Chapter 1

Propositional Logic (PL)

Let $\mathsf{WFF}_{\mathsf{PL}}(\mathcal{P})$ be the set of well-formed formulas of *propositional logic* over the set $\mathcal{P}$ of *propositional variables*. We say a set $\Gamma \subseteq \mathsf{WFF}_{\mathsf{PL}}(\mathcal{P})$ is *finitely satisfiable* iff every finite subset of Γ is satisfiable. If Γ is a finite set, then "finitely satisfiable" coincides with "satisfiable". Satisfiability of wff's is defined in Appendix B.1.

We write $\mathsf{models}(\Gamma)$ to denote the set of models of Γ. In the propositional case, $\mathsf{models}(\Gamma)$ is the set of all truth assignments to the propositional variables that satisfy every $\varphi \in \Gamma$. Thus, Γ is satisfiable iff $\mathsf{models}(\Gamma) \neq \varnothing$. The next lemma is a preliminary result for the Compactness Theorem.

Lemma 1. *Let $\Gamma \subseteq \mathsf{WFF}_{\mathsf{PL}}(\mathcal{P})$ and $\varphi \in \mathsf{WFF}_{\mathsf{PL}}(\mathcal{P})$. If Γ is finitely satisfiable, then $\Gamma \cup \{\varphi\}$ or $\Gamma \cup \{\neg\varphi\}$ (or possibly both) is finitely satisfiable.*

Proof. Suppose the conclusion of the lemma does not hold: Both $\Gamma \cup \{\varphi\}$ and $\Gamma \cup \{\neg\varphi\}$ are not finitely satisfiable. Hence, there are finite subsets $\Gamma_1 \subseteq \Gamma$ and $\Gamma_2 \subseteq \Gamma$ such that both $\Gamma_1 \cup \{\varphi\}$ and $\Gamma_2 \cup \{\neg\varphi\}$ are not satisfiable. Hence, both:

$$\mathsf{models}(\Gamma_1) \cap \mathsf{models}(\varphi) \;=\; \varnothing \qquad \text{and} \qquad \mathsf{models}(\Gamma_2) \cap \mathsf{models}(\neg\varphi) \;=\; \varnothing.$$

Hence, both $\mathsf{models}(\Gamma_1) \subseteq \mathsf{models}(\neg\varphi)$ and $\mathsf{models}(\Gamma_2) \subseteq \mathsf{models}(\varphi)$. Hence,

$$\mathsf{models}(\Gamma_1 \cup \Gamma_2) \;=\; \mathsf{models}(\Gamma_1) \cap \mathsf{models}(\Gamma_2) \;\subseteq\; \mathsf{models}(\neg\varphi) \cap \mathsf{models}(\varphi) \;=\; \varnothing.$$

Hence, the finite subset $\Gamma_1 \cup \Gamma_2$ does not have models, *i.e.*, is not satisfiable. Hence, Γ is not finitely satisfiable, and the hypothesis of the lemma does not hold either. □

1.1 Compactness in PL

Theorem 2 (Compactness for Propositional Logic, Version I)**.** *Let $\Gamma \subseteq \mathsf{WFF}_{\mathsf{PL}}(\mathcal{P})$. It then follows that: Γ is satisfiable $\Leftrightarrow$ Γ is finitely satisfiable.*

Proof. The implication "$\Rightarrow$" is immediate. The non-trivial implication is "$\Leftarrow$": If Γ is finitely satisfiable, then Γ is satisfiable.

The set of propositional variables is $\mathcal{P} = \{p_0, p_1, p_2, \ldots\}$. Let $\varphi_1, \varphi_2, \varphi_3, \ldots$ be a fixed, countably infinite, enumeration of all the formulas in $\mathsf{WFF}_{\mathsf{PL}}(\mathcal{P})$. We define a nested sequence of supersets of Γ as follows:

$$\Delta_0 \quad = \Gamma,$$

$$\Delta_{i+1} = \begin{cases} \Delta_i \cup \{\varphi_i\} & \text{if } \Delta_i \cup \{\varphi_i\} \text{ is finitely satisfiable,} \\ \Delta_i \cup \{\neg\varphi_i\} & \text{otherwise.} \end{cases}$$

Clearly, $\Gamma = \Delta_0 \subseteq \Delta_1 \subseteq \Delta_2 \subseteq \Delta_3 \subseteq \cdots$. By induction on $i \geqslant 0$, using Lemma 1, every Δ_i is a finitely satisfiable set of propositional wff's. We now define:

$$\Delta \;=\; \bigcup_i \Delta_i \qquad \text{(the limit of the } \Delta_i\text{'s)}$$

Two facts about Δ follow from its definition:

1. For every propositional wff φ, either $\varphi \in \Delta$ or $\neg\varphi \in \Delta$, but not both.
   This is why Δ is said *maximal finitely satisfiable*, soon to be shown just *maximal satisfiable*.
2. Since every propositional variable p_i is a wff itself, either $p_i \in \Delta$ or $\neg p_i \in \Delta$, but not both.

We next define a truth assignment σ as follows:

$$\sigma(p_i) \;=\; \begin{cases} \mathit{true} & \text{if } p_i \in \Delta, \\ \mathit{false} & \text{if } \neg p_i \in \Delta. \end{cases}$$

Claim: σ satisfies a propositional wff φ iff $\varphi \in \Delta$. We leave the proof of this claim as an (easy) exercise.

Hence, σ is a valuation satisfying every wff in Δ, *i.e.*, $\sigma \in \mathsf{models}(\Delta)$. Hence, because $\Gamma \subseteq \Delta$, it is also the case that σ satisfies every wff in Γ. Hence, Γ is satisfiable. □

Exercise 3. Provide the details in the preceding proof showing that there is "a fixed, countably infinite, enumeration of all the formulas in $\mathsf{WFF}_{\mathsf{PL}}(\mathcal{P})$". Although not needed in the proof, we can state a stronger assertion: The fixed enumeration of all the formulas in $\mathsf{WFF}_{\mathsf{PL}}(\mathcal{P})$ is *computable*, *i.e.*, can be generated by an infinitely-running computer program. □

Exercise 4. In the definition of the nested sequence of Δ_i's in the preceding proof, we did *not* write:

$$\Delta_{i+1} \;=\; \begin{cases} \Delta_i \cup \{\varphi_i\} & \text{if } \Delta_i \cup \{\varphi_i\} \text{ is finitely satisfiable,} \\ \Delta_i \cup \{\neg\varphi_i\} & \text{if } \Delta_i \cup \{\neg\varphi_i\} \text{ is finitely satisfiable.} \end{cases}$$

Explain why. *Hint*: Exhibit a set Γ of wff's and a single wff φ such that both $\Gamma \cup \{\varphi\}$ and $\Gamma \cup \{\neg\varphi\}$ are satisfiable. □

Exercise 5. Prove the **claim** in the penultimate paragraph of the proof of Theorem 2. There is no harm in simplifying the syntax a little, by restricting the logical connectives to two, say, $\{\neg, \vee\}$ or $\{\neg, \wedge\}$. *Hint*: Use structural induction on propositional wff's. □

Lemma 6. *Let $\Gamma \subseteq \mathsf{WFF}_{\mathsf{PL}}(\mathcal{P})$ and $\varphi \in \mathsf{WFF}_{\mathsf{PL}}(\mathcal{P})$, both arbitrary. We then have:*
$\Gamma \models \varphi \;\Leftrightarrow\;$ *"$\Gamma \cup \{\neg\varphi\}$ is unsatisfiable" – or, equivalently, $\Gamma \not\models \varphi \;\Leftrightarrow\;$ "$\Gamma \cup \{\neg\varphi\}$ is satisfiable".*

Proof. We have the following sequence of equivalences:

$$\begin{aligned} \Gamma \models \varphi \quad &\Leftrightarrow \quad \mathsf{models}(\Gamma) \subseteq \mathsf{models}(\varphi) \\ &\Leftrightarrow \quad \mathsf{models}(\Gamma) \cap \mathsf{models}(\neg\varphi) = \varnothing \\ &\Leftrightarrow \quad \mathsf{models}(\Gamma \cup \{\neg\varphi\}) = \varnothing \\ &\Leftrightarrow \quad \Gamma \cup \{\neg\varphi\} \text{ is unsatisfiable,} \end{aligned}$$

which is the desired conclusion. □

Corollary 7 (Compactness for Propositional Logic, Version II)**.** *Consider arbitrary $\Gamma \subseteq \mathsf{WFF}_{\mathsf{PL}}(\mathcal{P})$ and $\varphi \in \mathsf{WFF}_{\mathsf{PL}}(\mathcal{P})$. We then have: $\Gamma \models \varphi \;\Leftrightarrow$ there is a finite subset $\Gamma_0 \subseteq \Gamma$ such that $\Gamma_0 \models \varphi$.*

Proof. The implication "$\Leftarrow$" is immediate. For the implication "$\Rightarrow$", we prove the contrapositive. So, suppose $\Gamma_0 \not\models \varphi$ for every finite subset $\Gamma_0 \subseteq \Gamma$. We have the following equivalences:

$$\begin{array}{rcl} \Gamma_0 \not\models \varphi \text{ for every finite } \Gamma_0 \subseteq \Gamma & \Leftrightarrow & \Gamma_0 \cup \{\neg\varphi\} \text{ satisfiable for every finite } \Gamma_0 \subseteq \Gamma \text{ (by Lemma 6)} \\ & \Leftrightarrow & \Gamma \cup \{\neg\varphi\} \text{ finitely satisfiable (by definition)} \\ & \Leftrightarrow & \Gamma \cup \{\neg\varphi\} \text{ satisfiable (by Theorem 2)} \\ & \Leftrightarrow & \Gamma \not\models \varphi \text{ (by Lemma 6)}\ , \end{array}$$

which is the desired conclusion. □

Exercise 8 shows one way in which Compactness breaks down. The exercise involves an extension of PL which is called the *infinitary propositional logic* (*Infinitary PL*).

Exercise 8. We can restrict the logical connectives to $\{\neg, \vee, \wedge\}$. The set of propositional variables is again the countably infinite $\mathcal{P} = \{p_0, p_1, p_2, \ldots\}$. Suppose we extend this syntax with two new connectives, denoted $\bigvee\!\!\!\bigvee$ and $\bigwedge\!\!\!\bigwedge$, each taking as a single argument a set, finite or countably infinite, of previously defined wff's. The resulting syntax is one version of *Infinitary PL*. If Γ is the finite or countably infinite set $\{\varphi_1, \varphi_2, \varphi_3, \ldots\}$, then $\bigvee\!\!\!\bigvee\Gamma$ can be informally understood as:

$$\bigvee\!\!\!\bigvee\Gamma \;=\; \varphi_1 \vee \varphi_2 \vee \varphi_3 \vee \cdots\ ,$$

and similarly for $\bigwedge\!\!\!\bigwedge\Gamma$. Keep in mind that "$\varphi_1 \vee \varphi_2 \vee \varphi_3 \vee \cdots$" is not a legal formal expression; we use $\bigvee\!\!\!\bigvee$ as a unary connective, applied to a single argument which must be a set, and similarly for $\bigwedge\!\!\!\bigwedge$. There are three parts in this exercise:

1. Define the syntax of *Infinitary PL*, preferably in an extended BNF (Backus-Naur Form). Try to be as precise as you can, paying special attention to the presence of ellipses "$\ldots$" in the definition – or can you think of a formulation that avoids any mention of ellipses?
2. Define the semantics of *Infinitary PL*, by structural induction on the syntax in Part 1, starting from an assignment σ of truth values to every member of $\mathcal{P}$ (for the base case of the induction).
3. Show that Theorem 2 does not hold, and therefore nor does Corollary 7, for *Infinitary PL*.

   Hint: Define a countably infinite set Γ of wff's such that every finite $\Gamma_0 \subseteq \Gamma$ is satisfiable, but Γ is not. *Further Hint*: Include wff $\varphi = \bigvee\!\!\!\bigvee\{\neg p_0, \neg p_1, \neg p_2, \ldots\}$ in your proposed Γ. □

Remark 9. The *infinitary propositional logic* (Infinitary PL) extends *finitary propositional logic*, which is just what is commonly called *propositional logic* (our PL here). The distinction between *infinitary* and *finitary* extends to other formal logics and their proof systems besides PL. You can take the phrase "finitary proof system" to qualify a system that generates new finite expressions (*e.g.*, the judgments of PL in natural-deduction style or the wff's of propositional logic in Hilbert- or Frege- style) from previously generated ones by means of finitely many rules that require each finitely many premises or antecedents – without using any notion of infinite sequence or any notion of infinite set.[9] □

[9]The words "finitary" and "infinitary" are also used elsewhere, in several areas of mathematics and theoretical computer science with different though related meanings, and sometimes a little too loosely. As the words suggest, they are intended to mean certain things are "finite" and other things "infinite", but more precisely in relation to particular aspects of mathematically defined notions in different contexts.

