A PAIR OF FOUR-ELEMENT HORIZONTAL GENERATING SETS OF A PARTITION LATTICE

GÁBOR CZÉDLI

Dedicated to the memory of my local colleague and co-author Árpád Kurusa

Abstract. Let $\lfloor x \rfloor$ and $\lceil x \rceil$ denote the lower integer part and the upper integer part of a real number x, respectively. Our main goal is to construct four partitions of a finite set A with $n \geq 7$ elements such that each of the four partitions has exactly $\lceil n/2 \rceil$ blocks and any other partition of A can be obtained from the given four by forming joins and meets in a finite number of steps. We do the same with $\lceil n/2 \rceil - 1$ instead of $\lceil n/2 \rceil$, too. To situate the paper within lattice theory, recall that the *partition lattice* $\mathrm{Eq}(A)$ of a set A consists of all partitions (equivalently, of all equivalence relations) of A. For a natural number n, $[n]$ and $\mathrm{Eq}(n)$ will stand for $\{1, 2, \dots, n\}$ and $\mathrm{Eq}([n])$, respectively. In 1975, Heinrich Strietz proved that, for any natural number $n \geq 3$, $\mathrm{Eq}(n)$ has a four-element generating set; half a dozen papers have been devoted to four-element generating sets of partition lattices since then. We give a simple proof of his just-mentioned result. We call a generating set X of $\mathrm{Eq}(n)$ *horizontal* if each member of X has the same height, denoted by $h(X)$, in $\mathrm{Eq}(n)$; no such generating sets have been known previously. We prove that for each natural number $n \geq 4$, $\mathrm{Eq}(n)$ has two four-element horizontal generating sets X and Y such that $h(Y) = h(X) + 1$; for $n \geq 7$, $h(X) = \lfloor n/2 \rfloor$.

1. Notes on the dedication

Árpád Kurusa, 1961–2024, was an excellent geometer. The present paper is dedicated to his memory. In addition to his high reputation in geometry, his editorial and technical editorial work for several mathematical journals as well as his textbooks (in Hungarian) were also deeply acknowledged. From 2000 to 2018, he led the Department of Geometry at the Bolyai (Mathematical) Institute of the University of Szeged. As the title of [3] shows, our collaboration has added a piece to the traditionally strong interrelation between geometry and lattice theory. At the motivational level, the present paper has some (but very slight) connection to the just-mentioned joint paper. Indeed, partition lattices form a specific subclass of *geometric* lattices, and the term "horizontal" is rooted in a *geometric* perspective of these lattices.

2. Introduction and our theorem

Given a set A, the collection of *equivalences*, that is, the collection of reflexive, symmetric, transitive relations of A form a lattice $\mathrm{Eq}(A)$, the *equivalence lattice* of A. In this lattice, the meet and the join are the intersection and the transitive hull of the union, respectively. By the well-known bijective correspondence between the equivalences of A and the partitions of A, $\mathrm{Eq}(A)$ is isomorphic to the *partition lattice* of A, which consists of all partitions of A. By the just-mentioned correspondence, we make no sharp distinction between equivalences and partitions in our terminology and notations. To explain that we use the notation $\mathrm{Eq}(A)$ rather than something like $\mathrm{Part}(A)$, note that equivalences are more

1991 *Mathematics Subject Classification.* 06B99, 06C10.

Key words and phrases. Partition lattice; equivalence lattice; minimum-sized generating set; horizontal generating set, four-element generating set.

This research was supported by the National Research, Development and Innovation Fund of Hungary, under funding scheme K 138892.

August 21, 2024.

appropriate for performing the lattice operations and forming restrictions. For a natural number n, we let $[n] := \{1, 2, \dots, n\}$, and we usually abbreviate Eq$([n])$ to Eq(n).

Partition lattices play an important role in lattice theory since congruence lattices, which play a central role in universal algebra, are naturally embedded in partition lattices. In fact, every lattice is embeddable into a partition lattice by Whitman [10] and each finite lattice into a finite partition lattice by Pudlák and Tůma [7]; note that these facts can be exploited in some proofs, for example, in [1]. Furthermore, every partition lattice Eq(A) is known to be a *geometric lattice*, that is, an atomistic semimodular lattice; see, e.g., Grätzer [5, Section IV.4] or [6, Section V.3]. Being *atomistic* means that each element x of Eq(A) is the join of all atoms below x. *Semimodularity* is understood as upper semimodularity, that is, for any $x, y, z \in \text{Eq}(A)$, $x \preceq y$ implies that $x \vee z \preceq y \vee z$, where $\preceq$ is the "is covered by or equal to" relation.

