x + 7).$$

Exercise 146. Consider the graph G in the following figure.



(a) Determine in how many ways the vertices of G can be coloured with the colours of set $\{a, b, c, d, e\}$ so that adjacent vertices have different colours.

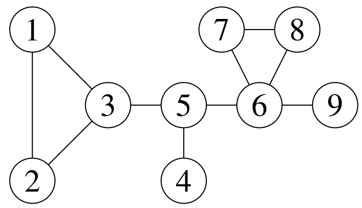
(b) Find the chromatic number of $G + Q_3$, where Q_3 denotes the standard cube.

Solution:

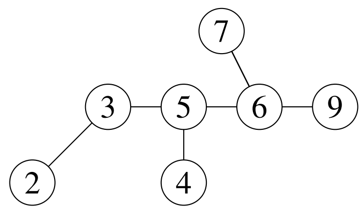
(a) The number of ways to label the vertices of G with labels of a set of cardinality x, so that adjacent vertices have different labels, coincides with the chromatic polynomial of G. To calculate said polynomial, start from the complete graph of vertices 2, 3, 5, whose polynomial is $x(x-1)(x-2$ and afterwards apply Proposition 73. Thus, $P_G(x) = x(x-1)^2(x-2)^3$. For a set of cardinality 5 we have $P(5) = 2160$.

(b) $\chi(G + Q_3) = \chi(G) + \chi(Q_3) = 5$. □

Exercise 147. Determine the chromatic polynomial of the graph in the figure.

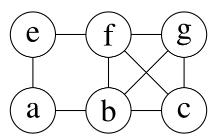


Solution: Let T be the tree of the figure:

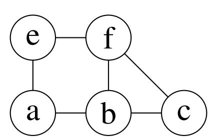


The chromatic polynomial of T is $P_T(x) = x(x-1)^6$ and the chromatic polynomial of G is $P_G(x) = P_T(x)(x-2)^2 = x(x-1)^6(x-2)^2$. □

Exercise 148. Determine the chromatic polynomial of the graph G in the figure.



Solution: Consider the graph of the figure:



G_1

The chromatic polynomial of G_1 is

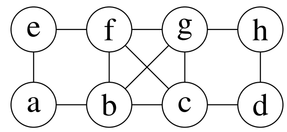
$$P_{G_1}(x) = P_{C_4}(x)(x-2) = x(x-1)(x^2-3x+3)(x-2).$$

Thus, the chromatic polynomial of G is

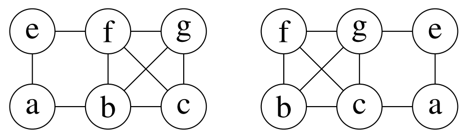
$$P_G(x) = P_{G_1}(x)(x-3) = x(x-1)(x-2)(x-3)(x^2-3x+3).$$

□

Exercise 149. Determine the chromatic polynomial of the graph G in the figure.



Solution: Consider the following subgraphs of G:



G_1 G_2

The chromatic polynomial of these graphs is

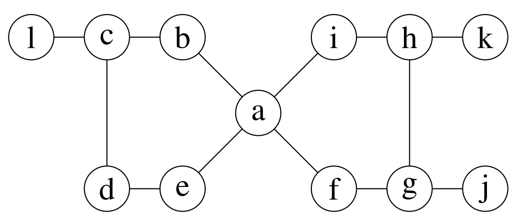
$$P_{G_1}(x) = P_{G_2}(x) = x(x-1)(x^2-3x+3)(x-2)(x-3).$$

Thus, the chromatic polynomial of G is

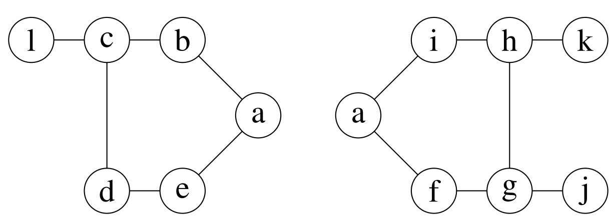
$$P_G(x) = \frac{P_{G_1}(x)P_{G_2}(x)}{P_{K_4}(x)} = x(x-1)(x-2)(x-3)(x^2-3x+3)^2.$$

□

Exercise 150. Determine the chromatic polynomial of the graph G in the figure.



Solution: Consider the following subgraphs of G:



G_1 G_2

The chromatic polynomial of these graphs are:

$$P_{G_1}(x) = P_{C_5}(x)(x-1) = x(x-1)^2(x-2)(x^2-2x+2),$$

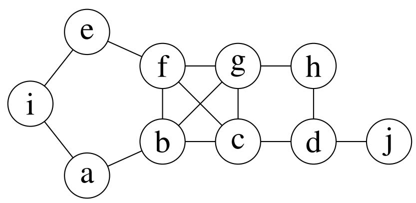
$$P_{G_2}(x) = P_{C_5}(x)(x-1)^2 = x(x-1)^3(x-2)(x^2-2x+2).$$

From here we have that the chromatic polynomial of G is

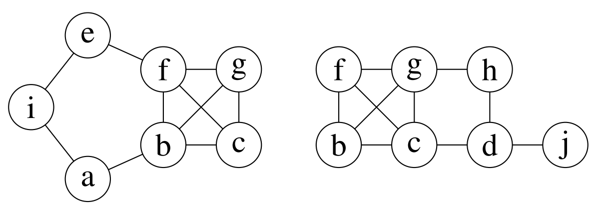
$$P_G(x) = \frac{P_{G_1}(x)P_{G_2}(x)}{P_{K_1}(x)} = x(x-1)^5(x-2)^2(x^2-2x+2)^2.$$

□

Exercise 151. Determine the chromatic polynomial of the graph G in the figure.



Solution: Consider the following subgraphs of G:



G_1 G_2

The chromatic polynomial of these graphs are:

$$P_{G_1}(x) = P_{C_5}(x)(x-2)(x-3) = x(x-1)(x-2)^2(x-3)(x^2-2x+2),$$

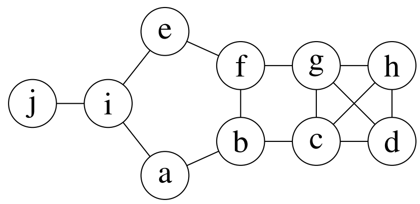
$$P_{G_2}(x) = P_{C_4}(x)(x-1)(x-2)(x-3) = x(x-1)^2(x-2)(x-3)(x^2-3x+3).$$

Thus, the chromatic polynomial of G is:

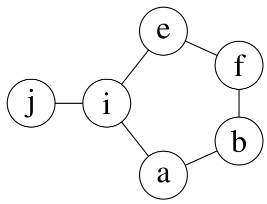
$$P_G(x) = \frac{P_{G_1}(x)P_{G_2}(x)}{P_{K_4}(x)} = x(x-1)^2(x-2)^2(x-3)(x^2-2x+2)(x^2-3x+3).$$

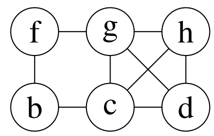
□

Exercise 152. Calculate the chromatic polynomial of the graph G in the figure.



Solution: Consider the following subgraphs of G:





G_1 G_2

The chromatic polynomial of these graphs is:

$$P_{G_1}(x) = P_{C_5}(x)(x-1) = x(x-1)^2(x-2)(x^2-2x+2),$$

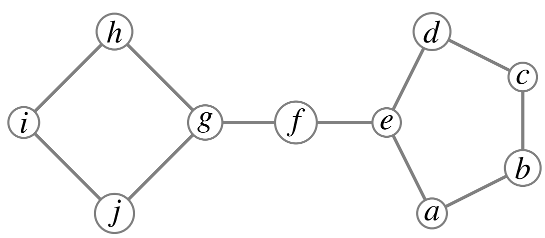
$$P_{G_2}(x) = P_{C_4}(x)(x-2)(x-3) = x(x-1)(x-2)(x-3)(x^2-3x+3).$$

Thus, the chromatic polynomial of G is

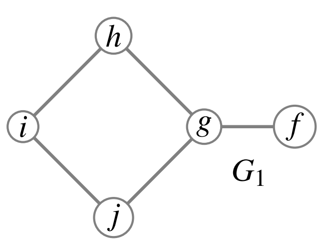
$$P_G(x) = \frac{P_{G_1}(x)P_{G_2}(x)}{P_{K_2}(x)} = x(x-1)^2(x-2)^2(x-3)(x^2-2x+2)(x^2-3x+3).$$

□

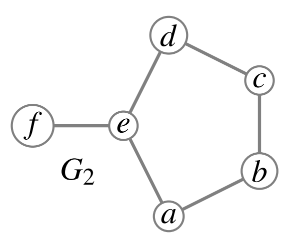
Exercise 153. Determine the chromatic polynomial of the following graph:



Solution: Consider the following subgraphs of G:



G_1



G_2

Then we have

$$P_{G_1}(x) = P_{C_4}(x)(x-1) = x(x-1)^2(x^2-3x+3),$$
$$P_{G_2}(x) = P_{C_5}(x)(x-1) = x(x-1)^2(x-2)(x^2-2x+2).$$

Thus, the chromatic polynomial of G is

$$P_G(x) = \frac{P_{G_1}(x)P_{G_2}(x)}{P_{K_1}(x)} = x(x-1)^4(x-2)(x^2-3x+3)(x^2-2x+2).$$

□

Exercise 154. Two graphs are chromatically equivalent if they have the same chromatic polynomial. A unicyclic graph is a connected graph containing exactly one cycle. Which unicyclic graphs of the same order are chromatically equivalent?

Solution: Let G be a unicyclic graph of order n having a cycle of length k. The chromatic polynomial of the cycle C_k is $P_{C_k}(x) = (x-1)^k + (-1)^k(x-1)$. Hence, the chromatic polynomial of G is

$$P_G(x) = P_{C_k}(x)(x-1)^{n-k} = (x-1)^n + (-1)^k(x-1)^{n-k+1}.$$

Therefore, two unicyclic graphs of order n are chromatically equivalent if and only if their cycles have the same length. □

Exercise 155. Determine the chromatic polynomial of $K_{2,4}$.