For example, closer to our concerns here, a *finitary operation* (resp. *relation*) is one which has *finite arity*, otherwise it is said to be an *infinitary operation* (resp. *infinitary relation*). If we say an algebra or a relational structure is *finite* (resp. *infinite*), we mean its domain or universe is finite (resp. infinite) – this convention is firmly established – but if some authors say an algebra or a relational structure is *finitary* (resp. *infinitary*), they mean its operations and relations have each finite arity (resp. at least one has infinite arity), regardless of the size of its universe.

In a different context, a *finitary* logic means one whose formulas and formal derivations all have finite length, otherwise the logic is said to be *infinitary*. Elsewhere, some authors have used the phrase *finitary mathematics* to mean mathematics that can be expressed without invoking infinite sets in any way. If you want to go deeper into the usages of "finite" versus "finitary", and "infinite" versus "infinitary", search the Web.

1.2 From Compactness in PL to Completeness in PL

We are now ready for the transition, from Compactness to Completeness. This is also an opportunity to present two fundamental concepts: the *Deduction Theorem* (here called a lemma) and *Consistency.*

Lemma 10 (Deduction Theorem). *Let $\Gamma \subseteq \mathsf{WFF}_{\mathsf{PL}}(\mathcal{P})$ and $\varphi, \psi \in \mathsf{WFF}_{\mathsf{PL}}(\mathcal{P})$, all arbitrary. It then holds that:* $\Gamma \vdash (\varphi \to \psi) \quad \Leftrightarrow \quad \Gamma \cup \{\varphi\} \vdash \psi$.

Exercise 11. Write the proof of Lemma 10.

Hint: Review the proof rules for PL in the Appendix C.1. This is a very easy exercise, especially when a formal proof is written as a natural deduction; all you need to consider are the two rules ($\to$I) and ($\to$E) and how they are used. □

A set of wff's $\Gamma \subseteq \mathsf{WFF}_{\mathsf{PL}}(\mathcal{P})$ is said to be *consistent* iff $\Gamma \nvdash \bot$. The set Γ is *inconsistent* iff it is not consistent, *i.e.*, $\Gamma \vdash \bot$. The next lemma is our first result connecting "$\vdash$" and "$\models$". Note how the earlier proof of Compactness (Theorem 2) dovetails with the next proof; it helps to understand the former before reading on.

Lemma 12. *Let $\Gamma \subseteq \mathsf{WFF}_{\mathsf{PL}}(\mathcal{P})$ be arbitrary. It then holds that if $\Gamma \models \bot$, then $\Gamma \vdash \bot$. In words, if Γ is unsatisfiable, then Γ is inconsistent.*

Proof. We prove the contrapositive: If $\Gamma \nvdash \bot$, then $\Gamma \not\models \bot$. In words, if Γ is *consistent*, then Γ is *satisfiable.*

First, observe that given an arbitrary wff φ, either $\Gamma \cup \{\varphi\}$ is consistent, or $\Gamma \cup \{\neg\varphi\}$ is consistent, or possibly both are consistent separately (but certainly not their union $\Gamma \cup \{\varphi, \neg\varphi\}$!). Indeed, if $\Gamma \cup \{\varphi\}$ is inconsistent, then $\Gamma \cup \{\varphi\} \vdash \bot$ so that, by Lemma 10, $\Gamma \vdash (\varphi \to \bot)$. From the latter judgment, we easily get the judgment $\Gamma \vdash \neg\varphi$ (review the proof rule ($\neg$I) in the Appendix C.1). Similarly, if $\Gamma \cup \{\neg\varphi\}$ is inconsistent, then $\Gamma \vdash \neg\neg\varphi$.

Hence, if both $\Gamma \cup \{\varphi\}$ and $\Gamma \cup \{\neg\varphi\}$ are inconsistent, then both $\Gamma \vdash \neg\varphi$ and $\Gamma \vdash \neg\neg\varphi$. The latter two judgments imply that Γ is inconsistent – contradicting our hypothesis that Γ is consistent. Hence, it cannot be that both $\Gamma \cup \{\varphi\}$ and $\Gamma \cup \{\neg\varphi\}$ are inconsistent, and at least one of the two is consistent.

As in the proof of Theorem 2, we consider a fixed, countably infinite, enumeration of all the formulas in $\mathsf{WFF}_{\mathsf{PL}}(\mathcal{P})$, say, $\varphi_1, \varphi_2, \varphi_3, \ldots$. We define a nested sequence of consistent sets:

$$\Gamma \stackrel{\text{def}}{=} \Delta_0 \subseteq \Delta_1 \subseteq \Delta_2 \subseteq \cdots, \quad \text{where}$$

$$\Delta_{i+1} \stackrel{\text{def}}{=} \begin{cases} \Delta_i \cup \{\varphi_{i+1}\} & \text{if } \Delta_i \cup \{\varphi_{i+1}\} \text{ is consistent,} \\ \Delta_i \cup \{\neg\varphi_{i+1}\} & \text{if } \Delta_i \cup \{\varphi_{i+1}\} \text{ is inconsistent,} \\ & \text{which implies that } \Delta_i \cup \{\neg\varphi_{i+1}\} \text{ is consistent,} \end{cases}$$

for every $i \geqslant 0$. We now define:

$$\Delta = \bigcup_i \Delta_i \qquad \text{(the limit of the } \Delta_i\text{'s)}$$

The set Δ is a *maximal consistent* set of wff's, in the sense that given an arbitrary wff φ, either $\varphi \in \Delta$ or $\neg\varphi \in \Delta$. In particular, for every propositional variable $p \in \mathcal{P}$, either $p \in \Delta$ or $\neg p \in \Delta$. We next define a truth assignment σ as follows:

$$\sigma(p_i) = \begin{cases} \mathit{true} & \text{if } p_i \in \Delta, \\ \mathit{false} & \text{if } \neg p_i \in \Delta. \end{cases}$$

As in the proof of Theorem 2, it is a straightforward exercise to show that σ satisfies a propositional wff φ iff $\varphi \in \Delta$. (See also Exercise 5.) Hence, $\sigma \models \Delta$ and, since $\Gamma \subseteq \Delta$, also $\sigma \models \Gamma$. Since Γ has a model σ and $\bot$ has no models, it follows that $\Gamma \not\models \bot$. □

Theorem 13 (Completeness for Propositional Logic)**.** *Let Γ be a set of propositional wff's (possibly infinite), and ψ a propositional wff. If $\Gamma \models \psi$, then $\Gamma \vdash \psi$.*

Proof. Suppose $\Gamma \models \psi$. From $\{\psi, \neg\psi\} \models \bot$, it follows that $\Gamma \cup \{\neg\psi\} \models \bot$. By Lemma 12, we thus have that $\Gamma \cup \{\neg\psi\} \vdash \bot$. It follows, by Lemma 10, that $\Gamma \vdash (\neg\psi \to \bot)$, which implies that $\Gamma \vdash \neg\neg\psi$ and again $\Gamma \vdash \psi$ (make sure you understand this last step, by reviewing the proof rules in the Appendix C.1). □

The crucial result in proving Completeness in the preceding theorem is Lemma 12, whose proof follows closely the steps of the proof of Compactness in Theorem 2. Note the interesting parallel: the proof of Theorem 2 builds a *maximal satisfiable* set, the proof of Lemma 12 builds a *maximal consistent* set.

There is another way of reaching Completeness from Compactness, through an intermediate result which we may call Weak Completeness and which does not depend on Compactness. The proof of Weak Completeness is far more involved technically than the proof of Lemma 12, and we only give an appropriate reference.[10]

Proposition 14 (Weak Completeness for Propositional Logic)**.** *Let $\varphi_1, \ldots, \varphi_n, \psi$ be propositional wff's. If $\varphi_1, \ldots, \varphi_n \models \psi$ then $\varphi_1, \ldots, \varphi_n \vdash \psi$.*

Proof. This lemma is called the Completeness Theorem in the book by M. Huth and M. Ryan [9], which is strictly weaker than our (and commonly called) Completeness Theorem. The proof is in Section 1.4.4 of that book; specifically, this is the left-to-right implication in Corollary 1.39. □

Nevertheless, to reach Completeness via Weak Completeness, it is not possible to bypass Compactness. This is the next (alternative) proof of Completeness. Compactness remains the linchpin.

Alternative Proof of Theorem 13. If $\Gamma \models \psi$, there is a finite $\Gamma_0 \subseteq \Gamma$ such that $\Gamma_0 \models \psi$, by Corollary 7. Hence, by Proposition 14, $\Gamma_0 \vdash \psi$. Padding Γ_0 with the redundant premises in $(\Gamma - \Gamma_0)$, we conclude $\Gamma \vdash \psi$. □

1.3 Applications and Exercises

All the examples and exercises in this section involve formal modeling using propositional logic. Many of these are applications of Compactness. The final result in each of these applications can certainly be obtained by other means, but this is not immediately obvious and may require some tricky combinatorics. Compactness provides an elegant alternative. When Compactness is invoked, most of the hard work is to formulate necessary and sufficient conditions for a solution in the form of an infinite set of propositional wff's; after which, satisfaction of those conditions is obtained by a relatively easy invocation of Compactness.

In several places below we use the notation "$\bigwedge$" and "$\bigvee$", which stand for using (finitely) many times the binary connectives "$\wedge$" and "$\vee$". For example, $\bigwedge\{\varphi_1, \varphi_2, \ldots, \varphi_k\}$ is a shorthand for the wff $\big((\varphi_1 \wedge \varphi_2) \cdots \wedge \varphi_k\big)$. The symbols "$\bigwedge$" and "$\bigvee$" are distinct from "$⩕$" and "$⩖$" in Exercise 8; the former are shorthands for using the binary "$\wedge$" and "$\vee$" finitely many times, the latter are not.

[10] Many of the technical complications are the result of our choice of a proof system in natural-deduction style. It is possible to choose and adjust another proof system, notably in Hilbert-style rather than in natural-deduction style, which reduces the technical overhead in the proof of Weak Completeness. We choose a natural-deduction style for many other benefits.

1.3.1 Graphs and Simple Graphs

We take a *simple graph* to be an undirected graph, without multiple edges, and without self loops. A simple graph $G \stackrel{\text{def}}{=} (V, E)$ is specified by its set of vertices V and set of edges E. We take E as a binary relation on V, which is also commutative (because G is undirected). With no loss of generality, we assume V to be a finite initial segment of the positive integers, $\{1, 2, \ldots, n\}$ for some $n \geqslant 1$ if G is finite, or the set of all positive integers $\{1, 2, \ldots\}$ if G is infinite.

In a simple graph $G \stackrel{\text{def}}{=} (V, E)$, we call the two endpoints of an edge *neighbors in* G, which are necessarily distinct. For every vertex $v \in V$ we define the set of neighbors of v in G as:

$$\mathsf{neighbors}_G(v) \stackrel{\text{def}}{=} \Big\{ w \in V \;\Big|\; \{v, w\} \in E \Big\},$$

which we extend to every subset of vertices $W \subseteq V$:

$$\mathsf{neighbors}_G(W) \stackrel{\text{def}}{=} \bigcup \Big\{ \mathsf{neighbors}_G(v) \;\Big|\; v \in W \Big\}.$$

If the graph G is understood from the context, we write $\mathsf{neighbors}(v)$ instead of $\mathsf{neighbors}_G(v)$.

Example 15 (*Cliques in Simple Graphs, I*)**.** A *clique* C in a simple graph G is a subset of vertices such that, between every two vertices in C, there is an edge. The *k-clique problem* asks whether a given simple graph $G \stackrel{\text{def}}{=} (V, E)$ contains a clique with k vertices. See Figure 1.1 for an example of a simple graph and its cliques.