A subset X of Eq(A) is a *generating set* of Eq(A) if X extends to no proper subset S of Eq(A) such that S is closed with respect to joins and meets. In the seventies, Strietz [8] and [9] proved that, for any natural number $n \geq 3$, Eq(n) has a four-element generating set. His result is optimal, since Eq(n) does not have a three-element generating set provided that $n \geq 4$. Since Strietz's pioneering work was published in [8] and [9], five additional papers have already been devoted to the four-element generating sets of equivalence lattices; see [4], the 2nd-, the 3rd-, and the 4th-item in the "References" section of [4], and Zádori [11].

For $n \geq 3$, which is always assumed, each permutation of $[n]$ extends to an automorphism[1] of Eq(n), and such an automorphism sends generating sets to generating sets. We say that two generating sets of Eq(n) are *essentially different* if no such automorphism sends one of them to the other one. We know even from Strietz [8] and [9] that, for n large enough, Eq(n) has several essentially different four-element generating sets. Many more (essentially different) four-element generating sets have been given in [4]. However, it is very likely by the computer-assisted section of [4] that only an infinitesimally small percentage of the four-element generating sets of Eq(n) are known for n large. Exploring more such generating sets seems to be a reasonable target in its own right, and there is an additional motivation: Namely, the more small generating sets of Eq(n) are available, the more the cryptographic ideas of [2] can benefit from equivalence lattices. (If there are and *we know* many four-element generating sets, then we can extend them to small generating sets in very many ways.)

Before explaining what sort of new four-element generating sets of Eq(n) we are going to present, note that even at the very beginning of this type or research in the seventies, Strietz himself paid attention to some lattice theoretical properties of his four-element generating sets. For $n \geq 4$, he showed that a four-element generating set is either an *antichain* (that is, a subset with no comparable elements) or it is of order type $1+1+2$, that is, exactly two out of the four generators are comparable. He managed to prove that Eq(n) has a four-element generating set of order type $1+1+2$ for every integer $n \geq 10$. Briefly saying, Eq(n) is $(1+1+2)$-*generated* for $n \geq 10$. With ingenious constructions, Zádori [11] improved "$n \geq 10$" to $n \geq 7$, and he gave a visual proof of Strietz's result that Eq(n) has a four-element generating set; his proofs are simpler than Strietz's ones. Zádori [11] left open the problem whether Eq(5) and Eq(6) are $(1+1+2)$-generated. This problem was solved as recently as 2020 in [4], where an affirmative answer for Eq(6) was given but a computer-assisted negative answer for Eq(5) was provided.

As Eq(n) is a geometric lattice, there is a natural property of a subset, which is more restrictive than being an antichain. To introduce it, recall that the *length* of an n-element chain is $n-1$. The least element and the largest element of Eq(n) or Eq(A) will be denoted by Δ and ∇, respectively. If confusion threatens, we write Δ_n, ∇_A, etc.. The height of an element $\mu \in \text{Eq}(n)$ is the length of a maximal chain in the interval $[\Delta, \mu]$; we know from the Jordan-Hölder Chain Condition for semimodular lattices, see, e.g., Grätzer [5, Theorem IV.2.1 on page 226] or [6, Theorem 377], that no matter which maximal chain is taken. We denote the *height* of μ by $h(\mu)$. A subset X of Eq(n) is *horizontal* if its elements are of

[1]It is worth noting that by K. Kearnes: *Automorphisms of a finite partition lattice,* Version 2023-11-28, https://math.stackexchange.com/q/4814790, each automorphism of Eq(n) is obtained in this way.

the same height; in this case, the common height of the elements of X is denoted by $h(X)$. A horizontal subset of Eq(n) is necessarily an antichain. Clearly, Eq(n) for $n \geq 3$ has a *horizontal generating set*, since the set of atoms is such. To get a better insight into the four-element generating sets of partition lattices, it is reasonable to determine those natural numbers n for which Eq(n) has a *four-element horizontal generating set*. In fact, we are going to do more by showing that whenever Eq(n) has a four-element antichain at all, that is, whenever $n \geq 4$, then it has two four-element horizontal generating sets of neighboring heights. To smooth our terminology, let us introduce the notation

$$\mathrm{HFHGS}(n) := \{h(X) : X \text{ is a four-element horizontal generating set of } \mathrm{Eq}(n)\};$$

the acronym above comes from the **h**eights of **f**our-element **h**orizontal **g**enerating **s**ets. For a real number r, we denote by $\lfloor r \rfloor$ and $\lceil r \rceil$ the *lower integer part* and the *upper integer part* of r; for example, $\lfloor \sqrt{2} \rfloor = 1$ and $\lceil \sqrt{2} \rceil = 2$. Let $\mathbb{N}^+$ denote the set of positive integers.