Solution: Notice that $K_{2,4} = N_2 + N_4$. Let x be the number of colours. We consider two cases; the vertex-colourations where the vertices of N_2 have the same colour, which are $x(x-1)^4$; and the vertex-colourations where the vertices of N_2 have different colour, which gives $x(x-1)(x-2)^4$. By the addition principle,

$$P_{K_{2,4}}(x) = x(x-1)^4 + x(x-1)(x-2)^4 = x(x-1)((x-1)^3 + (x-2)^4).$$

□

Exercise 156. Determine the chromatic polynomial of $N_2 + C_4$.

Solution: Let x be the number of colours. We consider two cases; the vertex-colourations where the vertices of N_2 have different colour, which are

$$x(x-1)P_{C_4}(x-2);$$

and the vertex-colourations where the vertices of N_2 have the same colour, which are

$$xP_{C_4}(x-1).$$

By the addition principle,

$$P_{N_2+C_4}(x) = x(x-1)P_{C_4}(x-2) + xP_{C_4}(x-1).$$

Now, the chromatic polynomial of C_4 is

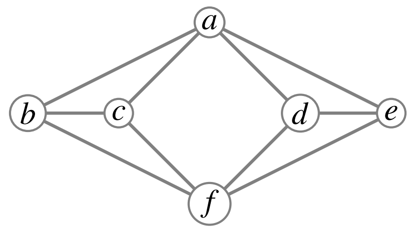
$$P_{C_4}(x) = x(x-1)(x^2-3x+3).$$

Therefore,

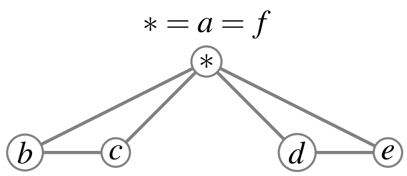
$$P_{N_2+C_4}(x) = x(x-1)(x-2)((x-3)((x-2)^2-3(x-2)+3) + (x-1)^2-3(x-1)+3).$$

□

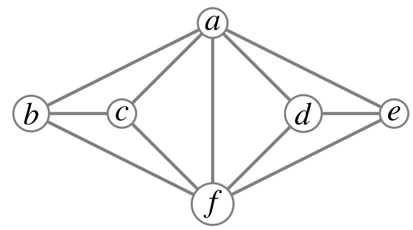
Exercise 157. Determine the chromatic polynomial of the following graph:



Solution: Consider the following graphs:



G_1



G_2

Then we have $P_G(x) = P_{G_1}(x) + P_{G_1}(x)$. Furthermore,

$$G_1 \cong K_1 + (K_2 \cup K_2) \text{ and } G_2 \cong K_1 + (K_1 + (K_2 \cup K_2)) = K_1 + G_1.$$

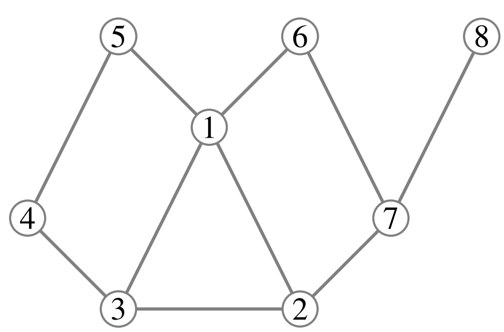
Hence,

$$P_{G_1}(x) = xP_{K_2 \cup K_2}(x-1) = x(x-1)^2(x-2)^2,$$
$$P_{G_2}(x) = xP_{G_1}(x-1) = x(x-1)(x-2)^2(x-3)^2.$$

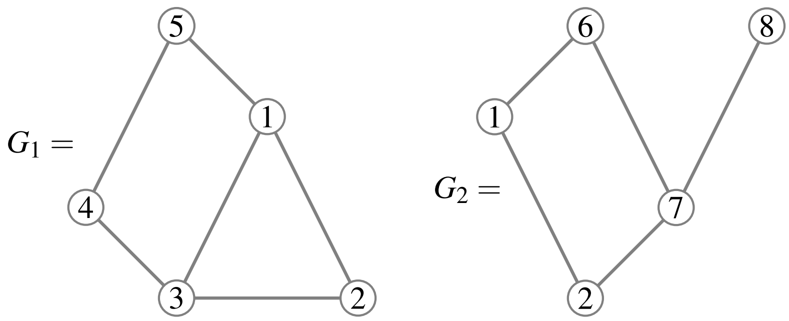
Therefore, the chromatic polynomial of G is

$$\begin{aligned} P_G(x) =& x(x-1)^2(x-2)^2 + x(x-1)(x-2)^2(x-3)^2 \\ =& x(x-1)(x-2)^2(x^2-5x+8) \\ =& x^6 - 10x^5 + 41x^4 - 84x^3 + 84x^2 - 32x. \quad \square \end{aligned}$$

Exercise 158. Determine the chromatic polynomial of the following graph.



Solution: The following graphs are subgraphs of G.



Hence,

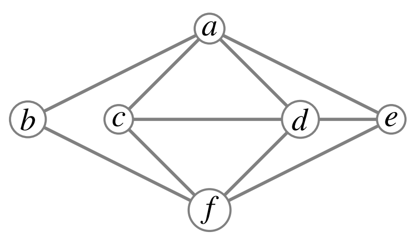
$$P_{G_1}(x) = (x-2)P_{C_4}(x) = x(x-1)(x-2)(x^2-3x+3).$$
$$P_{G_2}(x) = (x-1)P_{C_4}(x) = x(x-1)^2(x^2-3x+3).$$

Therefore

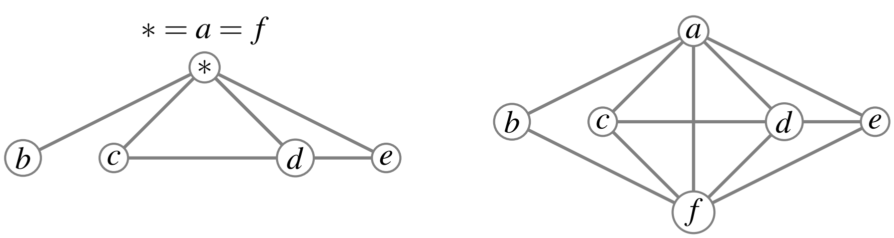
$$P_G(x) = \frac{P_{G_1}(x)P_{G_2}(x)}{P_{K_2}(x)} = x(x-1)^2(x-2)(x^2-3x+3)^2.$$

□

Exercise 159. Determine the chromatic polynomial of the following graph.



Solution: Consider the following graphs:



G_1 G_2

Then we have $P_G(x) = P_{G_1}(x) + P_{G_1}(x)$. Furthermore,

$$G_1 \cong K_1 + (K_1 \cup P_3) \text{ and } G_2 \cong K_1 + (K_1 + (K_1 \cup P_3)) = K_1 + G_1.$$

Hence,

$$P_{G_1}(x) = xP_{K_1 \cup P_3}(x-1) = x(x-1)^2(x-2)^2,$$
$$P_{G_2}(x) = xP_{G_1}(x-1) = x(x-1)(x-2)^2(x-3)^2.$$

Therefore, the chromatic polynomial of G is

$$\begin{aligned}
P_G(x) =& x(x-1)^2(x-2)^2 + x(x-1)(x-2)^2(x-3)^2 \\
=& x(x-1)(x-2)^2(x^2-5x+8) \\
=& x^6 - 10x^5 + 41x^4 - 84x^3 + 84x^2 - 32x.
\end{aligned}$$

□

Chapter 9

Characteristic sets and related invariants

9.1 Domination in graphs

Recall that the set of neighbours or *open neighbourhood* of a vertex v of a graph is denoted by $N(v)$. Now, the *closed neighbourhood* of v is defined to be $N[v] = N(v) \cup \{v\}$. Thus, the degree of v is given by $\delta(v) = |N(v)|$ and so $|N[v]| = \delta(v) + 1$.

Definition 57. A set $S \subseteq V$ is a *dominating set* of G if $N(v) \cap S \neq \varnothing$ for every vertex $v \in V \setminus S$.

The open neighbourhood of a set $S \subseteq V$ is given by

$$N(S) = \bigcup_{u \in S} N(s),$$

while he closed neighbourhood of $S \subseteq V$ is defined as

$$N[S] = N(S) \cup S = \bigcup_{u \in S} N[u].$$

Hence, $S \subseteq V$ is a dominating set of $G = (V, E)$ if and only if $N[S] = V$.

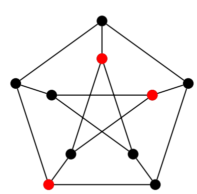
Definition 58. The *domination number* of a graph G is defined to be

$$\gamma(G) = \min\{|S| : \ S \text{ is a dominating set of } G\}.$$

A dominating set of cardinality $\gamma(G)$ will be called a $\gamma(G)$-set.

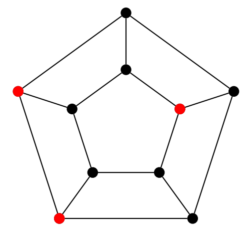
For instance, $\gamma(K_n) = 1$, $\gamma(Q_3) = 2$, $\gamma(C_5) = 2$ and $\gamma(K_{r,s}) = 2$.

Example 85. If G is the Petersen graph, then $\gamma(G) = 3$.



□

Example 86. $\gamma(C_5 \Box K_2) = 3$.



$\square$

We now proceed to show some basic results on domination in graphs.

Definition 59. A dominating set S is a *minimal dominating set* if for every vertex $v \in S$ the set $S \setminus \{v\}$ is not a dominating set.