In this example we assume G is a finite simple graph with $n \geqslant 1$ vertices. We want to write a propositional wff $\varphi(G)$ such that $\varphi(G)$ *is satisfied by G iff G contains a k-clique.* To model this problem in propositional logic, we may choose a set X of doubly-indexed propositional variables, with one variable for every two-element subset of $\{1, \ldots, n\}$:[11]

$$X \;\stackrel{\text{def}}{=}\; \Big\{ x_{i,j} \;\Big|\; 1 \leqslant i < j \leqslant n \Big\}.$$

To say that $\varphi(G)$ is satisfied by G means that $\varphi(G)$ is satisfied by the truth assignment induced by G, namely, by the assignment $\sigma_G : X \to \{\mathit{true}, \mathit{false}\}$ where:

$$\sigma_G(x_{i,j}) = \mathit{true} \text{ iff } \{i, j\} \in E \quad \text{and} \quad \sigma_G(x_{i,j}) = \mathit{false} \text{ iff } \{i, j\} \notin E.$$

Note that we make our desired propositional wff depend on G by parametrizing it with "G" and writing "$\varphi(G)$", *i.e.*, it is a function of a given finite G:

$$\varphi(G) \;\stackrel{\text{def}}{=} \bigvee \Big\{ \bigwedge \Big\{ x_{i,j} \;\Big|\; i, j \in K \text{ and } 1 \leqslant i < j \leqslant n \Big\} \;\Big|\; K \subseteq V \text{ and } \big|K\big| = k \Big\}$$

It is straightforward to justify (left to you) that $\varphi(G)$ is satisfied by G iff G contains a k-clique. One limitation is that $\varphi(G)$ works for finite simple graphs only. We cannot use $\varphi(G)$, nor is there an obvious generalization to another propositional wff – or set of propositional wff's – $\varphi^\star(G)$, which is satisfied by an arbitrary G (finite or infinite) iff G contains a k-clique. We need more expressive formal logics to be able to lift this limitation. □

Exercise 16 (*Cliques in Simple Graphs, II*)**.** Show that with the *Infinitary* PL defined in Exercise 8 we can lift the restriction to finite simple graphs of size $n \geqslant 1$ in Example 15. Specifically, for a fixed $k \geqslant 1$, show there is a single wff $\varphi(G)$ of the *Infinitary* PL such that for every simple graph G, finite or infinite, it is the case that $\varphi(G)$ *is satisfied by G iff G contains a k-clique.* □

[11] Note how we have indexed the variables in X: In the variable name "$x_{i,j}$" we have required that $i < j$, which we can do because there is no direction on the edge $\{i, j\}$ and there are no self-loops.

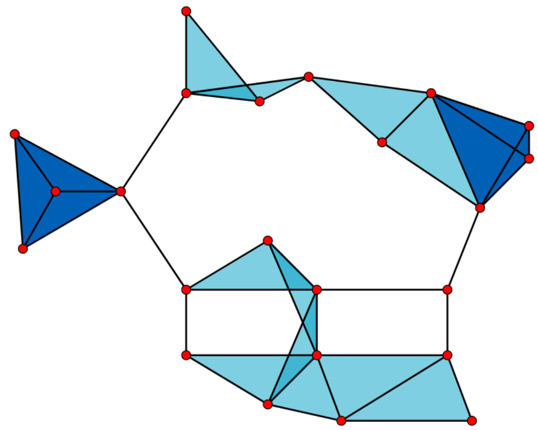

Figure 1.1: Courtesy of Wikipedia: A graph with: $23 \times$ 1-vertex cliques (the vertices), $42 \times$ 2-vertex cliques (the edges), $19 \times$ 3-vertex cliques (light and dark blue triangles), $2 \times$ 4-vertex cliques (dark blue areas). The 11 light blue triangles form *maximal* cliques. The two dark blue 4-cliques are both *maximum* and *maximal*.

Exercise 17 (*Cliques in Simple Graphs, III*)**.** This continues the examination of the clique problem in Example 15 and Exercise 16.

We say that a set Δ of propositional wff's is satisfied by a graph G, finite or infinite, iff for every $\varphi \in \Delta$, it is the case that φ is satisfied by G. The latter notion is defined in Example 15. There are two parts in this exercise:

1. Let $k \geqslant 1$ be fixed. Show that it is possible to define an infinite set $\Gamma(G)$ of propositional wff's over the set of variables X (as defined in Example 15) such that for every simple graph G, finite or infinite, it is the case that $\Gamma(G)$ *is satisfied by* G *iff* G *does* not *contain a* k*-clique.*

Thus, the *absence* of k-cliques in a graph G, finite or infinite, can be expressed by a set $\Gamma(G)$ of propositional wff's. Note the contrast with the conclusions in Example 15 and Exercise 16, where it is pointed out that the *presence* of k-cliques in an infinite graph G cannot be expressed in PL, whether by a single wff or by a set of wff's.

2. A graph is called *planar* if it can be drawn on the plane in such a way that no edges cross each other. Let G be an *infinite* planar graph. Show that for every $k \geqslant 5$, G does not contain a k-clique.

   Hint 1: If G does not contain a k-clique, then it does not contain a k'-clique for every $k' > k$. Given a *finite* planar graph G, it is known from graph theory that G does not contain any 5-cliques. The result from graph theory is stated and then proved for *finite* planar graphs, but not for *infinite* planar graphs. You are here asked to extend the result to infinite planar graphs.

   Hint 2: For the given infinite planar graph G, you need to show that every finite subset of $\Gamma(G)$ can be satisfied, and then invoke Compactness. □

Exercise 18 (*Dominating Sets in Simple Graphs, I*)**.** Given a finite graph G with set of vertices $V = \{1, \ldots, n\}$ and $1 \leqslant k \leqslant n$, the *$k$-dominating set problem* asks whether G contains a set K of k vertices such that all the vertices in $(V - K)$ have a neighbor in K.

Your task is to define a propositional wff $\varphi(G)$, and then justify its correctness, such that: $\varphi(G)$ *is satisfied by* G *iff* G *contains a* k*-dominating set.*

Hint: Review the setup introduced in Example 15 and what it means for a simple graph G to satisfy a propositional wff φ.

Further Hint: The k-clique problem in Example 15 written in plain text fits into the following pattern: ***there is*** a subset $K \subseteq V$, ***for every*** pair of vertices in K, ... which, when translated into propositional logic, fits into a pattern of the form: $\bigvee\{ \bigwedge\{ \cdots \mid \cdots \} \mid \cdots \}$.

Similarly, the k-dominating set problem, when written in plain text, fits into a pattern of the form: ***there is*** a subset $K \subseteq V$, ***for every*** vertex in $(V - K)$, ***there is*** an edge, ... This pattern, when translated into propositional logic, should have the form: $\bigvee\{ \bigwedge\{ \bigvee\{ \cdots \mid \cdots \} \mid \cdots \} \mid \cdots \}$. Note the correspondence between *existential* and *universal* quantifications, neither of which is available in propositional logic, and the connectives "$\bigvee$" and "$\bigwedge$", respectively. □

Exercise 19 (*Dominating Sets in Simple Graphs, II*)**.** In Exercise 18, the wff $\varphi(G)$ expresses the *presence* of a k-dominating set in a *finite* simple graph G. It is a limitation of propositional logic that we cannot lift the restriction to *finite* graphs, even if we use a set $\Phi(G)$ of wff's instead of a single wff $\varphi(G)$.

However, to express the *absence* of a k-dominating set in a a simple graph G, finite or infinite, propositional logic does provide us with the appropriate means. Once more, we assume that the set of vertices of a finite graph G is a proper initial segment of the positive integers $\{1, 2, \ldots\}$, otherwise if G is infinite, assume that its vertices are all the positive integers.

Your task is to define an infinite set $\Phi(G)$ of propositional wff's over the set X of variables (as defined in Example 15) such that for every simple graph G, finite or infinite, $\Phi(G)$ *is satisfied by* G *iff* G *does* not *contain a k-dominating set*. □

Example 20 (*Topological Sorting*)**.** A standard exercise in an undergraduate course on discrete algorithms is to show that every finite *directed acyclic graph* (dag) G can be *topologically sorted*, which means that the vertices of G can be linearly ordered on a horizontal line such that all the edges of G are drawn in the same direction, from left to right. We extend this result to infinite graphs: *Every infinite dag can be topologically sorted.*

Let $G \stackrel{\text{def}}{=} (V, E)$ be an infinite directed graph, where V is the set of vertices which we choose to name with the positive integers $\{1, 2, \ldots\}$, and $E \subseteq V \times V$ is the set of edges. For convenience, we use two sets of doubly-indexed propositional variables, $\mathcal{Q}$ and $\mathcal{R}$, instead of $\mathcal{P}$:

$$\mathcal{Q} \stackrel{\text{def}}{=} \left\{ q_{i,j} \,\middle|\, i, j \in \{1, 2, \ldots\} \right\} \quad \text{and} \quad \mathcal{R} \stackrel{\text{def}}{=} \left\{ r_{i,j} \,\middle|\, i, j \in \{1, 2, \ldots\} \right\}.$$

The propositional wff's in this example are in $\mathsf{WFF}_{\mathsf{PL}}(\mathcal{Q} \cup \mathcal{R})$. To facilitate our modeling of G's properties below, we purposely use names of vertices, such as i and j, as indices to identify variables $q_{i,j}$ and $r_{i,j}$. We consider initial finite fragments of the graph G, based on increasingly larger subsets of vertices:

$$V_1 \stackrel{\text{def}}{=} \{1\},\ V_2 \stackrel{\text{def}}{=} \{1, 2\}, \ldots,\ V_n \stackrel{\text{def}}{=} \{1, 2, \ldots, n\},\ \ldots \quad \text{so that also } V_1 \subseteq V_2 \subseteq \ \cdots \ \subseteq V_n \subseteq \cdots .$$

We write G_n for the finite subgraph of G induced by the vertices in V_n, *i.e.*, $G_n \stackrel{\text{def}}{=} (V_n, E_n)$ where $E_n = E \cap (V_n \times V_n)$. By this definition, G_n is a finite subgraph of $G_{n'}$, which is in turn a finite subgraph of the full graph G, for all $1 \leqslant n < n'$. Satisfaction of the following wff:

$$\pi_n(G) \stackrel{\text{def}}{=} \bigwedge \left\{ q_{i,j} \,\middle|\, (i,j) \in E_n \right\} \ \wedge\ \bigwedge \left\{ \neg q_{i,j} \,\middle|\, (i,j) \notin E_n \right\}$$

determines subgraph G_n up to isomorphism. This is so because $q_{i,j}$ is satisfied (*i.e.*, assigned truth-value *true*) iff there is an edge from i to j, and $\neg q_{i,j}$ is satisfied (*i.e.*, assigned truth-value *true*) iff there is no edge from i to j.

Said differently, satisfaction of $q_{i,j}$ corresponds to the assertion "there is a path of length $= 1$ from vertex i to vertex j". We want to use satisfaction of variable $r_{i,j}$ to model the more general assertion "there is a finite path of length $\geqslant 1$ from vertex i to vertex j". We thus define another wff $\rho_n(G)$, one for every $n \geqslant 1$, as:

$$\rho_n(G) \stackrel{\text{def}}{=} \bigwedge\left\{ q_{i,j} \to r_{i,j} \,\middle|\, (i,j) \in E_n \right\} \wedge \bigwedge\left\{ q_{i,j} \wedge r_{j,k} \to r_{i,k} \,\middle|\, (i,j) \in E_n \text{ and } k \in \{1, \ldots, n\} \right\}$$

Note carefully how the second part of $\rho_n(G)$ is defined. Informally in words, satisfaction of $r_{i,j}$ implies the existence of a path from i to j in G_n or, equivalently, the existence of an edge in the *transitive closure* of G_n.

We define one more wff θ_n, one for every $n \geqslant 1$, which does not depend on G and which models the assertion that "for one of the vertices i there is a path from i back to i in the transitive closure of G_n":

$$\theta_n \stackrel{\text{def}}{=} \bigvee \left\{ r_{i,i} \;\middle|\; i \in \{1, \ldots, n\} \right\}.$$

The wff θ_n is satisfied by G_n iff G_n contains a cycle. Hence, $\neg\theta_n$ is satisfied by G_n iff G_n is acyclic. Finally, consider the infinite set $\Gamma(G)$ of propositional wff's defined by:

$$\Gamma(G) \stackrel{\text{def}}{=} \left\{ \pi_n(G) \;\middle|\; n \geqslant 1 \right\} \cup \left\{ (\pi_n(G) \wedge \rho_n(G) \to \neg\theta_n) \;\middle|\; n \geqslant 1 \right\}.$$

If the full graph G is acyclic, then each of the finite subgraphs G_n is acyclic, which in turn implies that every finite subset of $\Gamma(G)$ is satisfiable (left to you to check carefully!). By Compactness, the full set $\Gamma(G)$ is satisfiable, which implies the full set of vertices $\{1, 2, \ldots, n, \ldots\}$ can be linearly ordered such that all the edges are drawn from left to right. □

Exercise 21 (*Vertex Coloring in Simple Graphs*)**.** All graphs $G \stackrel{\text{def}}{=} (V, E)$ in this exercise are simple, finite or infinite. Once more, we take the set V of vertices as an initial fragment of the positive integers $\{1, 2, \ldots\}$, finite or infinite, and its set of edges $E \subseteq V \times V$.

Such a graph G is said *k-colorable* if it is possible to assign only one of k colors to every vertex of G such that the two endpoints of every edge are assigned different colors. A famous (and very difficult to prove) result of graph theory is that *every finite planar graph is 4-colorable.* In this exercise you are asked to show that this result can be extended to *infinite* planar graphs using Compactness for PL.