Theorem 1. *For every natural number $n \geq 4$, the partition lattice* Eq(n) *has two four-element horizontal generating sets X and Y such that $h(Y) = h(X) + 1$ holds for their heights. Furthermore,*

$$\mathrm{HFHGS}(n) \supseteq \{\lfloor n/2 \rfloor, \lfloor n/2 \rfloor + 1\} \text{ for all integers } n \geq 7 \text{ and also for } n = 5, \text{ and} \tag{2.1}$$

$$\mathrm{HFHGS}(n) \subseteq \{k \in \mathbb{N}^+ : \lfloor (n-1)/4 \rfloor + 1 \leq k \leq n - \lceil \sqrt[4]{n} \rceil\} \text{ for all integers } n \geq 4. \tag{2.2}$$

Based on the following statement, we conjecture that "$\supseteq$" in (2.1) is never an equality for $n \geq 7$. We do not know whether $\lim_{n\to\infty} |\mathrm{HFHGS}(n)| = \infty$ and HFHGS(n) is always a convex subset of 0. We know HFHGS(n) only for $n \in \{4, 5, 6, 7, 8\}$. In the proposition below, each occurrence of the relation symbol $\overset{\text{comp}}{=}$ denotes an equality that we could prove only with the assistance of the brute force of a computer.

Proposition 1. *We have the following equalities and inclusions:*

$$\mathrm{HFHGS}(4) = \{1, 2\}, \tag{2.3}$$

$$\mathrm{HFHGS}(5) = \{2, 3\}, \tag{2.4}$$

$$\{2, 3\} \subseteq \mathrm{HFHGS}(6) \subseteq \{2, 3, 4\}, \text{ in fact, } \mathrm{HFHGS}(6) \overset{\text{comp}}{=} \{2, 3\}, \tag{2.5}$$

$$\{2, 3, 4\} \subseteq \mathrm{HFHGS}(7) \subseteq \{2, 3, 4, 5\}, \text{ in fact, } \mathrm{HFHGS}(7) \overset{\text{comp}}{=} \{2, 3, 4\}, \text{ and} \tag{2.6}$$

$$\{3, 4, 5\} \subseteq \mathrm{HFHGS}(8) \subseteq \{2, 3, 4, 5, 6\}, \text{ in fact, } \mathrm{HFHGS}(8) \overset{\text{comp}}{=} \{3, 4, 5\}. \tag{2.7}$$

Remark 1. (2.3) and (2.5) witness that (2.1) fails for $n \in \{4, 6\}$. Note also that concrete four-element horizontal generating sets witnessing (2.1) and (2.3)–(2.7) are defined by Lemma 5 combined with Assertion 1, by Lemmas 6, 7 and 8 combined with both (the Key) Lemma 4 and Assertion 1, and in the rest of the lemmas presented in Section 5. For n large, the just-mentioned four-element horizontal generating sets are given only inductively; the inductive feature could be eliminated but we do not strive for non-inductive definitions of these generating sets.

The rest of the paper is devoted to proving Theorem 1 and Proposition 1. Unless explicitly stated otherwise, we assume that $4 \leq n \in \mathbb{N}^+$ for the remainder of the paper.

3. Some lemmas, the Key Lemma, and a new proof of one of Strietz's results

For a finite nonempty set A, if $\{a_{1,1}, \dots, a_{1,t_1}\}$, $\dots$, $\{a_{k,1}, \dots, a_{k,t_k}\}$ is a repetition-free list of the blocks of a partition $\mu \in \mathrm{Eq}(A)$, then we denote both μ and the corresponding equivalence by

$$\mathrm{eq}(a_{1,1}, \dots, a_{1,t_1}; \dots; a_{k,1}, \dots, a_{k,t_k}) \quad \text{or} \quad \mathrm{eq}(a_{1,1} \dots a_{1,t_1}; \dots; a_{k,1} \dots a_{k,t_k}).$$

That is, we omit the commas when no confusion threatens but not the block-separating semicolons. Usually, the elements in a block and the blocks are listed in lexicographic order. For example,

$$\Delta_4 = \mathrm{eq}(1; 2; 3; 4), \ \nabla_4 = \mathrm{eq}(1234), \text{ and } \nabla_{11} = \mathrm{eq}(1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11);$$

for more involved examples, see Lemmas 5–15. For $u, v \in A$, the least equivalence of A collapsing u and v will be denoted by $\mathrm{at}(u,v)$ or, if confusion threatens, by $\mathrm{at}_A(u,v)$. For example, in $\mathrm{Eq}(6)$, $\mathrm{at}(2,5) = \mathrm{eq}(1;25;3;4;6)$. Note that $\mathrm{at}(u,v)$ is an ***at**om* of $\mathrm{Eq}(A)$ (that is, a cover of Δ), and every atom of $\mathrm{Eq}(A)$ is of this form.