Proposition 78. *A dominating set S of a graph $G = (V,E)$ is a minimal dominating set if and only if for each vertex $v \in S$, one of the following conditions holds:*

(a) *v is an isolated vertex of the subgraph induced by S,*

(b) *there exists a vertex $u \in V \setminus S$ such that $N(u) \cap S = \{v\}$.*

Proof. First, assume that S is a minimal dominating set of G. Thus, for every vertex $v \in S$ the set $S \setminus \{v\}$ is not a dominating set, and so there exists $u \in (V \setminus S) \cup \{v\}$ such that $N(u) \cap (S \setminus \{v\}) = \varnothing$. Hence, if $u = v$, then (a) holds, while if $u \neq v$, then $N(u) \cap S = \{v\}$ which means that (b) holds.

Conversely, assume that S is a dominating set and for for each $v \in S$, one of the two conditions holds. Suppose to the contrary that S is not a minimal dominating set. That is, there exists $v \in S$ such that $S_v = S \setminus \{v\}$ is a dominating set. In such a case, (a) does not hold for v, and for every $u \in V \setminus S$ we have that $N(u) \cap S_v \neq \varnothing$, which means that (b) does not hold for v. Therefore, neither (a) nor (b) are satisfied, which contradicts our assumption that at least one of these conditions holds. $\square$

Proposition 79. *Let $G = (V,E)$ be a graph of minimum degree $\delta_{\min}(G) \geq 1$. If $S \subseteq V$ is a minimal dominating set, then $V \setminus S$ is a dominating set.*

Proof. Let $v \in S$. By Proposition 78, either $N(v) \cap S = \varnothing$ or there exists a vertex $u \in V \setminus S$ such that $N(u) \cap S = \{v\}$. In the first case, since $\delta(v) \geq 1$, there exists a vertex $w \in V \setminus S$ such that $v \in N(w)$. In the second case, v is adjacent to $u \in V \setminus S$. Therefore, $V \setminus S$ is a dominating set of G. $\square$

Corollary 80. *For any graph G of order n and minimum degree $\delta_{\min}(G) \geq 1$,*

$$\gamma(G) \leq \left\lfloor \frac{n}{2} \right\rfloor.$$

Proof. Let S be a $\gamma(G)$-set. By Proposition 79, $V \setminus S$ is a dominating set. Hence, $\gamma(G) \leq |V \setminus S| = n - \gamma(G)$, which implies that $\gamma(G) \leq \frac{n}{2}$. Therefore, the result follows. $\square$

It is known that the equality $\gamma(G) = \frac{n}{2}$ holds is and only if the components of G are isomorphic to C_4 or to a corona graph of the form $H \odot K_1$ where H is a connected graph.

Exercise 160. Let G be a graph of order n. Prove that for any graph H,

$$\gamma(G \odot H) = n.$$

Solution: By definition of corona graph, $V(G)$ is a dominating set of $G \odot H$, which implies that $\gamma(G \odot H) \leq n$.

Now, for any $v \in V(G)$, the copy of H associated to v in $G \odot H$ will be denoted by H_v. Since for any $\gamma(G \odot H)$-set S we have $|N_{G\odot H}[S] \cap V(H_v)| \geq 1$ for every $v \in V(G)$, and $N_{G\odot H}(V(H_v)) \cap N_{G\odot H}(V(H_u)) = \varnothing$ for every pair $u, v \in V(G)$, we conclude that $\gamma(G \odot H) \geq n$. Therefore, the result follows. □

Exercise 161. Prove that for any graph G of order n and maximum degree $\Delta(G)$,

$$\left\lceil \frac{n}{1+\Delta(G)} \right\rceil \leq \gamma(G) \leq n - \Delta(G).$$

Solution: For any $\gamma(G)$-set S we have

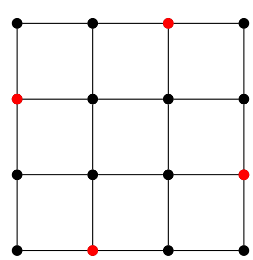
$$n \leq |S| + \sum_{v \in S} \delta(v) \leq |S| + |S|\Delta(G) = \gamma(G)(1+\Delta(G)).$$

Therefore, the lower bound follows.

In order to prove the upper bound, we only need to take a vertex $v \in V$ of maximum degree and observe that $V \setminus N(v)$ is a dominating set of G. Thus, $\gamma(G) \leq |V \setminus N(v)| = n - \Delta(G)$. □

The lower bound is achieved for the hypercube Q_3 and for complete graphs. The upper bound is achieved by the complete graph.

Example 87. Notice that $\gamma(P_4 \square P_4) \geq \lceil \frac{n}{1+\Delta} \rceil = \lceil \frac{16}{1+4} \rceil = 4$. Since the set of red vertices is a dominating set, $\gamma(P_4 \square P_4) \leq 4$. Therefore, $\gamma(P_4 \square P_4) = 4$.



□

Exercise 162. Let G be a graph of order n and minimum degree $\delta(G)$. Prove that if G has diameter two, then

$$\gamma(G) \leq \delta(G).$$

Solution: If v is a vertex of minimum degree and G has diameter two, then $N(v)$ is a dominating set. Therefore, $\gamma(G) \leq |N(v)| = \delta(G)$. □

Exercise 163. Show that if G is a graph of order n and minimum degree $\delta(G) \geq 1$, then

$$\gamma(G) \leq \left\lfloor \frac{n-\delta(G)+2}{2} \right\rfloor.$$

Solution: Let v be a vertex of minimum degree and let $X_v = \{u \in V : N(u) = N(v)\}$, the set of isolated vertices of the subgraph of G induced by $V \setminus N(v)$. Notice that if $V = X_v \cup N(v)$, then $\gamma(G) = 1$ or $\gamma(G) = 2$. In this case, if $\delta(G) = n-1$, then $G \cong K_n$ and so $\gamma(G) = 1 \leq \left\lfloor \frac{n-(n-1)+2}{2} \right\rfloor = \left\lfloor \frac{n-\delta(G)+2}{2} \right\rfloor$, while if $\delta(G) \leq n-2$, then $\gamma(G) \leq 2 = \left\lfloor \frac{n-(n-2)+2}{2} \right\rfloor \leq \left\lfloor \frac{n-\delta(G)+2}{2} \right\rfloor$. Therefore, in this case, the result follows.

From now on, we assume that $V \neq X_v \cup N(v)$. Let G' be the subgraph of G induced by $X_v \cup N(v)$ and let G'' be the subgraph of G induced by $V \setminus (X_v \cup N(v))$.

Since G'' does not have isolated vertices, Corollary 80 leads to

$$2\gamma(G'') \leq n - \delta(G) - |X_v|.$$

On the other side,

$$\gamma(G) \leq \gamma(G') + \gamma(G'').$$

Hence,

$$2\gamma(G) \leq 2\gamma(G') + n - \delta(G) - |X_v|.$$

Now, if $|X_v| = 1$, then $\gamma(G') = 1$, while $|X_v| \geq 2$, then we have $\gamma(G') \leq 2$. In both cases we obtain $2\gamma(G') - |X_v| \leq 2$, which implies the desired inequality

$$2\gamma(G) \leq n - \delta(G) + 2.$$

Therefore, the result follows. □

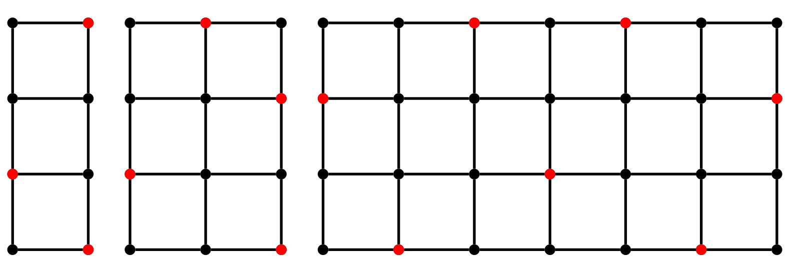
Exercise 164. Show that for any graph G,

$$\gamma(G) \leq \chi(G^c).$$

Solution: Let $\Pi = \{V_1, V_2, \ldots, V_{\chi(G^c)}\}$ be a partition of $V(G^c) = V(G)$ associated to a vertex-colouring of G^c. Since for every set $V_i \in \Pi$ the subgraph of G^c induced by V_i is empty, the subgraph of G induced by V_i is complete. Hence, we can form a dominating set S of G, of cardinality $\chi(G)$, by chosen one vertex of each set $V_i \in \Pi$. Therefore, $\gamma(G) \leq |S| = |\Pi| = \chi(G^c)$. □

Exercise 165. Find the domination number of the graphs $P_k \square P_4$ for $k \in \{2,3,7\}$.

Solution: We proceed to show that $\gamma(P_2 \square P_4) = 3$, $\gamma(P_3 \square P_4) = 4$ and $\gamma(P_7 \square P_4) = 7$. In order to see that these values are upper bounds, we can consider the dominating sets defined from the red-coloured vertices.



It remains to show that these values are lower bounds.

Case $k = 2$. Suppose that $S = \{x, y\}$ is a dominating set of $G = P_2 \Box P_4$. In such a case,

$$8 = n(P_2 \Box P_4) = |N_G[x] \cup N_G[y]| = |N_G[x]| + |N_G[y]| - |N_G[x] \cap N[y]|.$$

But this equality is only satisfied if x and y have degree 3 and $N_G[x] \cap N[y] = \varnothing$. Since there are no two vertices with such properties, $\gamma(P_2 \Box P_4) \geq 3$.

In order to discuss the remaining cases, we proceed to introduce the following notation. The vertex set of P_n is $V(P_n) = \{1, \ldots, n\}$. Let S be a $\gamma(P_k \Box P_4)$-set, let H_x be the subgraph of $P_k \Box P_4$ isomorphic to P_4, which is associated to $x \in V(P_k)$, and let $S_x = S \cap V(H_x)$.