Let $k \geqslant 1$ be fixed, the number of available colors. For convenience, use two separate sets of propositional variables, $\mathcal{Q}$ and $\mathcal{C}$, instead of $\mathcal{P}$:

$$\mathcal{Q} \stackrel{\text{def}}{=} \left\{ q_{i,j} \;\middle|\; i, j \in \{1, 2, \ldots\} \right\} \quad \text{and} \quad \mathcal{C} \stackrel{\text{def}}{=} \left\{ c_i^j \;\middle|\; i \in \{1, 2, \ldots\} \text{ and } 1 \leqslant j \leqslant k \right\}.$$

All wff's in this exercise should be in $\mathsf{WFF}_{\mathsf{PL}}(\mathcal{Q} \cup \mathcal{C})$. Use variables in $\mathcal{Q}$ to model a given graph G: there is an edge connecting two distinct vertices i and j iff $q_{i,j}$ is set to truth value *true*. Use variables in $\mathcal{C}$ to model G's coloring: vertex i is assigned color $j \in \{1, \ldots, k\}$ iff c_i^j is set to truth value *true*. There are three parts in this exercise, the first two are not limited to planar graphs:

1. Let $G \stackrel{\text{def}}{=} (V, E)$ be a finite simple graph. Write a wff $\varphi(G)$ which is satisfied iff G is k-colorable.

   Hint: Define $\varphi(G)$ in two parts, one part specifies the structure of G (including conditions that there are no self-loops and that G is undirected), and one part specifies that G is k-colorable. You will find it useful to review Example 20.

2. Let $G \stackrel{\text{def}}{=} (V, E)$ be an infinite simple graph, with $V = \{1, 2, \ldots\}$. Show that G is k-colorable iff every finite subgraph of G is k-colorable.

   Hint: If an infinite G is k-colorable, then so is every finite subgraph of G. This is the easy implication. Compactness for PL should give you the converse, after writing an infinite set $\Phi(G)$ of wff's in $\mathsf{WFF}_{\mathsf{PL}}(\mathcal{Q} \cup \mathcal{C})$ expressing the k-colorability of G.

3. Let $G \stackrel{\text{def}}{=} (V, E)$ be an infinite planar graph, with $V = \{1, 2, \ldots\}$. Invoke the preceding part to conclude that G is 4-colorable. □

Exercise 22 (*Edge Coloring in Simple Graphs*)**.** This continues the examination in Exercise 21. An edge coloring of a graph $G \stackrel{\text{def}}{=} (V, E)$ is an assignment of colors to the edges of G such that no two adjacent edges, *i.e.*, two edges that share an endpoint, have the same color.

As in the preceding exercise, let $k \geqslant 1$ be fixed, the number of available colors. We use two separate sets of propositional variables, $\mathcal{Q}$ and $\mathcal{C}$:

$$\mathcal{Q} \stackrel{\text{def}}{=} \left\{ q_{i,j} \;\middle|\; i,j \in \{1,2,\ldots\} \right\} \quad \text{and} \quad \mathcal{C} \stackrel{\text{def}}{=} \left\{ c^{\ell}_{i,j} \;\middle|\; i,j \in \{1,2,\ldots\} \text{ and } 1 \leqslant \ell \leqslant k \right\}.$$

Use variables in $\mathcal{Q}$ to model a given graph G: there is an edge connecting two distinct vertices i and j iff $q_{i,j}$ is set to truth value *true*. Use variables in $\mathcal{C}$ to model G's edge coloring: edge $\{i,j\}$ is assigned color $\ell \in \{1,\ldots,k\}$ iff $c^{\ell}_{i,j}$ is set to truth value *true*. There are two parts:

1. Let $G \stackrel{\text{def}}{=} (V,E)$ be a finite simple graph. Write a wff $\varphi(G)$ which is satisfied iff the edges of G are k-colorable.

   Hint: Define $\varphi(G)$ in two parts, one part specifies the structure of G (including conditions that there are no self-loops and that G is undirected), and one part specifies that the edges of G are k-colorable. For the first of these two parts, you will find it helpful to read Example 20.

2. Let $G \stackrel{\text{def}}{=} (V,E)$ be an infinite simple graph, with $V = \{1,2,\ldots\}$. Show that the edges of G are k-colorable iff for every finite subgraph of G its edges are k-colorable.

   Hint: If the edges of G are k-colorable, then trivially the edges of every finite subgraph of G are k-colorable. Compactness for PL should give you the converse implication, after writing an infinite set $\Phi(G)$ of wff's in $\mathsf{WFF}_{\mathsf{PL}}(\mathcal{Q} \cup \mathcal{C})$ expressing that the edges of G are k-colorable. □

Exercise 23 (*Regular Graphs*)**.** A simple graph $G \stackrel{\text{def}}{=} (V,E)$ is *regular* if every vertex has the same number of neighbors. A regular simple graph for which there is a fixed $d \in \mathbb{N}$ such that every vertex has d neighbors is called *d-regular*.

All the wff's in this exercise should be in $\mathsf{WFF}_{\mathsf{PL}}(X)$ where X is the following set of propositional variables:

$$X \stackrel{\text{def}}{=} \left\{ x_{i,j} \;\middle|\; i,j \in V \right\}.$$

Variable $x_{i,j}$ will be assigned truth-value *true* (resp. *false*) if $\{i,\,j\} \in E$ (resp. $\{i,\,j\} \notin E$). There are four parts in this exercise, the first two are very easy as they are about expressibility of 0-regularity in PL, which is the case of graphs consisting entirely of isolated vertices:

1. Let $G \stackrel{\text{def}}{=} (V,E)$ be finite. Write a wff $\varphi_0(G) \in \mathsf{WFF}_{\mathsf{PL}}(X)$ which is satisfied by G iff G is 0-regular.

2. Let $G \stackrel{\text{def}}{=} (V,E)$ be infinite. Write a set of wff's $\Phi_0(G) \subseteq \mathsf{WFF}_{\mathsf{PL}}(X)$ which is satisfied by G iff G is 0-regular.

3. Let $G \stackrel{\text{def}}{=} (V,E)$ be finite and $d \geqslant 1$. Write a wff $\varphi_d(G) \in \mathsf{WFF}_{\mathsf{PL}}(X)$ which is satisfied by G iff G is d-regular.

   Hint: We give the solution for the case $d = 1$ which you can take as a guide to define the solution for an arbitrary $d \geqslant 1$:

   $$\varphi_1(G) \stackrel{\text{def}}{=} \bigwedge \left\{ \bigvee \left\{ x_{i,j} \wedge \bigwedge \left\{ \neg x_{i,k} \;\middle|\; k \in V - \{i,j\} \right\} \;\middle|\; j \in \mathsf{neighbors}_G(i) \right\} \;\middle|\; i \in V \right\}$$

   Note how the nesting of "$\bigwedge$" and "$\bigvee$" in: $\bigwedge\{\bigvee\{x_{i,j} \wedge \bigwedge\{\cdots \mid \cdots\} \mid \cdots\} \mid \cdots\}$ translates the following pattern in plain text:

   for every vertex i ***there is*** a vertex j such that $\{i,j\}$ is an edge and ***for every*** vertex k . . . .

   "$\bigwedge$" and "$\bigvee$" correspond to the *universal* and *existential* quantifiers, respectively, which are not available in propositional logic.

4. Let $G \stackrel{\text{def}}{=} (V,E)$ be infinite, $d \geqslant 1$, and $\mathsf{neighbors}_G(i)$ be finite for every $i \in V$. Show that there is a set of wff's $\Phi_d(G) \subseteq \mathsf{WFF}_{\mathsf{PL}}(X)$ which is satisfied by G iff G is d-regular. □

Exercise 24 (*Matchings in Simple Graphs*)**.** A *matching* in a simple graph $G \stackrel{\text{def}}{=} (V, E)$ is a subset $M \subseteq E$ of the edges such that no two edges in M share an endpoint. With M being a set of edges, *i.e.*, two-element sets, it makes sense that we can take the union $\bigcup M$, which is the set of vertices said to be *covered* by M. The matching M is said to be *perfect* if it covers all the vertices in V, which may or may not exist. In particular, if M is perfect, then $\lvert V \rvert$ is even; if $\lvert V \rvert$ is odd, at most $\lvert V \rvert - 1$ vertices can be covered by a maximum-size matching M, in which case M is called *near-perfect*.

In this exercise we use two separate sets of propositional variables, $\mathcal{Q}$ and $\mathcal{Y}$:

$$\mathcal{Q} \stackrel{\text{def}}{=} \left\{ q_{i,j} \;\middle|\; i, j \in \{1, 2, \ldots\} \right\} \quad \text{and} \quad \mathcal{Y} \stackrel{\text{def}}{=} \left\{ y_{i,j} \;\middle|\; i, j \in \{1, 2, \ldots\} \right\}.$$

We use variables in $\mathcal{Q}$ to model a given simple graph $G \stackrel{\text{def}}{=} (V, E)$: there is an edge connecting two distinct vertices i and j iff $q_{i,j}$ is set to truth value *true*. We use variables in $\mathcal{Y}$ to model a matching M in G, *i.e.*, a selection of subset of E: edge $\{i, j\}$ is selected by M iff $y_{i,j}$ is set to truth value *true*.

All the wff's in this exercise should be in $\mathsf{WFF}_{\mathsf{PL}}(\mathcal{Q} \cup \mathcal{Y})$. There are four parts:

1. Let $G \stackrel{\text{def}}{=} (V, E)$ be a finite simple graph such that $\lvert V \rvert$ is even. Write a wff $\varphi(G)$ which is satisfied iff there is a perfect matching M in G.

   Hint: Define $\varphi(G)$ in two parts, one part specifies the structure of G (including conditions that there are no self-loops and that G is undirected), and one part specifies that there is a matching in G. For the first of these two subparts, you will find it useful to review Example 20.

2. Let $G \stackrel{\text{def}}{=} (V, E)$ be infinite and $\mathsf{neighbors}_G(i)$ be finite for every $i \in V$. Show that there is a set of wff's $\Phi(G) \subseteq \mathsf{WFF}_{\mathsf{PL}}(\mathcal{Q} \cup \mathcal{Y})$ which is satisfied by G iff there is a perfect matching M in G.

3. Let $G \stackrel{\text{def}}{=} (V, E)$ be infinite and $\mathsf{neighbors}_G(i)$ be finite for every $i \in V$. Use the set $\Phi(G)$ from the preceding part and Compactness for PL to show that there is a perfect matching in G iff there is a perfect matching in every finite subgraph of G of even size.

4. Show that the restriction that $\mathsf{neighbors}_G(i)$ is finite in the preceding part is essential. More precisely, define an infinite simple graph G which does not have a perfect matching, even though every finite subgraph of G of even size has a perfect matching. □

Example 25 (*Hall's Marriage Theorem, I*)**.** A famous result of discrete mathematics is *Hall's Marriage Theorem*, which has several equivalent formulations. We here use the graph-theoretic formulation in terms of bipartite graphs.

A simple graph $G \stackrel{\text{def}}{=} (V, E)$ is *bipartite* if V can be partitioned into two subsets, L and R, of left vertices and right vertices, respectively, so that $V = L \cup R$, $L \cap R = \varnothing$, and $E \subseteq L \times R$.

Hall's Marriage Theorem is usually limited to *finite* bipartite graphs. In fact, though it is not explicitly stated, it suffices to restrict the set L of left vertices to be finite, with no restriction on the set R of right vertices, which may or may not be finite. Given such a bipartite graph G where L is finite, the theorem asserts that there is a matching that covers every vertex in L iff the following condition is satisfied:

Hall's Condition *For every subset $L_0 \subseteq L$, where L is finite, it holds that* $\lvert L_0 \rvert \leqslant \lvert \mathsf{neighbors}(L_0) \rvert$; in words, the vertices in every $L_0 \subseteq L$ are connected to at least as many vertices in R.

The proof of the theorem is by strong induction on finite cardinality of L – which is standard in introductory books on graph theory (left to you to consult). For later reference, we write $\mathbf{HC}(G)$ for *Hall's Condition* relative to a bipartite graph G with a finite set L of left vertices.

For ease of presentation we use metavariables i and j to range over vertices in L, and metavariables k and ℓ to range over vertices in R. We use two disjoint sets of propositional variables, $\mathcal{Q}$ and $\mathcal{Y}$:

$$\mathcal{Q} \stackrel{\text{def}}{=} \left\{ q_{i,k} \;\middle|\; i \in L,\ k \in R \right\} \quad \text{and} \quad \mathcal{Y} \stackrel{\text{def}}{=} \left\{ y_{i,k} \;\middle|\; i \in L,\ k \in R \right\}.$$

We use variables in $\mathcal{Q}$ to formally model the graph G: there is an edge connecting two distinct vertices $i \in L$ and $k \in R$ iff $q_{i,k}$ is set to truth value *true*. We use variables in $\mathcal{Y}$ to formally model a matching M in G, *i.e.*, a selection of subset of E: edge $\{i, k\}$ is selected by M iff $y_{i,k}$ is set to truth value *true*.