We define the *graph* $G(S)$ of a sublattice S of $\mathrm{Eq}(A)$ by letting A be the *vertex set* of $G(S)$ and letting $\{(a,b) : a \neq b \text{ and } \mathrm{at}(a,b) \in S\}$ be the *edge set* of $G(S)$. (No matter if we consider (a,b) and (b,a) equal or different.) A *Hamiltonian circle* of $G(S)$ is a permutation $a_1, a_2, \dots, a_n$ of the elements of A such that $\mathrm{at}(a_{i-1}, a_i) \in S$ for $i \in [n] - \{1\}$ and $\mathrm{at}(a_n, a_1) \in S$. Of course, $G(S)$ need not have a Hamiltonian circle. The following lemma occurs, explicitly or implicitly, in several papers dealing with generating sets of equivalence lattices; see, for example, Czédli and Oluoch [4, Lemma 2.5]. For the reader's convenience, we are going to outline its trivial proof.

Lemma 1 ((Hamiltonian Cycle Lemma))**.** *For a finite set A with at least three elements and a sublattice S of $\mathrm{Eq}(A)$, we have that $S = \mathrm{Eq}(A)$ if and only if $G(S)$ has a Hamiltonian circle.*

Proof. The "only if" part is trivial. To prove the "if" part, let $a_1, \dots, a_n$ be a Hamiltonian circle of $G(S)$. As each element of the atomistic lattice $\mathrm{Eq}(A)$ is the join of some atoms, it suffices to show that for all $i \neq j$, $i, j \in [n]$, we have that $\mathrm{at}(a_i, a_j) \in S$. This membership follows from

$$\begin{aligned}\mathrm{at}(a_i,a_j) = &\Big(\mathrm{at}(a_i,a_{i+1}) \vee \mathrm{at}(a_{i+1},a_{i+2}) \vee \dots \vee \mathrm{at}(a_{j-1},a_j)\Big)\\ &\wedge \Big(\mathrm{at}(a_i,a_{i-1}) \vee \mathrm{at}(a_{i-1},a_{i-2}) \vee \dots \vee \mathrm{at}(a_2,a_1)\\ &\vee \mathrm{at}(a_1,a_n) \vee \mathrm{at}(a_n,a_{n-1}) \vee \mathrm{at}(a_{n-1},a_{n-2}) \vee \dots \vee \mathrm{at}(a_{j+1},a_j)\Big)\end{aligned}$$

and the "commutativity" $\mathrm{at}(x,y) = \mathrm{at}(y,x)$. ☐

Let $\mathbb{Z}_4 := (\{0,1,2,3\}, +)$ denote the cyclic group of order 4; the addition in it is performed modulo 4. To give the lion's share of the proof of (2.3) and also to present an easy consequence of Lemma 1, we present the following lemma, in which the addition is understood in $\mathbb{Z}_4$.

Lemma 2. *Both $X := \{\mathrm{at}(i,i+1) : i \in \mathbb{Z}_4\}$ and $Y := \{\mathrm{at}(i,i+1) \vee \mathrm{at}(i+1,i+2) : i \in \mathbb{Z}_4\}$ are four-element horizontal generating sets of $\mathrm{Eq}(\mathbb{Z}_4) \cong \mathrm{Eq}(4)$.*

Proof. Let S be the sublattice of $\mathrm{Eq}(\mathbb{Z}_4)$ generated by Y. Since

$$\mathrm{at}(i,i+1) = \big(\mathrm{at}(i,i+1) \vee \mathrm{at}(i+1,i+2)\big) \wedge \big(\mathrm{at}(i-1,i) \vee \mathrm{at}(i,i+1)\big) \in S \text{ for } i \in \mathbb{Z}_4, \tag{3.1}$$

the sequence 0,1,2,3 is a Hamilton cycle in $G(S)$. Hence, Y is a generating set by Lemma 1. Lemma 1 applies to X without (3.1) immediately. The rest of Lemma 2 is trivial. ☐