Case $k = 3$. If $S_i = \varnothing$ for some $i \in \{1, 2, 3\}$, then $\gamma(P_3 \Box P_4) = |S| \geq \sum_{j \in N_G(i)} |S_j| \geq 4$, as required. Now, assume $S_i \neq \varnothing$ for every i. Suppose that $S = \{w_1, w_2, w_3\}$, where $S_i = \{w_i\}$ for every $i \in \{1, 2, 3\}$. If $w_2 = (2, 1)$, then w_1 has to be $(1, 3)$ and w_3 has to be $(2, 3)$, as the vertices of H_1 and H_3 have to be dominated by S. But, in such a case, $(2, 4)$ is not dominated by S. The case $w_2 = (2, 4)$ is analogous to the previous one, while the cases $w_2 = (2, 2)$ and $w_2 = (2, 3)$ are impossible, as there would be a vertex of H_1 (either $(1, 1)$ or $(1, 4)$) that would not be dominated by S. Therefore, $\gamma(P_3 \Box P_4) \geq 4$.

Case $k = 7$. Let $G_1 \cong P_2 \Box P_4$ be the subgraph of $P_7 \Box P_4$ induced by $\{1, 2\} \times V(P_4)$ and let $G_2 \cong P_3 \Box P_4$ be the subgraph of $P_7 \Box P_4$ induced by $\{5, 6, 7\} \times V(P_4)$. Let S_3' be the projection of S_3 on H_2 and let S_5' be the projection of S_4 on H_5. Since $S_1 \cup S_2 \cup S_3'$ is a dominating set of G_1 and $S_4' \cup S_5 \cup S_6 \cup S_7$ is a dominating set of G_2,

$$\begin{aligned}
\gamma(P_7 \Box P_4) &= \sum_{i=1}^{7} |S_i| \\
&\geq |S_1 \cup S_2 \cup S_3'| + |S_4' \cup S_5 \cup S_6 \cup S_7| \\
&\geq \gamma(G_1) + \gamma(G_2) \\
&= 7.
\end{aligned}$$

□

9.1.1 The case of paths and cycles

If H is a spanning subgraph of a graph G, then any dominating set of H is a dominating set of G. Therefore, the following statement holds.

Remark 81. *If H is a spanning subgraph of a graph G, then*

$$\gamma(H) \geq \gamma(G).$$

In particular $\gamma(P_n) \geq \gamma(C_n)$ for every integer $n \geq 3$.

Exercise 166. Prove that for any integer $n \geq 3$,

$$\gamma(C_n) = \gamma(P_n) = \left\lceil \frac{n}{3} \right\rceil.$$

Solution: Since $\gamma(C_n) \leq \gamma(P_n)$, we only need to show that $\gamma(C_n) \geq \left\lceil \frac{n}{3} \right\rceil$ and also $\gamma(P_n) \leq \left\lceil \frac{n}{3} \right\rceil$.

First, by Exercise 161, $\gamma(C_n) \geq \left\lceil \frac{n}{1+\Delta(C_n)} \right\rceil = \left\lceil \frac{n}{3} \right\rceil$.

Now, let $V(P_n) = \{u_1, \ldots, u_n\}$. We proceed to construct a dominating set D of P_n of cardinality $|D| = \left\lceil \frac{n}{3} \right\rceil$.

- If $n \equiv 0 \pmod 3$, then $D = \{u_2, u_5, \ldots, u_{n-1}\}$. In this case, $|D| = \frac{n}{3} = \left\lceil \frac{n}{3} \right\rceil$.
- If $n \equiv 1 \pmod 3$, then $D = \{u_2, u_5, \ldots, u_{n-2}, u_n\}$. Now, $|D| = \frac{n-1}{3} + 1 = \left\lceil \frac{n}{3} \right\rceil$.
- If $n \equiv 2 \pmod 3$, then $D = \{u_2, u_5, \ldots, u_n\}$. In this case, $|D| = \frac{n-2}{3} + 1 = \left\lceil \frac{n}{3} \right\rceil$.

By simple inspection, we can verify that D is a dominating set of P_n, which implies that $\gamma(P_n) \leq \left\lceil \frac{n}{3} \right\rceil$. Therefore, the result follows from

$$\left\lceil \frac{n}{3} \right\rceil \leq \gamma(C_n) \leq \gamma(P_n) \leq \left\lceil \frac{n}{3} \right\rceil.$$

□

9.1.2 Domination number versus diameter and girth

Exercise 167. Prove that for any graph G of diameter $D(G)$,

$$\gamma(G) \geq \left\lceil \frac{D(G)+1}{3} \right\rceil.$$

Solution: Let P be a diametral path of G and let S be a $\gamma(G)$-set. For every $v \in S$ the subgraph induced by $N[v]$ contains at most two edges of P. Now, since S is a $\gamma(G)$-set, there are at most $\gamma(G) - 1$ edges of P connecting the close neighbourhoods of vertices in S. Hence,

$$D(G) = l(P) \leq 2\gamma(G) + (\gamma(G) - 1) = 3\gamma(G) - 1.$$

Therefore, the result follows. □

Exercise 168. Prove that for any graph G of diameter $D(G) \geq 3$,

$$\gamma(G^c) = 2.$$

Solution: Let $x, y \in V$ be two diametral vertices. Since $D(G) \geq 3$, we have that $\{x, y\}$ is a dominating set of G^c, and so $\gamma(G^c) \leq 2$. Now, since G does not have isolated vertices, $\gamma(G^c) \geq 2$. Therefore, $\gamma(G^c) = 2$. □

Recall that the girth of a graph G is the minimum order among all cycles in G.

Exercise 169. Prove that if G is a graph of minimum degree $\delta(G) \geq 2$ and girth $g(G) \geq 5$, then

$$\gamma(G) \leq \left\lfloor \frac{n - g(G)}{2} \right\rfloor + \left\lceil \frac{g(G)}{3} \right\rceil.$$

Solution: If G is a cycle, then we are done. Assume that G is not a cycle graph. Let C be a cycle of G of length $g(G)$, and let G' be the subgraph of G induced by $V(G)\setminus V(C)$. Since $g(G)\geq 5$, no vertex of G' has two neighbors in C. Thus, since $\delta(G)\geq 2$, we have that $\delta(G')\geq 1$. Hence, by Corollary 80, $\gamma(G')\leq \left\lfloor \frac{n-g(G)}{2}\right\rfloor$. Therefore,

$$\gamma(G)\leq\gamma(G')+\gamma(C)\leq\left\lfloor\frac{n-g(G)}{2}\right\rfloor+\left\lceil\frac{g(G)}{3}\right\rceil.$$

□

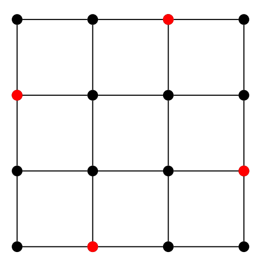
9.1.3 Domination number versus 2-packing number

Definition 60. Let $G=(V,E)$ be a graph. A set $X\subseteq V$ is a 2-*packing* of G if $N[u]\cap N[v]=\varnothing$ for every $u,v\in X$ with $u\neq v$. The 2-*packing number* is defined to be

$$\rho(G)=\max\{|X|:\ X \text{ is a 2-packing of } G.\}$$

A 2-packing of cardinality $\rho(G)$ is called a $\rho(G)$-set.

Example 88. The set of vertices in red form a $\rho(P_4\Box P_4)$-set. Observe that $\gamma(P_4\Box P_4)=\rho(P_4\Box P_4)=4$.



□

Exercise 170. Prove that for any graph G,

$$\gamma(G)\geq\rho(G).$$

Solution: Let S be a $\gamma(G)$-set and let X be a $\rho(G)$-set. Since $|S\cap N[v]|\geq 1$ for every $v\in X$, and $N[u]\cap N[v]=\varnothing$ for every $u,v\in X$, we have

$$\gamma(G)=|S|\geq\sum_{v\in X}|S\cap N[v]|\geq|X|=\rho(G).$$

□

Exercise 171. Let T be a tree. Use the induction method on the order of T to prove that $\rho(T)\geq\gamma(T)$.

Solution: If $n(T) \leq 3$, then it is easy to check that $\rho(T) \geq \gamma(T)$. These particular cases establish the base cases.

From now on, we consider that T has order at least four. Our induction hypothesis is that $\rho(T') \geq \gamma(T')$ for every tree T' with $n(T') < n(T)$.

Now, let $x_0, x_1, \ldots, x_k$ be a sequence of vertices of T forming a diametral path. Obviously, x_1 and x_k are leaves of T. We next analyse the following three cases.

Case 1: $|N(x_1)| \geq 3$. Since $x_0, x_1, \ldots, x_k$ form a diametral path of T, we have that $N(x_1) \setminus \{x_2\}$ is a subset of leaves of T. Hence, for $T' = T - \{x_0\}$ we have that $\rho(T) = \rho(T')$ and $\gamma(T) = \gamma(T')$. From the previous equalities and the induction hypothesis it follows that $\rho(T) = \rho(T') \geq \gamma(T') = \gamma(T)$, as desired.

Case 2: $N(x_1) = \{x_0, x_2\}$ and $|N(x_2)| \geq 3$. In this case, we take $T' = T - \{x_0, x_1\}$. Let S' be a $\rho(T')$-set such that $x_2 \notin S'$. Notice that $S' \cup \{x_0\}$ is a packing of T. Hence, $\rho(T) \geq \rho(T') + 1$. Moreover, if D' is a $\gamma(T')$-set, then $D' \cup \{x_1\}$ is a dominating set of T. Thus, $\gamma(T) \leq \gamma(T') + 1$. From the previous inequalities and the induction hypothesis it follows that $\rho(T) \geq \rho(T') + 1 \geq \gamma(T') + 1 \geq \gamma(T)$, as desired.