Our first task below is to define a propositional wff $\varphi(G)$ which is satisfiable iff there is a matching in G that covers every vertex in L. Our desired wff depends on a given G, which is why we parametrize it with "G" and write "$\varphi(G)$", which will be an expression in $\mathsf{WFF}_{\mathsf{PL}}(\mathcal{Q} \cup \mathcal{Y})$. Such a matching must satisfy three constraints:

1. *No two distinct left vertices i and j are matched with the same right vertex k*, which we can formalize with the set Δ_1 of propositional wff's:

$$\Delta_1 \stackrel{\text{def}}{=} \left\{ \neg(y_{i,k} \wedge y_{j,k}) \;\middle|\; i, j \in L,\ i \neq j,\ k \in \mathsf{neighbors}(i) \cap \mathsf{neighbors}(j) \right\}.$$

2. *No left vertex i is matched with two distinct right vertices k and ℓ*, which we can formalize with the set Δ_2 of propositional wff's:

$$\Delta_2 \stackrel{\text{def}}{=} \left\{ \neg(y_{i,k} \wedge y_{i,\ell}) \;\middle|\; i \in L,\ k, \ell \in \mathsf{neighbors}(i),\ k \neq \ell \right\}.$$

3. *Every left vertex i is matched with at least one right vertex k*, which we can formalize with the set Δ_3 of propositional wff's:

$$\Delta_3 \stackrel{\text{def}}{=} \left\{ \bigvee \{\, y_{i,k} \mid k \in \mathsf{neighbors}_G(i) \,\} \;\middle|\; i \in L \right\}.$$

The highlighted expression in Δ_3 is not in general a propositional wff unless the following condition holds:

($\diamondsuit$) For every $i \in L$ the set $\mathsf{neighbors}_G(i)$ is finite,

which makes $\bigvee \{y_{i,k} \mid k \in \mathsf{neighbors}_G(i)\}$ a finite disjunction of variables and, thus, a wff of PL. Of course, if G is finite, then condition ($\diamondsuit$) trivially holds. However, we do not want to impose size restrictions beyond those necessary for a formalization of our problem in PL.

If the set L of left vertices is finite and ($\diamondsuit$) holds, then each set in $\{\Delta_1, \Delta_2, \Delta_3\}$ is also finite, in which case we can define the desired wff $\varphi(G)$ as:

$$\varphi(G) \stackrel{\text{def}}{=} (\bigwedge \Delta_1) \wedge (\bigwedge \Delta_2) \wedge (\bigwedge \Delta_3).$$

From the preceding analysis, it follows that:

(a) If G is a bipartite graph with a finite set L of left vertices which also satisfies ($\diamondsuit$), then there is a matching in G that covers every vertex in L iff wff $\varphi(G)$ is satisfiable.

Hence, by *Hall's Marriage Theorem*, we also have the following result:

(b) If G is a bipartite graph with a finite set L of left vertices which also satisfies ($\diamondsuit$), then condition $\mathbf{HC}(G)$ holds iff wff $\varphi(G)$ is satisfiable.

Instead of working with a single wff $\varphi(G)$, we can also work with the set $\Phi(G)$ of propositional wff's defined by:

$$\Phi(G) \stackrel{\text{def}}{=} \Delta_1 \cup \Delta_2 \cup \Delta_3.$$

One benefit of working with the set $\Phi(G)$ instead of the single wff $\varphi(G)$ is that we do not need to restrict the set L to be finite. If L is infinite then so is $\Phi(G)$ an infinite set of propositional wff's, provided ($\diamondsuit$) still holds. We thus have the following result:

(c) If G is a bipartite graph, possibly with an infinite set L of left vertices, which satisfies ($\diamondsuit$), then there is a matching in G that covers every vertex in L iff the set $\Phi(G)$ is satisfiable.

Hence, by Compactness for PL, we also have:

(d) If G is a bipartite graph, possibly with an infinite set L of left vertices, which satisfies ($\diamondsuit$), then there is a matching in G that covers every vertex in L iff every finite subset $\Gamma \subseteq \Phi(G)$ is satisfiable.

Result (d) above involves PL, namely, the satisfaction of every finite subset $\Gamma \subseteq \Phi(G)$.

In Exercise 26 to follow, you are asked to continue the preceding analysis to reach a conclusion not involving any PL, which states a necessary and sufficient condition for the existence of a matching in a bipartite G that covers every vertex in L, possibly infinite, provided condition ($\diamondsuit$) holds in G. $\square$

Exercise 26 (*Hall's Marriage Theorem, II*)**.** This exercise is a continuation of Example 25, with the same notation and conventions. In particular, $G \stackrel{\text{def}}{=} (V, E)$ is a bipartite simple graph, with L and R as its sets of left and right vertices, respectively. The set L is now arbitrary, finite or infinite.

If $L_0 \subseteq L$ is a finite set of left vertices in G, let $G[L_0 \cup \mathsf{neighbors}_G(L_0)]$ denote the subgraph of G induced by the set of vertices:

$$L_0 \cup \Big(\bigcup\{\mathsf{neighbors}_G(i) \mid i \in L_0\}\Big) \;=\; L_0 \cup \mathsf{neighbors}_G(L_0).$$

If condition ($\diamondsuit$) holds in G, then $L_0 \cup \mathsf{neighbors}_G(L_0)$ is a finite set of vertices and the induced subgraph $G[L_0 \cup \mathsf{neighbors}_G(L_0)]$ is finite.

There are two parts in this exercise:

1. Given a bipartite simple graph G, with its set L of left vertices possibly infinite, which satisfies condition ($\diamondsuit$), your task is to prove the following conclusion:

   there is a matching in G that covers every vertex in L iff
   for every finite $L_0 \subseteq L$, it holds that $\boldsymbol{HC}(G[L_0 \cup \mathsf{neighbors}_G(L_0)])$.

   Hint: Read the analysis in Example 25 carefully. You need to continue from where it stops at assertion (d).

2. Show, with an appropriate counter-example, that if condition ($\diamondsuit$) is omitted, then the conclusion you proved in part 1 of this exercise fails. $\square$

1.3.2 Boards and Grids

Below are several examples and exercises on boards and grids whose solutions involve formal PL modeling. Before reading on, especially the parts involving infinite boards and grids, try to familiarize yourself with the material in Appendix G.

Exercise 27 (*Queens Problem*)**.** The *n-Queens Problem* is the problem of placing n queens on an $n \times n$ chessboard so that no two queens can attack each other. A solution of the problem when $n = 6$ is shown on the left of Figure 1.2. In this exercise we specify the requirements of a solution for the n-Queens Problem as a propositional wff ψ_n, with one such wff for every $n \geqslant 4$. (There are no solutions for $n = 2$ and $n = 3$.) For convenience, we use a set $\mathcal{Q}$ of doubly-indexed propositional variables, instead of $\mathcal{P}$, where the indices range over the positive integers:

$$\mathcal{Q} \stackrel{\text{def}}{=} \Big\{\, q_{i,j} \;\Big|\; i, j \in \{1, 2, \dots\} \,\Big\}.$$

The desired wff ψ_n in this exercise is in $\mathsf{WFF}_{\mathsf{PL}}(\mathcal{Q})$. We set the variable $q_{i,j}$ to truth value *true* (resp. *false*) if there is (resp. there is not) a queen placed in position (i, j) of the board, where we take the first index i (resp. the second index j) to range over the vertical axis downward (resp. the horizontal axis rightward); that is, i is a row number and j is a column number.[12] There are four parts in this exercise:

1. Write the wff ψ_n and justify how it accomplishes its task.

   Hint: Write ψ_n as a conjunction $\psi_n^{\text{row}} \wedge \psi_n^{\text{col}} \wedge \psi_n^{\text{diag1}} \wedge \psi_n^{\text{diag2}}$, where:

   (a) ψ_n^{row} is satisfied iff there is exactly one queen in each row,

   (b) ψ_n^{col} is satisfied iff there is exactly one queen in each column,

   (c) ψ_n^{diag1} is satisfied iff there is at most one queen in each diagonal,

   (d) ψ_n^{diag2} is satisfied iff there is at most one queen in each antidiagonal.

   Further Hint: Given any two distinct positions (i_1, j_1) and (i_2, j_2) along a diagonal, it is always the case that $i_1 - j_1 = i_2 - j_2$. And if the two positions are along an antidiagonal, then it is always the case that $i_1 + j_1 = i_2 + j_2$.

2. Imagine now an infinite chessboard, which occupies the entire south-east quadrant of the Cartesian plane. The coordinates along the vertical and horizontal axes are, respectively, i (increasing downward) and j (increasing rightward), both ranging over the positive integers $\{1, 2, \ldots\}$. In an attempt to repeat the argument in Example 20 and Exercise 21, someone once defined the set of wff's $\Gamma \overset{\text{def}}{=} \{\, \psi_n \mid n \geqslant 4 \,\}$, and wrote the following (in outline here):

   > The set Γ is finitely satisfiable and, therefore, satisfiable by Compactness. Hence, there exists a solution of the *Infinite Queens Problem*, which satisfies conditions $\{$(a), (b), (c), (d)$\}$ for all $n \geqslant 4$.

   What is wrong with the preceding argument? The answer is subtle and you need to be careful.

3. Your task is to define an infinite set $\Theta \overset{\text{def}}{=} \{\, \theta_k \mid k \geqslant 1 \,\}$ of distinct propositional wff's such that:

   (a) For every $k \geqslant 1$ there is $n \geqslant 1$ such that satisfaction of wff θ_k implies satisfaction of wff ψ_n defined in part 1 of this exercise, *i.e.*, satisfaction of θ_k defines a solution of the n-Queens Problem.

   (b) For all $k' > k \geqslant 1$, if satisfaction of wff's $\theta_{k'}$ and θ_k define solutions of the n'-Queens Problem and n-Queens Problem, respectively, then $n' > n$.

   (c) Every finite subset of Θ is satisfiable.

   Hint: For part (a), make θ_k define a particular n-Queens Problem, *i.e.*, θ_k is satisfied by exactly one truth assigment of the variables occurring in θ_k. For (b) and (c), read and understand the subsection entitled "*A second solution of the infinite Queens Problem*" in Appendix G.

4. Let Θ be the infinite set of wff's defined in the preceding part. Use Compactness for PL to give a rigorous argument that the *Infinite Queens Problem* has indeed a solution. $\square$

Exercise 28 (*The Queens Problem, Encoded Differently*)**.** Read the introductory paragraph in Exercise 27 before you start on this one. Our formal modeling of the *n-Queens Problem* here is a little different. The encoding is perhaps less natural, but involves less manipulation of the indices and easily lends itself to the case of an infinite chessboard forever extending south and

[12] This is the standard convention of identifying rows and columns in a two-dimensional matrix, which is not how we usually view the coordinates of the Cartesian plane, where the first coordinate is along the horizontal axis (going rightward) and the second coordinate is along the vertical axis (going upward). See Figure 1.2 for our convention. Also, following the conventions of two-dimensional matrices, a *diagonal* is a (-45^o)-*diagonal* directed downward starting from the west or north edge, and an *antidiagonal* is a $(+45^o)$-*diagonal* directed upward starting from the west or south edge.

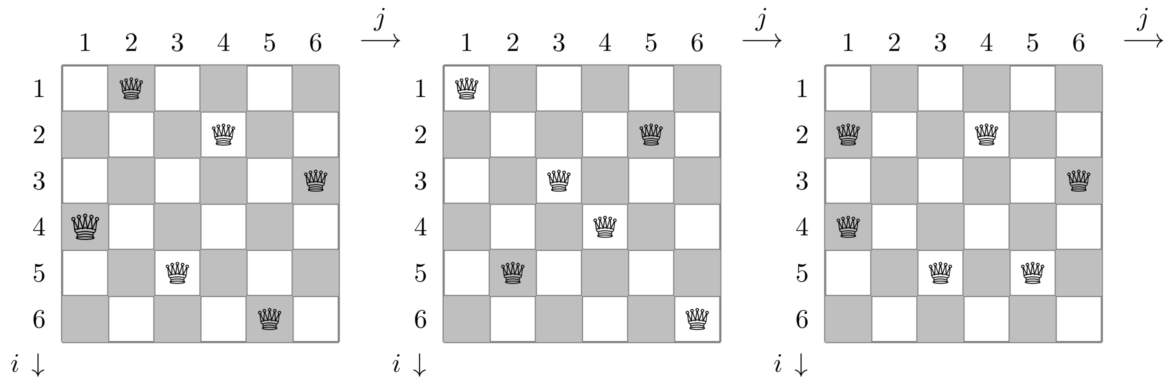

Figure 1.2: The 6-Queens Problem in Exercise 27: a solution *on the left*, a non-solution *in the middle* (satisfying conditions (a) and (b) only), a non-solution *on the right* (satisfying conditions (c) and (d) only).

east. We use four disjoint sets of propositional variables, the first of which is the same as in the previous exercise:

- $\mathcal{Q} \stackrel{\text{def}}{=} \left\{ q_{i,j} \;\middle|\; i,j \in \{1,2,3,\ldots\} \right\}$ where indices i and j refer to rows and columns, respectively;
- $\mathcal{R} \stackrel{\text{def}}{=} \left\{ r_i \;\middle|\; i \in \{1,2,3,\ldots\} \right\}$ where index i refers to $(+45^o)$-diagonals;
- $\mathcal{S} \stackrel{\text{def}}{=} \left\{ s_i \;\middle|\; i \in \{1,2,3,\ldots\} \right\}$ where index i refers to *lower* (-45^o)-diagonals;
- $\mathcal{T} \stackrel{\text{def}}{=} \left\{ t_i \;\middle|\; i \in \{1,2,3,\ldots\} \right\}$ where index i refers to *upper* (-45^o)-diagonal.

The numbers of $(+45^o)$-diagonals, lower (-45^o)-diagonals, and upper (-45^o)-diagonals, are respectively: $(2n-1)$, n, and $(n-1)$. For the case $n = 6$, Figure 1.3 is a graphic explanation. The wff's in this exercise are all in $\mathsf{WFF}_{\mathsf{PL}}(\mathcal{Q} \cup \mathcal{R} \cup \mathcal{S} \cup \mathcal{T})$. There are three parts in this exercise:

1. Write the wff ψ_n and justify how it accomplishes its task.

   Hint: Write ψ_n as a conjunction $\psi_n^{\text{row}} \wedge \psi_n^{\text{col}} \wedge \psi_n^{\text{diag1}} \wedge \psi_n^{\text{diag2}}$, where:

   (a) ψ_n^{row} is satisfied iff there is exactly one queen in each row,
   where you are allowed to use variables in $\mathcal{Q}$ only,

   (b) ψ_n^{col} is satisfied iff there is exactly one queen in each column,
   where you are allowed to use variables in $\mathcal{Q}$ only,

   (c) ψ_n^{diag1} is satisfied iff there are n queens on n $(+45^o)$-diagonals,
   where you are requested to use variables in $\mathcal{R}$ and minimize the use of those in $\mathcal{Q}$.

   (d) ψ_n^{diag2} is satisfied iff there are n queens on n (-45^o)-diagonals,
   where you are requested to use variables in $\mathcal{S} \cup \mathcal{T}$ and minimize the use of those in $\mathcal{Q}$.

   This *Hint* is reproduced from Exercise 27, but now you have to use the propositional variables in $\mathcal{Q} \cup \mathcal{R} \cup \mathcal{S} \cup \mathcal{T}$. And the *Further Hint* in Exercise 27 cannot be used anymore! Keep in mind that there are $(2n-1)$ $(+45^o)$-diagonals and $(2n-1)$ (-45^o)-diagonals.

2. Repeat the definition of the wff ψ_n but now you are not allowed to use variables in $\mathcal{Q}$, *i.e.*, your wff should be an expression in $\mathsf{WFF}_{\mathsf{PL}}(\mathcal{R} \cup \mathcal{S} \cup \mathcal{T})$.

   Hint: There is a queen on the i-th row iff the board satisfies the disjunction:

$$\bigvee \Big\{ \underbrace{(r_i \wedge s_i),\ (r_{i+1} \wedge s_{i-1}),\ \ldots,\ (r_{2i-1} \wedge s_1)}_{i \text{ conjuncts}},\ \underbrace{(r_{2i} \wedge t_1),\ (r_{2i+1} \wedge t_2),\ \ldots,\ (r_{i+n-1} \wedge t_{n-i})}_{n-i \text{ conjuncts}} \Big\}$$

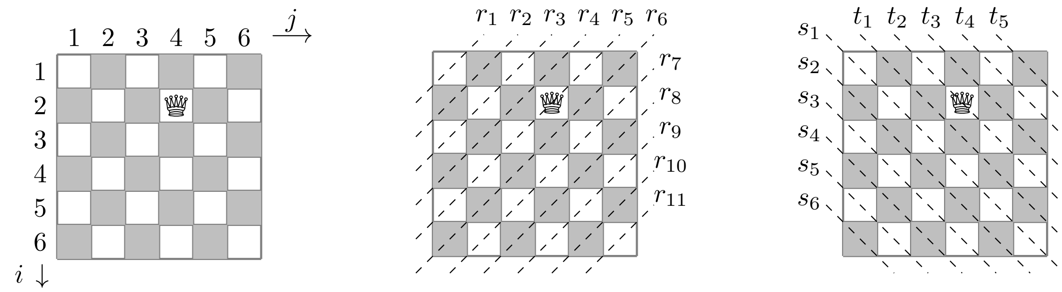

Figure 1.3: Chessboard of 6-Queen Problem in Exercise 28. Starting from the north-west corner, indices i and j of variable $q_{i,j}$ identify rows and columns (*on the left*). Variables r_i identify $(+45^o)$-diagonals (*in the middle*), variables s_i and t_i identify *lower* and *upper* (-45^o)-diagonals respectively (*on the right*). When there is a single queen on the chessboard as shown above, we have the following truth assignment: variables $q_{2,4}$, r_5, and t_2 are *true*, all the other variables are *false*.

There is a queen on the j-th column, $2 \leqslant j \leqslant n$, iff the board satisfies the disjunction:

$$\bigvee\Big\{\underbrace{(r_j \wedge t_{j-1}),\ (r_{j+1} \wedge t_{j-2}), \ldots,\ (r_{2j-2} \wedge t_1)}_{j-1 \text{ conjuncts}},\ \underbrace{(r_{2j-1} \wedge s_1),\ (r_{2j} \wedge s_2), \ldots, (r_{j+n-1} \wedge s_{n-j+1})}_{n-j+1 \text{ conjuncts}}\Big\}$$

The preceding disjunction has to be adjusted for the case $j = 1$, when there is a queen on the first column (left to you). Make sure you understand how variable indices are set.

3. Use Compactness to give a rigorous non-formal proof that the *Infinite Queens Problem* has a solution. (No credit if your argument does not invoke Compactness for PL.) □

Example 29 (*Queens&Rooks Problem, I*)**.** Read the definition of the *Queens&Rooks Problem* in Appendix G, in particular Section G.4, before you embark on this example.

We want to specify the requirements of a solution for the problem as a propositional wff ψ_n, with one such wff for every $n \geqslant 3$. (There are no solutions for $n \leqslant 2$.) For convenience, we use two sets of doubly-indexed propositional variables, $\mathcal{Q}$ and $\mathcal{R}$, where the indices range over the positive integers:

$$\mathcal{Q} \stackrel{\text{def}}{=} \Big\{ q_{i,j} \,\Big|\, i, j \in \{1, 2, \ldots\} \Big\} \quad \text{and} \quad \mathcal{R} \stackrel{\text{def}}{=} \Big\{ r_{i,j} \,\Big|\, i, j \in \{1, 2, \ldots\} \Big\}.$$

The desired wff ψ_n is in $\mathsf{WFF}_{\mathsf{PL}}(\mathcal{Q} \cup \mathcal{R})$. We set the variable $q_{i,j}$ to truth value *true* (resp. *false*) if there is (resp. there is not) a queen placed in position (i, j) of the board. And similarly, the variable $r_{i,j}$ is set to truth value *true* (resp. *false*) if there is (resp. there is not) a rook placed in position (i, j) of the board.

In this example, we select an initial random subset of p rows on which a queen is placed. Let $I \subseteq \{1, \ldots, n\}$ be a randomly selected subset with $|I| = p \leqslant n$. We specify several constraints that are satisfied by any solution σ, relative to this initial selection I of rows:

1. There is a *queen* on each of the p selected rows:

$$\varphi_{n,1} \stackrel{\text{def}}{=} \bigwedge\Big\{ \bigvee\{\, q_{i,j} \mid 1 \leqslant j \leqslant n \,\} \,\Big|\, i \in I \Big\}.$$

2. There is a *rook* on each of the remaining $(n - p)$ rows:

$$\varphi_{n,2} \stackrel{\text{def}}{=} \bigwedge\Big\{ \bigvee\{\, r_{i,j} \mid 1 \leqslant j \leqslant n \,\} \,\Big|\, i \in \{1, \ldots, n\} - I \Big\}.$$

3. No position on the board contains both a *queen* and a *knight*:

$$\varphi_{n,3} \stackrel{\text{def}}{=} \bigwedge\Big\{ q_{i,j} \to \neg r_{i,j} \,\Big|\, 1 \leqslant i, j \leqslant n \Big\}.$$

4. On every row i, if there is a *queen* in column j, then there is no *rook* and no other *queen* on row i:

$$\varphi_{n,4} \stackrel{\text{def}}{=} \bigwedge\Big\{ q_{i,j} \to \bigwedge\{\neg q_{i,\ell} \wedge \neg r_{i,\ell} \,\big|\, 1 \leqslant \ell \leqslant n \text{ and } \ell \neq j\} \;\Big|\; 1 \leqslant i,j \leqslant n \Big\}.$$

5. On every column j, if there is a *queen* in row i, then there is no *rook* and no other *queen* on column j:

$$\varphi_{n,5} \stackrel{\text{def}}{=} \bigwedge\Big\{ q_{i,j} \to \bigwedge\{\neg q_{k,j} \wedge \neg r_{k,j} \,\big|\, 1 \leqslant k \leqslant n \text{ and } k \neq i\} \;\Big|\; 1 \leqslant i,j \leqslant n \Big\}.$$

6. On every row i, if there is a *rook* in column j, then there is no *queen* and no other *rook* on row i:

$$\varphi_{n,6} \stackrel{\text{def}}{=} \bigwedge\Big\{ r_{i,j} \to \bigwedge\{\neg q_{i,\ell} \wedge \neg r_{i,\ell} \,\big|\, 1 \leqslant \ell \leqslant n \text{ and } \ell \neq j\} \;\Big|\; 1 \leqslant i,j \leqslant n \Big\}.$$

7. On every column j, if there is a *rook* in row i, then there is no *queen* and no other *rook* on column j:

$$\varphi_{n,7} \stackrel{\text{def}}{=} \bigwedge\Big\{ r_{i,j} \to \bigwedge\{\neg q_{k,j} \wedge \neg r_{k,j} \,\big|\, 1 \leqslant k \leqslant n \text{ and } k \neq i\} \;\Big|\; 1 \leqslant i,j \leqslant n \Big\}.$$

8. On every diagonal, if there is a *queen*, there is no *rook* and no other *queen* on the same diagonal:

$$\varphi_{n,8} \stackrel{\text{def}}{=} \bigwedge\Big\{ q_{i,j} \to \bigwedge\{\neg q_{k,\ell} \wedge \neg r_{k,\ell} \,\big|\, k-\ell = i-j \text{ and } (k,\ell) \neq (i,j)\} \;\Big|\; 1 \leqslant i,j \leqslant n \Big\}.$$

9. On every antidiagonal, if there is a *queen*, there is no *rook* and no other *queen* on the same antidiagonal:

$$\varphi_{n,9} \stackrel{\text{def}}{=} \bigwedge\Big\{ q_{i,j} \to \bigwedge\{\neg q_{k,\ell} \wedge \neg r_{k,\ell} \,\big|\, k+\ell = i+j \text{ and } (k,\ell) \neq (i,j)\} \;\Big|\; 1 \leqslant i,j \leqslant n \Big\}.$$

The desired ψ_n is the conjunction of the preceding nine wff's: $\psi_n \stackrel{\text{def}}{=} \varphi_{n,1} \wedge \varphi_{n,2} \wedge \cdots \wedge \varphi_{n,9}$. A truth assignment σ satisfying ψ_n indicates the positions of the p queens and the $(n-p)$ rooks on the chessboard.

The ψ_n just defined is not the most succint. For example, $\varphi_{n,4}$ and $\varphi_{n,6}$ can be omitted, because they are implied by the other seven $\bigwedge\{\varphi_{n,i}|1 \leqslant i \leqslant 9 \text{ and } 4 \neq i \neq 6\}$ (why?). Moreover, some of the $\varphi_{n,i}$'s can be simplified a little; *e.g.*, instead of $\varphi_{n,4}$ we may use $\varphi'_{n,4}$ (why?), which can also be omitted altogether from ψ_n:

$$\varphi'_{n,4} \stackrel{\text{def}}{=} \bigwedge\Big\{ q_{i,j} \to \bigwedge\{\neg r_{i,\ell} \,\big|\, 1 \leqslant \ell \leqslant n\} \;\Big|\; 1 \leqslant i,j \leqslant n \Big\}.$$

Other simplifications are possible (left to you).

Our definition of ψ_n here is relative to a randomly selected subset $I \subseteq \{1,\ldots,n\}$ of p rows. An alternative definition of ψ_n is relative to a randomly selected subset $J \subseteq \{1,\ldots,n\}$ of p columns.