Case 3: $|N(x_1)| = |N(x_2)| = 2$. Now, we take $T' = T - \{x_0, x_1, x_2\}$. If S' is a $\rho(T')$-set, then $S' \cup \{x_0\}$ is a 2-packing of T, which implies that $\rho(T) \geq |S'| + 1 = \rho(T') + 1$. Now, if D' is a $\gamma(T')$-set, then $D' \cup \{x_1\}$ is a dominating set of T, and so $\gamma(T) \leq |D'| + 1 = \gamma(T') + 1$. From the previous inequalities and the induction hypothesis it follows that $\rho(T) \geq \rho(T') + 1 \geq \gamma(T') + 1 \geq \gamma(T)$, as desired.

According to the three cases above, the proof is complete. □

Exercise 172. Give an alternative proof of $\rho(T) \geq \gamma(T)$ for every tree T.

Solution: It is readily seen that if T has diameter at most two, then $\gamma(T) = \rho(T) = 1$. From now on we assume that T has diameter at least three. Let $x_0, x_1, \ldots, x_k$ be a sequence of vertices of T forming a diametral path. Obviously, x_1 and x_k are leaves and $k \geq 3$. Now, let T_1 be the subtree of minimum order which contains all vertices z of T such that $d(z, x_1) > 1$.

Notice that either T_1 has diameter at most two or we can continue the same process recursively from T_1 until we obtain a subtree T_r of diameter at most two. Hence, in order to conclude that $\gamma(T) \leq \rho(T)$, our hypothesis is that $\gamma(T_1) \leq \rho(T_1)$.

Let P be a $\rho(T)$-set, P' a $\rho(T_1)$-set, S a $\gamma(T)$-set and S' a $\gamma(T_1)$-set. Since $S' \cup \{x_1\}$ is a dominating set of T, we can claim that

$$\gamma(T) \leq \gamma(T_1) + 1.$$

Now, we differentiate two cases for x_2.

Case 1:$x_2 \notin P'$. In this case, $P' \cup \{x_0\}$ is a 2-packing of T, and so $\rho(T) \geq \rho(T_1) + 1$. Therefore,

$$\rho(T) \geq \rho(T_1) + 1 \geq \gamma(T_1) + 1 \geq \gamma(T).$$

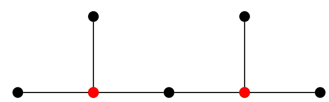
Case 2: $x_2 \in P'$. By definition of T_1, if x_2 has degree 2 in T, then $x_2 \notin V(T_1)$, which is a contradiction. Hence, there exists a vertex y of T_1 which is adjacent to x_2 and different from x_3. Since $x_0, x_1, \ldots, x_k$ is a diametral path in T, either y is a leaf or $N(y) \setminus \{x_2\}$ is a set of leaves.

Furthermore, since $x_2 \in P'$, we can claim that $P' \cap N[y] = \{x_2\}$, and so $Y = \{y\} \cup (P' \setminus \{x_2\})$ is a $\rho(T_1)$-set and $Y \cup \{x_0\}$ is a 2-packing of T. Thus, $\rho(T) \geq |Y| + 1 = |P'| + 1 = \rho(T_1) + 1$, and as in Case 1 we conclude that $\rho(G) \geq \gamma(T)$. Therefore, the proof is complete. □

From Exercises 170 and 171 we deduce the following result.

Proposition 82. $\gamma(T) = \rho(T)$ *for every tree* T.

Remark 83. *Although* $\gamma(T) = \rho(T)$ *for every tree* T*, there are cases where no* $\gamma(T)$*-set is a* $\rho(T)$*-set. For instance, in the next tree, the only* $\gamma(T)$*-set is formed by the red coloured vertices, but this set is not a* 2*-packing.*



□

9.1.4 Domination in Cartesian product graphs

The most famous open problem involving Cartesian product graphs on the topic of domination is known as Vizing's conjecture [10], which is an open problem stated by Vadim G. Vizing in 1963.

Conjecture 1 (Vizing's conjecture, open problem since 1963)**.** For any graphs G and H,

$$\gamma(G\Box H) \geq \gamma(G)\gamma(H).$$

For partial results on Vizing's conjecture, see [2, 6]. Obviously, young people with some curiosity for graph theory are kindly invited to try to solve the conjecture.

Next we consider a straightforward bound for Cartesian product graphs.

Exercise 173. Prove that for every pair of graphs G and H,

$$\gamma(G\Box H) \leq \min\{n(H)\gamma(G), n(G)\gamma(H)\}.$$

Solution: Let $S \subseteq V(G)$ be a $\gamma(G)$-set. For every $g \in V(G) \setminus S$ there exists $g \in S$ such that $g' \in N_G(g)$. Hence, for every $(g', h) \in (V(G) \setminus S) \times V(H)$, there exists $(g, h) \in S \times V(H)$ such that $(g', h) \in N_{G\Box H}(g, h)$, which implies that $S \times V(H)$ is a dominating set of $G\Box H$. Therefore,

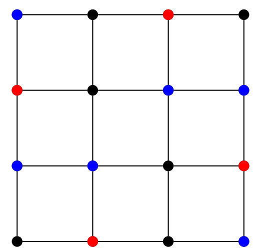
$$\gamma(G\Box H) \leq |S \times V(H)| = \gamma(G)n(H).$$

By analogy we deduce that $\gamma(G\Box H) \leq n(G)\gamma(H)$, which completes the proof. □

Exercise 174. Prove that for every pair of graphs G and H,

$$\gamma(G\Box H) \geq \frac{n(H)\gamma(G)}{\Delta(H)+1}.$$

Solution: Let D be a $\gamma(G\Box H)$-set, $V(H)=\{v_1,\ldots,v_{n(H)}\}$ and $G_j=\langle V(G)\times\{v_j\}\rangle\cong G$. Now, let S_j be the set of vertices in G_j which are not dominated by vertices in $D_j=D\cap V(G_j)$. We say that the vertices in S_j are horizontally undominated by D and they are vertically dominated by D. To show this notation and terminology, we consider the graph $P_4\Box P_4$ where D corresponds to the set of red-coloured vertices, while the vertices in blue are vertically dominated by D, i.e., $\bigcup_j S_j$ corresponds to the set of blue-coloured vertices.



Obviously, for every vertex $w=(x,y)\in D$, there are at most $|N_H(y)|$ vertices in $\bigcup_j S_j$ which are vertically dominated by w. Hence,

$$|D|\Delta(H)\geq\sum_{(x,y)\in D}|N_H(y)|\geq\sum_{j=1}^{n(H)}|S_j|.$$

On the other hand, every set $D_j\cup S_j$ is a dominating set of the graph $G_j\cong G$, which is the j-th copy of G in $G\Box H$. Thus, $|D_j|+|S_j|\geq\gamma(G)$ for every $j\in\{1,\ldots,n(H)\}$, which implies that

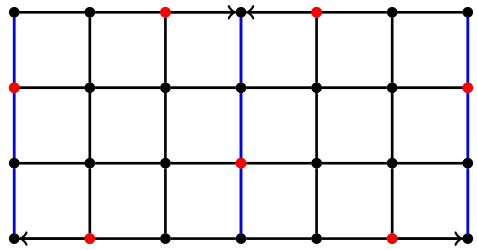
$$\begin{aligned}\gamma(G\Box H)\Delta(H)&=|D|\Delta(H)\\&\geq\sum_{(x,y)\in D}|N_H(y)|\\&\geq\sum_{j=1}^{n(H)}|S_j|\\&\geq\sum_{j=1}^{n(H)}(\gamma(G)-|D_j|)\\&=n(H)\gamma(G)-|D|\\&=n(H)\gamma(G)-\gamma(G\Box H).\end{aligned}$$

Therefore, the result follows. □

Exercise 175. Prove that for every pair of graphs G and H,

$$\gamma(G\Box H)\geq\max\{\gamma(G)\rho(H),\gamma(H)\rho(G)\}.$$

Solution: Let D be a $\gamma(G\Box H)$-set, and let P be a $\rho(G)$-set. If H_x denotes the copy of H associated to $x\in V(G)$ in $G\Box H$, then the projection of $D_x=D\cap(N_G[x]\times V(H))$ on H_x is a dominating set of $H_x\cong H$.



To show this idea, we have considered the graph $P_7\Box P_4$, where the copies H_x are represented in blue for every $x\in X=\{1,4,7\}$ and projections are indicated by arrows. $D_4=\{(4,2),(3,4),(5,4)\}$ and its projection on H_4 is $\{4,2),(4,4)\}$, which is a dominating set of H_4.

Since $N_G[x]\cap N[y]=\varnothing$ for every pair of different vertices $x,y\in P$,

$$\gamma(G\Box H)=|D|\geq\sum_{x\in P}|D_x|\geq\rho(G)\gamma(H).$$

By analogy we deduce that $\gamma(G\Box H)\geq\gamma(G)\rho(H)$, which completes the proof. □

Since $\gamma(T)=\rho(T)$ for every tree T, from the exercise above we deduce that if one factor in the Cartesian product graph is a tree, then Vizing's conjecture holds.

Corollary 84. *For every graph G and every tree T,*

$$\gamma(G\Box T)\geq\gamma(G)\gamma(T).$$

9.1.5 Independence and vertex cover

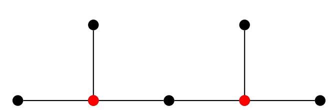
Definition 61. A set S of vertices of G is a *vertex cover* if every edge of G is incident with at least one vertex in S. The *vertex cover number* of G, denoted by $\beta(G)$, is the minimum cardinality among all vertex covers of G.

A vertex cover of cardinality $\beta(G)$ will be called n $\beta(T)$-set.

Definition 62. A set S of vertices of G is an *independent set* if the subgraph induced by S is empty. The *independence number* of G, denoted by $\alpha(G)$, is the maximum cardinality among all independent sets of G.