Finally, given that we know that the *n-Queens Problem* has a solution for every $n \geqslant 4$, it follows that the *Queens&Rooks Problem* also has a solution for every $n \geqslant 4$. That is, from an arbitrary solution of the former, it suffices to replace any $(n-p)$ queens by $(n-p)$ rooks to obtain a solution for the latter, which moreover works for every possible subset $I \subseteq \{1,\ldots,n\}$ such that $\big|I\big| = p$. The case $n = 3$ is anomalous: The 3-*Queens Problem* has no solution, and the *Queens&Rooks Problem* with 1 queen and 2 rooks has a solution only if $I = \{1\}$ or $I = \{3\}$, but not if $I = \{2\}$.

Exercise 30 (*Queens&Rooks Problem, II*)**.** Your task is to extend the *Queens&Rooks Problem* in Example 29 to the case of an infinite chessboard, which occupies the entire south-east quadrant of the Cartesian plane. The indexing of rows and columns is the same as in Exercise 27. We first ask you to consider complications arising from trying to find a simultaneous solution for two different board sizes, $n_1 \times n_1$ and $n_2 \times n_2$ with $n_1 \neq n_2$.

For the wff ψ_n in Example 29, we now add a supercript "I" as in "ψ_n^I" to indicate the subset of rows relative to which it is defined. For every $n \geqslant 3$, define the wff θ_n by:

$$\theta_n \stackrel{\text{def}}{=} \bigvee \left\{ \psi_n^I \;\middle|\; I \subseteq \{1, \dots, n\} \text{ with } |I| = p \right\}.$$

From the analysis in Example 29, for every truth assignment $\sigma : \mathcal{Q} \cup \mathcal{R} \to \{\mathit{false}, \mathit{true}\}$, it holds that $\sigma \models \theta_n$ iff σ corresponds to a solution of the *Queens&Rooks Problem* on the $n \times n$ board. This exercise has 7 parts:

1. Show that there is a truth assignment σ such that $\sigma \models \{\theta_3, \theta_4\}$.
2. Show that for every truth assignment σ we have $\sigma \not\models \{\theta_3, \theta_4, \theta_5\}$.

There are therefore finite subsets of $\Theta \stackrel{\text{def}}{=} \{\, \theta_n \mid n \geqslant 3 \,\}$ which are not satisfiable. Hence, we cannot invoke Compactness relative to Θ in order to conclude that there is a solution for the infinite *Queens&Rooks Problem*. We need to proceed more carefully:

3. Consider the configured board of queens and rooks in Figure G.7, which is a solution of the *Queens&Rooks Problem* for $n = 15$, with 5 queens and 10 rooks. Call it B_1. Your task is to define an infinite nested sequence of configured boards, starting with B_1:

   $$B_1 \;\subseteq\; B_2 \;\subseteq\; \cdots \;\subseteq\; B_k \;\subseteq\; \cdots \quad ,$$

   such that, for every $k \geqslant 1$, the configured board B_k represents a solution of the *Queens&Rooks Problem* on a $n_k \times n_k$ board for some $n_k \geqslant 3$. Thus, in particular, $n_1 = 15$. The sequence is *nested* because you must guarantee that B_k is embedded in B_{k+1}, which also implies that $15 = n_1 < n_2 < \cdots < n_k < \cdots$.

   Hint: Proceed by induction to define B_{k+1} from B_k. Figure 1.4 is a suggestion for how to do this.

4. For every $k \geqslant 1$, define a wff δ_k such that for every truth assignment σ it holds that $\sigma \models \delta_k$ iff the restriction of σ to the variables in $\mathrm{FV}(\delta_k)$ corresponds to the configured board B_k.

   Hint: Proceed by induction to define δ_{k+1} from δ_k.

5. Carefully argue that the infinite set $\Delta \stackrel{\text{def}}{=} \{\, \delta_k \mid k \geqslant 1 \,\}$ is finitely satisfiable. Hence, by Compactness, Δ is satisfiable and the *Infinite Queens&Rooks Problem* has a solution.
6. There is a close relationship between the two sets of propositional wff's, Θ and Δ. What is it? For every $k \geqslant 1$, state it as an implication from δ_k to every wff in an infinite subset Θ_k of Θ. You have to define Θ_k and justify the definition.
7. Is there an infinite subset Θ' of Θ relative to which you can use Compactness, once more and differently to show that the *Infinite Queens&Rooks Problem* has a solution?

   Hint: Yes. Define Θ' carefully, based on what you have discovered in the earlier parts of this exercise. $\square$

Example 31 (*Building a Network with Pre-Defined Pier Positions, I*)**.** This is a simple adaptation of a more general problem. Consider an $m \times n$ rectangular grid, with $m, n \geqslant 2$, on which are placed two or more *pier positions* of a network to be yet built, and zero or more *blocked positions*. Both kinds of positions are disjoint. An example of a $m \times n = 6 \times 10$ rectangular grid, with 5 *pier positions* and 8 *blocked positions*, is shown on the left in Figure 1.5.

The goal is to connect all the *pier positions* with vertical and horizontal line segments that avoid all the *blocked positions*. Put differently, we want to determine a (rootless) undirected tree T, if one exists, such that:

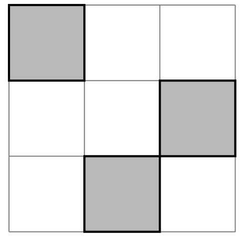

Figure 1.4: Shape of the configured board B_{k+1} in part 3 of Exercise 30: Its size is $(5 \cdot 3^{k+1}) \times (5 \cdot 3^{k+1})$, where $k \geqslant 1$. The non-empty sub-boards (shaded areas) are each a copy of B_k.

- the nodes of T include the given *pier positions* (T may include other nodes if necessary),
- the edges of T are each a sequence of vertical and horizontal segments that bypass all *blocked positions*.

Examples of two possible solutions, out of many, are shown on the right in Figure 1.5.

We can model the problem in *propositional logic* by choosing a set $\mathcal{Q}$ of doubly-indexed propositional variables, where the indices range over the positive integers:

$$\mathcal{Q} \stackrel{\text{def}}{=} \Big\{ q_{i,j} \;\Big|\; i, j \in \{1, 2, \dots\} \Big\}.$$

Let PierP and $\mathsf{BlockedP}$ be the sets of *pier positions* and *blocked positions*, each given as a pair of coordinates in the set $\{(i,j) \mid 1 \leqslant i \leqslant m,\ 1 \leqslant j \leqslant n\}$. We call the remaining positions on the grid the *free positions*, which are the coordinate pairs in the set FreeP. These three sets of positions are pairwise disjoint:

$$\mathsf{PierP} \cap \mathsf{BlockedP} = \varnothing, \quad \mathsf{PierP} \cap \mathsf{FreeP} = \varnothing, \quad \mathsf{BlockedP} \cap \mathsf{FreeP} = \varnothing.$$

A solution in the form of a tree T corresponds to a truth assignment σ which assigns *true* (resp. *false*) to variable $q_{i,j}$ iff $(i,j) \in T$ (resp. $(i,j) \notin T$). To simplify the notation a little, we define for every position (i,j) on the grid, the set of positions adjacent to (i,j):

$$\mathsf{AdjP}(i,j) \stackrel{\text{def}}{=} \Big\{ (k,\ell) \;\Big|\; k \in \{i-1, i+1\} \cap \{1, \dots, m\} \text{ and } \ell \in \{j-1, j+1\} \cap \{1, \dots, n\} \Big\}.$$

The set $\mathsf{AdjP}(i,j)$ contains 2 pairs (corner positions), or 3 pairs (boundary positions other than corner), or 4 pairs (all other positions). The constraints to be satisfied by a solution σ are:

1. For every $(i,j) \in \mathsf{PierP}$, it must hold that $\sigma(q_{i,j}) = \textit{true}$:

$$\varphi_1 \stackrel{\text{def}}{=} \bigwedge \Big\{ q_{i,j} \;\Big|\; (i,j) \in \mathsf{PierP} \Big\}.$$

2. For every $(i,j) \in \mathsf{BlockedP}$, it must hold that $\sigma(q_{i,j}) = \textit{false}$:

$$\varphi_2 \stackrel{\text{def}}{=} \bigwedge \Big\{ \neg q_{i,j} \;\Big|\; (i,j) \in \mathsf{BlockedP} \Big\}.$$

3. For every $(i,j) \in \mathsf{PierP}$, it must hold that there is at least one $q_{k,\ell} \in \mathsf{AdjP}(i,j)$ such that $\sigma(q_{k,\ell}) = \textit{true}$. This can be formulated in propositional logic by the following wff:

$$\varphi_3 \stackrel{\text{def}}{=} \bigwedge \Big\{ \bigvee \{ q_{k,\ell} \mid (k,\ell) \in \mathsf{AdjP}(i,j) \} \;\Big|\; (i,j) \in \mathsf{PierP} \Big\}.$$

4. For every $(i,j) \in \mathsf{FreeP}$, it must hold that if $\sigma(q_{i,j}) = \textit{true}$, then for at least two $q_{k,\ell} \in \mathsf{AdjP}(i,j)$ we have $\sigma(q_{k,\ell}) = \textit{true}$. This can be formulated in propositional logic by the following wff:

$$\varphi_4 \stackrel{\text{def}}{=} \bigwedge \Big\{ q_{i,j} \to \bigvee \{ q_{k,\ell} \wedge q_{k',\ell'} \mid (k,\ell), (k',\ell') \in \mathsf{AdjP}(i,j),\ (k,\ell) \neq (k',\ell') \} \;\Big|\; (i,j) \in \mathsf{FreeP} \Big\}.$$

The preceding constraints $\{\varphi_1, \varphi_2, \varphi_3, \varphi_4\}$ are *necessary* conditions that must be satisfied by a truth assignment σ corresponding to a solution T. However, they are not *sufficient*: They do not enforce that T must be a tree, *i.e.*, they allow the presence of cycles in T. For an example of a T with cycles, consider the two solutions T_1 and T_2 on the right in Figure 1.5 and two truth assignments, say $\sigma_1, \sigma_2 : \mathcal{Q} \to \{\mathit{false}, \mathit{true}\}$, modeling them. Define a new truth assignment σ as follows. For every $q_{i,j} \in \mathcal{Q}$, let:

$$\sigma(q_{i,j}) \stackrel{\text{def}}{=} \begin{cases} \sigma_1(q_{i,j}) \vee \sigma_2(q_{i,j}) & \text{if } \sigma_1(q_{i,j}) = \mathit{true} \text{ or } \sigma_2(q_{i,j}) = \mathit{true}, \\ \mathit{false} & \text{if } \sigma_1(q_{i,j}) = \mathit{false} \text{ and } \sigma_2(q_{i,j}) = \mathit{false}. \end{cases}$$

After omitting position $(3, 3)$ from PierP, the new σ satisfies every wff in $\{\varphi_1, \varphi_2, \varphi_3, \varphi_4\}$ as well as corresponds to a solution T containing two cycles. We handle the problem of cycles in Exercise 32.

Moreover, even if a solution T does not include cycles, the constraints in $\{\varphi_1, \varphi_2, \varphi_3, \varphi_4\}$ do not guarantee that T is a tree. This happens when T is a forest of two or more trees. For example, consider again truth assignment σ_1 which models the plain-edge tree T_1 on the right in Figure 1.5. Define another truth assignment σ by setting, for every $q_{i,j} \in \mathcal{Q}$:

$$\sigma(q_{i,j}) \stackrel{\text{def}}{=} \begin{cases} \sigma_1(q_{i,j}) & \text{if } (i,j) \notin \{(3,5), (3,6)\}, \\ \mathit{false} & \text{if } (i,j) \in \{(3,5), (3,6)\}. \end{cases}$$

The resulting σ satisfies every wff in $\{\varphi_1, \varphi_2, \varphi_3, \varphi_4\}$ as well as corresponds to a solution T consisting of two unconnected trees.

If we are only interested in the *existence* of a solution in the form of a single tree, it is a simple fact of graph theory that two separate components of a forest can be connected to form one tree, provided there are no blocked positions preventing the connection. If we are interested in the *construction* of a such a tree, the situation is a little more complicated; one approach is proposed in Exercise 33.

Exercise 32 (*Building a Network with Pre-Defined Pier Positions, II*)**.** Read carefully the preceding Example 31, where satisfaction of the constraints in $\{\varphi_1, \varphi_2, \varphi_3, \varphi_4\}$ does not eliminate the presence of cycles. There are different ways of handling this problem. In this exercise, we consider a perspicuous approach that eliminates cyles as well as produces a "shortest" solution T, in the sense that T traverses a smallest possible number of *free positions*. An example is on the right in Figure 1.5: solution T_1 (which includes 11 *free positions*) is shorter than solution T_2 (which includes 14 *free positions*).