An independent set of cardinality $\alpha(G)$ will be called an $\alpha(G)$-set.

Example 89. The set of red coloured vertices is a $\beta(G)$-set, while the set of black coloured vertices form an $\alpha(G)$-set.



□

The following well-known result, due to Gallai, states the relationship between the independence number and the vertex cover number of a graph.

Theorem 85 (Gallai's theorem, [4]). *For any graph G of order n,*

$$\alpha(G)+\beta(G)=n.$$

Proof. For any $\alpha(G)$-set S, the set $V(G)\setminus S$ is a vertex cover, which implies that $\beta(G)\leq |V(G)\setminus S|=n-\alpha(G)$. Thus,

$$\alpha(G)+\beta(G)\leq n.$$

On the other hand, for every $\beta(G)$-set D, we have that $V(G)\setminus D$ is an independent set. Hence, $\alpha(G)\geq |V(G)\setminus D|=n-\beta(G)$, which leads to

$$\alpha(G)+\beta(G)\geq n.$$

Therefore, the result follows. □

Exercise 176. Let G be a graph. Prove that if G does not have isolated vertices, then

$$\gamma(G)\leq \min\{\alpha(G),\beta(G)\}.$$

Solution: Let $X\subseteq V(G)$ be an $\alpha(G)$-set. If the exists $v\in V(G)\setminus X$ such that $N(v)\cap X=\varnothing$, then $X\cup\{v\}$ is an independent set, which is a contradiction. Therefore, X is a dominating set, which implies that $\gamma(G)\leq |X|=\alpha(G)$.

Now, let $Y\subseteq V(G)$ be a $\beta(G)$-set. If the exists $u\in V(G)\setminus Y$ such that $N(u)\cap Y=\varnothing$, then $N(u)\subseteq V(G)\setminus Y$, which is a contradiction, as $V(G)\setminus Y$ is an independent set and G does not have isolated vertices. Therefore, Y is a dominating set, which implies that $\gamma(G)\leq |Y|=\beta(G)$. □

There are some parameters, like the matching number, which are closely related to the vertex cover number. We suggest the book [5] for details on matchings, network flows and applications.

Exercise 177. Find $\alpha(Q_k)$ for every integer $k\geq 1$.

Solution: Since the hypercube Q_k is a k-regular bipartite graph, $\alpha(Q_k)=\frac{n(Q_k)}{2}=2^{k-1}$.

Exercise 178. Show that for any pair of graphs G and H,

$$\alpha(G\Box H)\geq \alpha(G)\alpha(H).$$

Solution: Let S_1 be an $\alpha(G)$-set and let S_2 be an $\alpha(H)$-set. We proceed to show that $S_1\times S_2$ is an independent set of $G\Box H$. We differentiate the following cases for two vertices $(g,h),(g',h')\in S_1\times S_2$.

Case 1. $g\neq g'$ and $h\neq h'$. By definition of Cartesian product graph, $(g,h)\not\sim (g',h')$.

Case 2. $g=g'$. Since, S_2 is an independent set, $h\not\sim h'$. Hence, by definition of Cartesian product graph, $(g,h)\not\sim (g',h')$.

Case 3. $h=h'$. Since, S_1 is an independent set, $g\not\sim g'$. Thus, by definition of Cartesian product graph, $(g,h)\not\sim (g',h')$.

According to the three cases above, $S_1\times S_2$ is an independent set of $G\Box H$. Therefore, $\alpha(G\Box H)\geq |S_1\times S_2|=|S_1||S_2|=\alpha(G)\alpha(H)$. □

9.2 Domination versus total domination

Definition 63. A set $S \subseteq V$ is a *total dominating set* of G if $N(v) \cap S \neq \varnothing$ for every vertex $v \in V$.

Definition 64. The *total domination number* of a graph G is defined to be

$$\gamma_t(G) = \min\{|S| : \ S \text{ is a total dominating set of } G\}.$$

A total dominating set of cardinality $\gamma_t(G)$ will be called a $\gamma_t(G)$-set.

For instance, $\gamma_t(K_n) = 2$ for $n \geq 2$, $\gamma_t(Q_3) = 4$, $\gamma_t(C_5) = 3$ and $\gamma_t(K_{r,s}) = 2$.
It is readily seen that for any graph with no isolated vertices

$$\gamma(G) \leq \gamma_t(G) \leq 2\gamma(G).$$

Exercise 179. Let G be a graph of order n and maximum degree $\Delta(G)$. Prove that if G does not have isolated vertices, then

$$\gamma_t(G) \geq \left\lceil \frac{n}{\Delta(G)} \right\rceil.$$

Solution: For any $\gamma_t(G)$-set S we have $V = \bigcup\limits_{v \in S} N(v)$, which implies that

$$n = |V| = \left| \bigcup_{v \in S} N(v) \right| \leq \sum_{v \in S} \delta(v) \leq |S|\Delta(G) = \gamma_t(G)\Delta(G).$$

Therefore, the lower bound follows. □

9.2.1 The case of paths and cycles

Exercise 180. Prove that for any integer $n \geq 3$,

$$\gamma_t(C_n) = \gamma_t(P_n) = \begin{cases} \left\lceil \frac{n}{2} \right\rceil, & \text{if } n \not\equiv 2 \pmod 4, \\ \frac{n}{2} + 1, & \text{if } n \equiv 2 \pmod 4. \end{cases}$$

Solution: Since $\gamma_t(C_n) \leq \gamma_t(P_n)$, we only need to show that the required value is a lower bound for the case of cycles and an upper bound for the case of paths.

First, let $V(C_n) = \{v_0, \ldots, v_{n-1}\}$ and assume that v_i is adjacent to v_{i+1} for any i, where the addition in the subscripts is taken modulo n. For any $\gamma(C_n)$-set S and any $i \in \{0, \ldots, n-1\}$ we have $|S \cap \{v_i, v_{i+1}, v_{i+2}, v_{i+3}\}| \geq 2$. Hence,

$$4\gamma(C_n) = 4|S| = \sum_{i=0}^{n-1} |S \cap \{v_i, v_{i+1}, v_{i+2}, v_{i+3}\}\}| \geq 2n.$$

Therefore, $\gamma_t(C_n) \geq \left\lceil \frac{n}{2} \right\rceil$, as required for $n \not\equiv 2 \pmod 4$. (Notice that this part can be deduced by Exercise 179.)

Now, suppose that $n \equiv 2 \pmod 4$ and $\gamma_t(C_n) = \frac{n}{2}$. In such a case, for any $i \in \{0,\dots,n-1\}$ we have $|S \cap \{v_i, v_{i+1}, v_{i+2}, v_{i+3}\}| = 2$. We can assume, without loss of generality, that $v_0, v_1 \in S$, $v_2, v_3 \notin S$, $v_4, v_5 \in S$, $\dots$,$v_{n-2}, v_{n-3} \in S$ and $v_{n-1} \notin S$, which is a contradiction, as $|S \cap \{v_0, v_{n-1}, v_{n-2}, v_{n-3}\}| = 3$. Therefore, for $n \equiv 2 \pmod 4$ we have $\gamma_t(C_n) \geq \frac{n}{2} + 1$.

Now, let $V(P_n) = \{u_1, \dots, u_n\}$. We proceed to construct a total dominating set D of P_n having the required cardinality.

- If $n \equiv 0 \pmod 4$, then $D = \{u_2, u_3, u_6, u_7, \dots, u_{n-2}, u_{n-1}\}$. In this case, $|D| = \frac{n}{2} = \left\lceil \frac{n}{2} \right\rceil$.
- If $n \equiv 1 \pmod 4$, then $D = \{u_2, u_3, u_6, u_7, \dots, u_{n-3}, u_{n-2}, u_{n-1}\}$. Now, $|D| = \frac{n+1}{2} = \left\lceil \frac{n}{2} \right\rceil$.
- If $n \equiv 2 \pmod 4$, then $D = \{u_2, u_3, u_6, u_7, \dots, u_{n-4}, u_{n-3}, u_{n-1}, u_n\}$. Now, $|D| = \frac{n}{2} + 1$.
- If $n \equiv 3 \pmod 4$, then $D = \{u_2, u_3, u_6, u_7, \dots, u_{n-1}, u_n\}$. Now, $|D| = \frac{n+1}{2} = \left\lceil \frac{n}{2} \right\rceil$.

By simple inspection, we can verify that D is a total dominating set of P_n, which implies that $\gamma_t(P_n) \leq \left\lceil \frac{n}{2} \right\rceil$ whenever $n \not\equiv 2 \pmod 4$, while $\gamma_t(P_n) \leq \frac{n}{2} + 1$ for $n \equiv 2 \pmod 4$. Therefore, we conclude the proof by combining these upper bounds on $\gamma_t(P_n)$ with the lower bounds obtained previously on $\gamma_t(C_n)$. □

9.2.2 The case of lexicographic product graphs

Exercise 181. Prove that for any graph G with no isolated vertex and any nontrivial graph H,

$$\gamma(G \circ H) = \begin{cases} \gamma(G), & \text{if } \gamma(H) = 1, \\ \gamma_t(G), & \text{if } \gamma(H) \geq 2. \end{cases}$$

Solution: let S be a $\gamma_t(G \circ H)$-set. We define the following subsets of $V(G)$, where H_v is the copy of H in $G \circ H$ associated to $v \in V(G)$:

$$\mathcal{A}_S = \{v \in V(G) : |S \cap V(H_v)| \geq 2\};$$

$$\mathcal{B}_S = \{v \in V(G) : |S \cap V(H_v)| = 1\};$$

$$\mathcal{C}_S = \{v \in V(G) : S \cap V(H_v) = \varnothing\}.$$

We differentiate the following two cases.