The approach we take in this exercise is based on what is called the *MaxSAT problem*, a variant of the SAT problem. In the MaxSAT problem, we deal with propositional wff's in *conjunctive normal form* (CNF). A propositional wff ψ is a CNF iff it is of the form $\psi \stackrel{\text{def}}{=} C_1 \wedge C_2 \wedge \cdots \wedge C_n$ where every C_i is a non-empty finite disjunction of literals (propositional variables and negated propositional variables). In the MaxSAT problem, we do not require that all the disjunctive clauses in $\{C_1,\ C_2,\ \ldots\ ,\ C_n\}$ be satisfied. More precisely, in this exercise, we use the *weighted MaxSAT problem*:[13]

Weighted MaxSAT: Let $\psi \stackrel{\text{def}}{=} C_1^{w_1} \wedge C_2^{w_2} \wedge \cdots \wedge C_n^{w_n}$ be a propositional wff in CNF where every clause C_i is assigned a weight w_i, positive or negative. The goal is to compute a truth assignment σ maximizing the total weight of the satisfied clauses. Call such a σ a *maximizing truth assignment* for the weighted CNF ψ. In applications, it is often desirable to distinguish between *hard clauses* (clauses that must be satisfied) and *soft clauses* (clauses that may or may not be satisfied by a maximizing truth assignment).

There are two parts in this exercise, the first of which does not use the MaxSAT problem:

[13] There is a large body of research devoted to the MaxSAT problem and many of its variations. The problem is *NP-hard*, because we can use its solution to find a solution for the SAT problem, which is *NP-complete*.

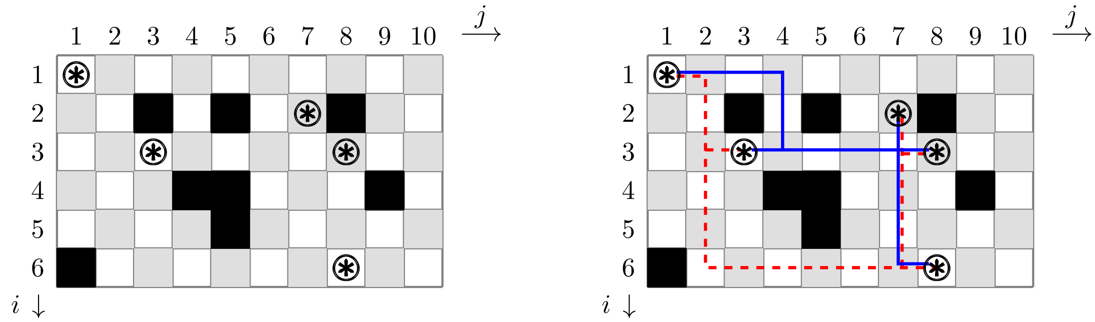

Figure 1.5: On the left is a 6×10 grid on which are placed 5 *pier positions*, each denoted ⊛, and 8 *blocked positions*, each indicated by ■. On the right is the same grid with two possible solutions (out of many). The plain-edge solution T_1 traverses 11 *free positions*, the dashed-edge solution T_2 traverses 14 *free positions*.

1. Transform the constraints in $\{\varphi_1, \varphi_2, \varphi_3, \varphi_4\}$ into a single wff in CNF $\varphi \stackrel{\text{def}}{=} C_1 \wedge C_2 \wedge \cdots \wedge C_n$. For every truth assignment σ it must holds that $\sigma \models \varphi$ iff $\sigma \models \varphi_1 \wedge \varphi_2 \wedge \varphi_3 \wedge \varphi_4$.
2. Modify the CNF φ obtained in part 1 to obtain a weighted CNF ψ such that if σ is a maximizing truth assignment for ψ, then σ corresponds to a forest T on the input grid, which is moreover shortest (among possible solutions).

   Hint 1: A shortest solution T is necessarily acyclic and therefore a forest.

   Hint 2: The clauses in the CNF φ in part 1 express *necessary conditions* for a solution to be a forest. They must therefore all become *hard clauses* when copied in the weighted CNF ψ. You must add (weighted) *soft clauses*, which have no counterparts in φ, such that satisfying a largest number of them will minimize the number of visited free positions. □

Exercise 33 (*Building a Network with Pre-Defined Pier Positions, III*)**.** Read carefully Example 31 before attempting this exercise. In the preceding Exercise 32, we use the *weighted MaxSAT problem* to eliminate cycles from a solution σ of the constraints in $\{\varphi_1, \varphi_2, \varphi_3, \varphi_4\}$, which may correspond to a forest with two or more trees. In this exercise, your task is to compute a truth assignment σ corresponding to a single tree T.

Hint: If there are at most three piers, the MaxSAT solution in Exercise 32 is already the desired solution for this exercise. If there are $p \geqslant 4$ piers, consider running your MaxSAT algorithm $(p-2)$ times, so that the desired truth assignment σ is defined incrementally in $(p-2)$ iterations of MaxSAT, with one iteration for an initial selection of three piers followed by one iteration for each additional pier beyond the initial three.

Warning: The solution constructed according to the *hint* may not be the shortest. The order in which piers are inserted affects the number of visited free positions. The latter number is minimized by choosing an appropriate order of pier insertion (a task beyond the goal of this exercise). □

Exercise 34 (*No-Three-In-Line Problem*)**.** This is an old problem of discrete geometry, not yet fully resolved as of this writing, which asks for the maximum number of pebbles that can be placed on an $n \times n$ chessboard so that no three pebbles are collinear, *i.e.*, not on the same row or column or diagonal. An upper bound on the number of pebbles is $2n$ because, by the Pigeonhole Principle, placing $2n + 1$ pebbles makes one row or one column necessarily contain three of them. But is $2n$ a reachable upper bound for all n? See Figure 1.6 for two solutions when $n = 10$, in which case the upper bound $2n = 20$ is reached.

We can take the *No-Three-In-Line Problem* as a more complex variation on the *n-Queens Problem* in Exercise 27, with "pebbles" instead of "queens". Specifically, we use the same set $\mathcal{Q}$ of doubly-indexed propositional variables, with variable $q_{i,j}$ assigned truth-value *true* (resp. *false*) if there

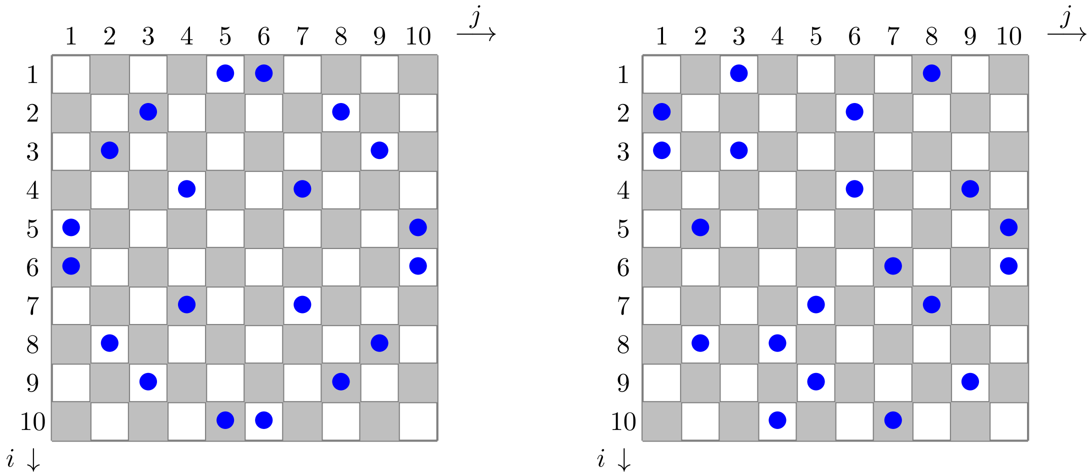

Figure 1.6: Two solutions of the Not-Three-In-Line Problem when $n = 10$ in Exercise 34, both satisfying conditions (a), (b), (c), and (d), in the exercise.

is (resp. there is not) a pebble placed in position (i, j) of the board, where the index i (resp. the index j) ranges over the vertical axis downward (resp. the horizontal axis rightward). There are two parts in this exercise:

1. Write a wff $\psi_n \stackrel{\text{def}}{=} \psi_n^{\text{row}} \wedge \psi_n^{\text{col}} \wedge \psi_n^{\text{diag1}} \wedge \psi_n^{\text{diag2}}$ in $\mathsf{WFF}_{\mathsf{PL}}(\mathcal{Q})$ according to the following specification:

   (a) ψ_n^{row} is satisfied iff there are exactly two pebbles in each row,

   (b) ψ_n^{col} is satisfied iff there are exactly two pebbles in each column,

   (c) ψ_n^{diag1} is satisfied iff there are at most two pebbles in each diagonal,

   (d) ψ_n^{diag2} is satisfied iff there are at most two pebbles in each antidiagonal.

   Hint: Given two distinct positions (i_1, j_1) and (i_2, j_2) along a diagonal, it is always that $i_1 - j_1 = i_2 - j_2$. And if the two positions are along an antidiagonal, then it is always that $i_1 + j_1 = i_2 + j_2$.

2. We extend the *Not-Three-In-Line Problem* to an infinite chessboard, which occupies the entire south-east quadrant of the Cartesian plane. Give a precise argument, based on Compactness, showing that if the problem can be solved for each $n \geqslant 4$ (there is no solution for $n < 4$), then a solution exists for the *Infinite Not-Three-In-Line Problem.* [14] □

[14] It is known that the *Not-Three-In-Line Problem* has a solution for every $n \leqslant 46$, but not for $n > 46$ as here formulated. It is conjectured that, for $n > 46$, no matter how you place $2n$ pebbles on the board, you are doomed to find three of them that are collinear. Put differently, for $n > 46$, it is conjectured that you are forced to place fewer than $2n$ pebbles to avoid three collinear pebbles. For more information on the *Not-Three-In-Line Problem*, search the Web.

1.3.3 Partial Orders

Exercise 35 (*Dilworth's Theorem and its Extension to Infinite Partial Orders*). A *partially ordered set*, or *poset* for short, is a set A with a binary relation $\trianglelefteq \subseteq A \times A$, called a *partial order*. For all $a, b \in A$, it is convenient to write "$a \trianglelefteq b$" instead of "$(a, b) \in \trianglelefteq$". The three properties characterizing a poset are:

1. *reflexivity*: $a \trianglelefteq a$ for all $a \in A$.
2. *anti-symmetry*: if $a \trianglelefteq b$ and $b \trianglelefteq a$, then $a = b$, for all $a, b \in A$.
3. *transitivity*: if $a \trianglelefteq b$ and $b \trianglelefteq c$, then $a \trianglelefteq c$, for all $a, b, c \in A$.

The poset is a *total order* (or *linear order*) if it satisfies the additional property: $a \trianglelefteq b$ or $b \trianglelefteq a$, for all $a, b \in A$; in words, all the elements in A are *comparable*.

The poset $(A, \trianglelefteq)$ has *width at most n* iff every subset of A of pairwise incomparable elements has size $\leqslant n$. A *chain* in the poset is a subset of A on which $\trianglelefteq$ is a total order.

Dilworth's Theorem Every finite poset $(A, \trianglelefteq)$ of width at most n can be partitioned into n chains.

Your task in this exercise is to extend Dilworth's Theorem to an arbitrary infinite poset using Compactness for the *propositional logic*. To simplify a little, let the domain of the poset be $\mathbb{N}$ (the set of natural numbers) and assume the width of $(\mathbb{N}, \trianglelefteq)$ as a poset is 3. You thus have to show that $\mathbb{N}$ can be partitioned into three chains, each totally ordered by $\trianglelefteq$.

Hint 1: Use four sets of propositional variables:

$\mathcal{Q} \stackrel{\text{def}}{=} \{q_{i,j} \mid i, j \in \mathbb{N}\}$ (think of $q_{i,j}$ as set to *true* if $i \trianglelefteq j$ and to *false* if $i \ntrianglelefteq j$)

$\mathcal{R} \stackrel{\text{def}}{=} \{ r_i \mid i \in \mathbb{N}\}$ (think of r_i as set to *true* if i is in the first chain, and to *false* otherwise)

$\mathcal{S} \stackrel{\text{def}}{=} \{ s_i \mid i \in \mathbb{N}\}$ (think of s_i as set to *true* if i is in the second chain, and to *false* otherwise)

$\mathcal{T} \stackrel{\text{def}}{=} \{ t_i \mid i \in \mathbb{N}\}$ (think of t_i as set to *true* if i is in the third chain, and to *false* otherwise)

Hint 2: Write sets of propositional wff's to express the desired conclusions: Use the variables in $\mathcal{Q}$ to express the partial order $\trianglelefteq$; the variables in $\mathcal{R}$, $\mathcal{S}$, and $\mathcal{T}$ to express that the corresponding indices (*i.e.*, elements in $\mathbb{N}$) form three disjoint chains; and that every $i \in \mathbb{N}$ is a member of one of the three chains, *i.e.*, exactly one of the variables in $\{r_i, s_i, t_i\}$ is set to *true*.

Hint 3: You are allowed to invoke Dilworth's Theorem, which is explicitly limited to *finite posets*, without proving it (the proof is not trivial). $\square$