Case 1: $\gamma(H) = 1$. Let $v \in V(H)$ be a universal vertex. Observe that for any $\gamma(G)$-set D we have that $D' = D \times \{v\}$ is a dominating set of $G \circ H$. Hence, $\gamma(G \circ H) \leq |D'| = |D| = \gamma(G)$.

Notice that $S' = \mathcal{A}_S \cup \mathcal{B}_S$ is a dominating set of G and so

$$\gamma(G) \leq |S'| \leq |S| = \gamma(G \circ H).$$

Therefore, $\gamma(G \circ H) = \gamma(G)$.

Case 2: $\gamma(H) \geq 2$. In this case, we take D as a $\gamma_t(G)$-set and $v \in V(H)$ an arbitrary vertex of H to conclude that $D' = D \times \{v\}$ is a dominating set of $G \circ H$. Hence, $\gamma(G \circ H) \leq |D'| = |D| = \gamma_t(G)$.

Now, we define a set $S' \subseteq V(G)$ as follows.

- For every vertex $x \in \mathcal{A}_S \cup \mathcal{B}_S$, set $x \in S'$.
- For every vertex $x \in \mathcal{A}_S$, choose a vertex $x' \in N(x) \setminus (\mathcal{A}_S \cup \mathcal{B}_S)$ (if any) and set $x' \in S'$.

Since G does not have isolated vertices, S' is a total dominating set of G. Hence,

$$\gamma_t(G) \leq |S'| \leq |S| = \gamma(G \circ H),$$

which completes the proof. □

Exercise 182. Show that for any graph G with no isolated vertex and any graph H,

$$\gamma_t(G \circ H) = \gamma_t(G).$$

Solution: Let D be a $\gamma_t(G)$-set and let $v \in V(H)$. Observe that $D' = D \times \{v\}$ is a total dominating set of $G \circ H$. Hence, $\gamma_t(G \circ H) \leq |D'| = |D| = \gamma_t(G)$.

Now, let S be a $\gamma_t(G \circ H)$-set. We define the following subsets of $V(G)$, where H_v is the copy of H in $G \circ H$ associated to $v \in V(G)$:

$$\mathcal{A}_S = \{v \in V(G) : |S \cap V(H_v)| \geq 2\};$$

$$\mathcal{B}_S = \{v \in V(G) : |S \cap V(H_v)| = 1\};$$

$$\mathcal{C}_S = \{v \in V(G) : S \cap V(H_v) = \varnothing\}.$$

Now, we define $S' \subseteq V(G)$ as follows.

- For every vertex $x \in \mathcal{A}_S \cup \mathcal{B}_S$, set $x \in S'$.
- For every vertex $x \in \mathcal{A}_S$, choose a vertex $x' \in N(x) \setminus (\mathcal{A}_S \cup \mathcal{B}_S)$ (if any) and set $x' \in S'$.

Since G does not have isolated vertices, S' is a total dominating set of G. Hence,

$$\gamma_t(G) \leq |S'| \leq |S| = \gamma_t(G \circ H),$$

which completes the proof. □

9.3 Roman domination and the differential of a graph

Let G be a graph, and let $f : V(G) \longrightarrow \{0,1,2\}$ be a function and $V_i = \{v \in V(G) : f(v) = i\}$ for $i \in \{0,1,2\}$. We will identify f with these subsets of $V(G)$ induced by f, and write $f(V_0,V_1,V_2)$. The *weight* of f is defined to be

$$\omega(f) = f(V(G)) = \sum_{v \in V(G)} f(v) = \sum_i i|V_i|.$$

Definition 65. A *Roman dominating function*, abbreviated RDF, on a graph G is a function $f(V_0,V_1,V_2)$ satisfying the condition that every vertex $u \in V_0$ is adjacent to at least one vertex $v \in V_2$.

The *Roman domination number*, denoted by $\gamma_R(G)$, is the minimum weight among all RDFs on G, *i.e.*,

$$\gamma_R(G) = \min\{\omega(f) : f \text{ is an RDF on } G\}.$$

An RDF of weight $\gamma_R(G)$ is called a $\gamma_R(G)$-function.

Exercise 183. Show that $\gamma(G) \leq \gamma_R(G) \leq 2\gamma(G)$, for every graph G.

Solution: Let $f(V_0,V_1,V_2)$ be a $\gamma_R(G)$-function. Since $V_1 \cup V_2$ is a dominating set,

$$\gamma(G) \leq |V_1| + |V_2| \leq |V_1| + 2|V_2| = \gamma_R(G).$$

Now, for any $\gamma(G)$-set S, the function $f(V(G)\setminus S, \varnothing, S)$ is a Roman dominating function, which implies that $\gamma_R(G) \leq \omega(f) = 2\gamma(G)$. □

Exercise 184. A *Roman graph* is a graph G with $\gamma_R(G) = 2\gamma(G)$. Give three examples of families of Roman graphs.

Solution: We leave the details to the reader:

(a) Every graph with $\gamma(G) = 1$ is a Roman graph, as in such a case $\gamma_R(G) = 2\gamma(G) = 2$.

(b) For any graph G and any graph H of order $n(H) \geq 2$, the corona graph $G \odot H$ is a Roman graph, as $n(H) \geq 2$ leads to $\gamma_R(G \odot H) = 2n(G) = 2\gamma(G \circ H)$.

(c) For any pair of integers $r \geq 3$ and $s \geq 3$, the complete bipartite graph $K_{r,s}$ is a Roman graph. In this case, $\gamma_R(K_{r,s}) = 2\gamma(K_{r,s}) = 4$.

□

Exercise 185. Characterize the graphs with $\gamma_R(G) = \gamma(G)$.

Solution: Assume $\gamma_R(G) = \gamma(G)$. Let $f(V_0,V_1,V_2)$ be a $\gamma_R(G)$-function. Since $V_1 \cup V_2$ is a dominating set,

$$\gamma(G) \leq |V_1| + |V_2| \leq |V_1| + 2|V_2| = \gamma_R(G) = \gamma(G).$$

Hence, $V_2 = \varnothing$, which implies that $|V(G)| = |V_1| = \gamma(G)$, and so G is an empty graph. Obviously, $\gamma_R(N_n) = n$. Therefore, $\gamma_R(G) = \gamma(G)$ if and only if G is an empty graph. □

Exercise 186. Find $\gamma_R(G \odot K_1)$ for every graph G.

Solution: For every $u_i \in V(G)$, the vertex of the copy of K_1 associated to u_i will be denoted by v_i. For any $\gamma(G)$-set S, the function $g(X_0,X_1,X_2)$ defined from $X_2 = S$ and $X_1 = \{v_i : u_i \in V(G)\setminus S\}$, is a Roman dominating function. Hence,

$$\gamma_R(G \odot K_1) \leq \omega(g) = 2|X_2| + |X_1| = n(G) + \gamma(G).$$

Now, let $f(V_0,V_1,V_2)$ be a $\gamma_R(G \odot K_1)$-function such that $V_2 \subseteq V(G)$ and $|V_2|$ is maximum among all $\gamma_R(G \odot K_1)$-functions. Note that the existence of such a function is always possible. Observe the following facts.

- No vertex in V_2 is adjacent to a vertex in V_1.
- $V_1 \cap V(G) = \varnothing$.
- If $f(u_i) = 0$, then $f(v_i) = 1$.

- If $f(u_i)=2$, then $f(v_i)=0$.

Hence, V_2 is a dominating set of G and $|V_2|+|V_1|=n(G)$, which implies that

$$n(G)+\gamma(G)\leq 2|V_2|+|V_1|=\gamma_R(G\odot K_1)\leq n(G)+\gamma(G).$$

Therefore, $\gamma_R(G\odot K_1)=n(G)+\gamma(G)$. □

Exercise 187. Show that for any graph G of order n and maximum degree Δ,

$$\gamma_R(G)\geq \frac{2n}{\Delta+1}.$$

Solution: For any $\gamma_R(G)$-function $f(V_0,V_1,V_2)$,

$$\begin{aligned}2n =&(2|V_0|+|V_1|)+(|V_1|+2|V_2|)\\ =&(2|V_0|+|V_1|)+\gamma_R(G)\\ \leq&(2\Delta|V_2|+|V_1|)+\gamma_R(G)\\ \leq&\Delta(2|V_2|+|V_1|)+\gamma_R(G)\\ =&\Delta\gamma_R(G)+\gamma_R(G).\end{aligned}$$

Therefore, $\gamma_R(G)\geq \frac{2n}{\Delta+1}$. □

Exercise 188. Show that if G is a graph of order n with no isolated vertex, then $\gamma_R(G)=n$ if and only if $G=\bigcup_{i=1}^{\frac{n}{2}} K_2$.

Solution: Since $\gamma_R(K_2)=2$, we have that $\gamma_R\left(\bigcup_{i=1}^{\frac{n}{2}} K_2\right)=n$.

Now, assume that G does not have isolated vertices and $\gamma_R(G)=n$. Suppose that there exists a vertex $v\in V(G)$ of degree $\delta(v)\geq 2$. In such a case, we construct a Roman dominating function $f(V_0,V_1,V_2)$ on G by $V_2=\{v\}$, $V_1=V(G)\setminus N[v]$ and $V_0=N(v)$. Thus, $\gamma_R(G)=\omega(f)<n$, which is a contradiction. Therefore, every vertex of G has degree one, and so $G=\bigcup_{i=1}^{\frac{n}{2}} K_2$. □

Exercise 189. Let G be a connected graph of order n. Show that $\gamma_R(G)=\gamma(G)+1$ if and only if there exists a vertex of G of degree $n-\gamma(G)$.

Solution: First, assume that G is connected graph and has a vertex v of degree $\delta(v)=n-\gamma(G)$. We already know that $\gamma_R(G)\geq\gamma(G)+1$, as $G\ncong N_n$. Thus, the function $f(V_0,V_1,V_2)$ defined on G by $V_2=\{v\}$, $V_1=V(G)\setminus N[v]$ and $V_0=N(v)$, is a Roman dominating function. Thus, $\gamma_R(G)\leq\omega(f)=\gamma(G)+1$. Therefore, $\gamma_R(G)=\gamma(G)+1$.

From now on we assume that G is a connected graph of order $n$and $\gamma_R(G)=\gamma(G)+1$. Let $g(X_0,X_1,X_2)$ be a $\gamma_R(G)$-function. Since $X_1\cup X_2$ is a dominating set and $|X_1|+2|X_2|=\gamma(G)+1$, we have two possibilities.

Case 1. $|X_1|=\gamma(G)+1$ and $X_2=\varnothing$. In this case $X_1=V(G)$ and so $n-1=\gamma(G)\leq n-\Delta$. Thus, the maximum degree of G is $\Delta=1$ and, by the connectivity of G, we conclude that $G\cong K_2$, which implies there exists a vertex of degree $n-\gamma(G)$, as required.

Case 2. $|X_1|=\gamma(G)-1$ and $X_2=\{v\}$ for some $v\in V(G)$. Since $N(v)=X_0$, we conclude that $\delta(V)=|N(v)|=n-|X_1|-1=n-\gamma(G)$, as required. □

The differential of a graph

Given a graph G, the *open neighbourhood of a set* $S \subseteq V(G)$ is defined as

$$N(S) = \bigcup_{v \in S} N(v),$$

while the *external neighbourhood* of S, or boundary of S, is defined as

$$N_e(S) = N(S) \setminus S.$$

Definition 66. The *differential of a set* $S \subseteq V(G)$ is defined as

$$\partial(S) = |N_e(S)| - |S|,$$

while the *differential of a graph G* is defined to be

$$\partial(G) = \max\{\partial(S) : S \subseteq V(G)\}.$$

Lewis et al. [8] motivated the definition of differential from the following game, what we call *graph differential game*. "*You are allowed to buy as many tokens as you like, say k tokens, at a cost of one dollar each. You then place the tokens on some subset D of k vertices of a graph G. For each vertex of G which has no token on it, but is adjacent to a vertex with a token on it, you receive one dollar. Your objective is to maximize your profit, that is, the total value received minus the cost of the tokens bought*". Obviously, $\partial(D) = |N_e(D)| - |D|$ is the profit obtained with the placement D, while the maximum profit equals $\partial(G)$.

A Gallai-type theorem has the form $a(G) + b(G) = n$, where $a(G)$ and $b(G)$ are parameters defined on G. This terminology comes from Theorem 85, which was stated in 1959 by the prolific Hungarian mathematician Tibor Gallai.

Theorem 86 (Gallai-type theorem for the differential and the Roman domination number)**.** *For any graph G of order n,*

$$\gamma_R(G) + \partial(G) = n.$$

Proof. Let D be a $\partial_s(G)$-set. We define a function $g(W_0, W_1, W_2)$ from the sets $W_2 = D$, $W_0 = N_e(D)$ and $W_1 = V(G) \setminus (W_0 \cup W_2)$. Observe that g is a Roman dominating function on G, which implies that

$$\begin{aligned}
\gamma_R(G) \leq& \omega(g) \\
=& 2|W_2| + |W_1| \\
=& 2|D| + (n - |D| - |N_e(D)|) \\
=& n - \partial(D) \\
=& n - \partial(G).
\end{aligned}$$

We proceed to show that $\gamma_R(G) \geq n - \partial_s(G)$. Let $f(V_0, V_1, V_2)$ be a $\gamma_R(G)$-function. It is readily seen that,

$$\begin{aligned}
\partial_s(G) &\geq \partial_s(V_2) \\
&= |N_e(V_2)| - |V_2| \\
&= |V_0| - |V_2| \\
&= (|V_0| + |V_1| + |V_2|) - (|V_1| + 2|V_2|) \\
&= n - \omega(f) \\
&= n - \gamma_R(G).
\end{aligned}$$

Therefore, the result follows. □

9.4 Secure domination

Definition 67. A set $S \subseteq V$ is a *secure dominating set* of G if the following two conditions hold.

- S is a dominating set.
- For every $u \in V(G) \setminus S$ there exists $v \in S \cap N(u)$ such that $(S \setminus \{v\}) \cup \{u\}$ is a dominating set.

Definition 68. The *secure domination number* of a graph G is defined to be

$$\gamma_s(G) = \min\{|S| : \ S \text{ is a secure dominating set of } G\}.$$

A secure dominating set of cardinality $\gamma_s(G)$ will be called a $\gamma_s(G)$-set.

Exercise 190. Let G be the Petersen graph. Find $\gamma_s(G)$.

Solution: $\gamma_s(G) = 4$ □

Exercise 191. Show that for every graph G,

$$\gamma_s(G) \leq \gamma(G) + \alpha(G) - 1.$$

Solution: Let D be a $\gamma(G)$-set and let G' be the graph induced by $V(G) \setminus D$. Let I be an $\alpha(G')$-set. Note that for every $x \in V(G) \setminus (D \cup I)$ we have that $D \cap N(x) \neq \varnothing$ and $I \cap N(x) \neq \varnothing$. Thus, $D \cup I$ is a secure dominating set of G, which implies that

$$\gamma_s(G) \leq |D| + |I| = \gamma(G) + \alpha(G').$$

Hence, if $\alpha(G') \leq \alpha(G) - 1$, then we are done. From now on, we assume $\alpha(G') = \alpha(G)$. For a vertex $v \in I$, we will show that $S = D \cup I \setminus \{v\}$ is a secure dominating set of G.

Let $pn(v, I)$ be the set of private neighbours of v with respect to I. Observe that the subgraph induced by $pn(v, I)$ is a complete subgraph, otherwise I is not an $\alpha(G)$-set, which is a contradiction. With these facts in mind we differentiate the following cases on $x \in V(G) \setminus S$.

Case 1. $x = v$. In this case, for any $y \in N(v) \cap D$, the set $(S \setminus \{y\}) \cup \{v\} = (D \setminus \{y\}) \cup I$ is a dominating set of G.

Case 2. $x \in pn(v,I)$. Since every vertex in $pn(v,I) \setminus \{x\}$ is adjacent to x, we have that for any $y \in N(x) \cap D$ the set $(S \setminus \{y\}) \cup \{x\}$ is a dominating set of G.

Case 3. $x \neq v$ and $x \notin pn(v,I)$. In this case, there exists $v' \in N(x) \cap (I \setminus \{v\})$ and the set $(S \setminus \{v'\}) \cup \{x\}$ is a dominating set of G.

According to the three cases above, $S = D \cup I \setminus \{v\}$ is a secure dominating set of G, which implies that

$$\gamma_s(G) \leq |D| + |I| - 1 = \gamma(G) + \alpha(G) - 1.$$

□

Exercise 192. Find $\gamma_s(K_n + P_k)$ for any two integers $n \geq 1$ and $k \geq 3$. In the solution, you can omit the details, but find the correct value.

Solution: If $n \geq 2$, then $\gamma_s(K_n + P_k) = 2$, while

$$\gamma_s(K_1 + P_k) = \begin{cases} \frac{k}{3} + 1 & \text{if } k \equiv 0 \pmod 3 \\ \frac{k-1}{3} + 1 & \text{if } k \equiv 1 \pmod 3 \\ \frac{k-2}{3} + 1 & \text{if } k \equiv 2 \pmod 3. \end{cases}$$

□

Exercise 193. Let G be a graph of order $n \geq 1$. Find $\gamma_s(G \odot P_k)$ for any integer $k \geq 3$. In the solution, you can omit the details, but find the correct value.

Solution:

$$\gamma_s(G \odot P_k) = \begin{cases} \frac{n(k+3)}{3} & \text{if } k \equiv 0 \pmod 3 \\ \frac{n(k+2)}{3} & \text{if } k \equiv 1 \pmod 3 \\ \frac{n(k+1)}{3} & \text{if } k \equiv 2 \pmod 3. \end{cases}$$

□

Bibliography

[1] Leonard M. Blumenthal. *Theory and applications of distance geometry*. Oxford University Press, 1953.

[2] Boštjan Brešar, Paul Dorbec, Wayne Goddard, Bert L. Hartnell, Michael A. Henning, Sandi Klavžar, and Douglas F. Rall. Vizing's conjecture: a survey and recent results. *Journal of Graph Theory*, 69(1):46–76, 2012.

[3] A. Cayley. On the colouring of maps. *Proceedings of the Royal Geographical Society and Monthly Record of Geography, New Monthly Series*, 1(4):259 – 261, 1879.

[4] Tibor Gallai. Über extreme Punkt- und Kantenmengen. *Annales Universitatis Scientarium Budapestinensis de Rolando Eötvös Nominatae, Sectio Mathematica*, 2:133–138, 1959.

[5] J. L. Gross and J. Yellen. *Graph theory and its applications*. Chapman & Hall/CRC, Boca Raton, FL, second edition, 2006.

[6] Richard Hammack, Wilfried Imrich, and Sandi Klavžar. *Handbook of product graphs*. Discrete Mathematics and its Applications. CRC Press, 2 edition, 2011.

[7] Wilfried Imrich and Sandi Klavžar. *Product graphs, structure and recognition*. Wiley-Interscience series in discrete mathematics and optimization. Wiley, 2000.

[8] J. L. Lewis, T. W. Haynes, S. M. Hedetniemi, S. T. Hedetniemi, and P. J. Slater. Differentials in graphs. *Util. Math.*, 69:43–54, 2006.

[9] Neil Robertson, Daniel Sanders, Paul Seymour, and Communicated Graham. A new proof of the four-colour theorem. *Electron. Res. Announc. Amer. Math. Soc.*, 2, 10 1996.

[10] Vadim G. Vizing. The cartesian product of graphs. *Vyčisl. Sistemy No.*, 9:30–43, 1963.