Next, we introduce a concept that is crucial in the proof of Theorem 1. By an *eligible system* we mean a 7-tuple

$$\mathcal{A} = (A, \alpha, \beta, \gamma, \delta, u, v)$$

such that A is a finite set, u and v are distinct elements of A, $\{\alpha, \beta, \gamma, \delta\}$ is a four-element generating set of $\mathrm{Eq}(A)$, and

$$\alpha \vee \delta = \nabla, \qquad \alpha \wedge \delta = \Delta, \tag{3.2}$$

$$\beta \wedge \big(\gamma \vee \mathrm{at}(u,v)\big) = \Delta, \qquad \gamma \wedge \big(\beta \vee \mathrm{at}(u,v)\big) = \Delta, \tag{3.3}$$

$$\text{and } \beta \vee \gamma \vee \mathrm{at}(u,v) = \nabla. \tag{3.4}$$

To present an example and also for a later reference, we formulate the following statement.

Lemma 3. *With* $\alpha = \mathrm{eq}(123;4)$, $\beta = \mathrm{eq}(14;2;3)$, $\gamma = \mathrm{eq}(1;2;34)$, *and* $\delta = \mathrm{eq}(1;24;3)$,

$$\mathcal{A} := ([4], \alpha, \beta, \gamma, \delta, 1, 2) \tag{3.5}$$

is an eligible system

Proof. Let S be the sublattice of Eq(4) generated by $\{\alpha, \beta, \gamma, \delta\}$. Since

$$\mathrm{at}(1,2) = \mathrm{eq}(12;3;4) = \alpha \wedge (\beta \vee \delta) \in S, \ \ \mathrm{at}(2,3) = \alpha \wedge (\gamma \vee \delta) \in S, \ \ \mathrm{at}(3,4) = \gamma \in S,$$

and $\mathrm{at}(4,1) = \beta \in S$, the sequence 1, 2, 3, 4 is a Hamiltonian cycle in $G(S)$. Thus, Lemma 1 implies that $\{\alpha, \beta, \gamma, \delta\}$ generates Eq(4). Since (3.2), (3.3), and (3.4) are trivially satisfied, the proof of Lemma 3 is complete. □

For $A \subseteq B$ and $\mu \in \mathrm{Eq}(A)$, the smallest equivalence of B that includes μ will be denoted by μ_B^{ext}. The superscript in the notation comes from "**ext**ension". As a partition, μ_B^{ext} consists of the blocks of μ and the singleton blocks $\{b\}$ for $b \in B - A$.

Lemma 4 ((Key Lemma)). *Assume that* $(A, \alpha, \beta, \gamma, \delta, u, v)$ *is an eligible system,* $|A| \geq 4$, $w \notin A$, *and* $B = A \cup \{w\}$. *Let*

$$\begin{aligned} &\alpha' := \beta_B^{\mathrm{ext}} \vee \mathrm{at}_B(u,w), \quad \beta' := \alpha_B^{\mathrm{ext}}, \quad \gamma' := \delta_B^{\mathrm{ext}}, \\ &\delta' := \gamma_B^{\mathrm{ext}} \vee \mathrm{at}_B(v,w), \quad u' := u, \quad v' := w. \end{aligned} \tag{3.6}$$

Then the **e**xtended **s**ystem

$$\mathrm{ES}(\mathcal{A}) := \mathcal{B} = \big(B, \alpha', \beta', \gamma', \delta', u', v'\big) \tag{3.7}$$

is also an eligible system. The heights of the partitions occurring in (3.6)–(3.7) *satisfy that*

$$h(\alpha') = h(\beta) + 1, \qquad h(\beta') = h(\alpha), \qquad h(\gamma') = h(\delta), \qquad h(\delta') = h(\gamma) + 1. \tag{3.8}$$

Proof. Assume that $\mathcal{A}$ is an eligible system and $\mathcal{B} = \mathrm{ES}(\mathcal{A})$ is as in (3.7). We will frequently but mostly implicitly use the obvious fact that the function $f\colon \mathrm{Eq}(A) \to \mathrm{Eq}(B)$ defined by $\mu \mapsto \mu_B^{\mathrm{ext}}$ is a lattice embedding and, for any $\mu \in \mathrm{Eq}(A)$, $h(f(\mu)) = h(\mu)$. Denote by S the sublattice generated by $\{\alpha', \beta', \gamma', \delta'\}$ in $\mathrm{Eq}(B)$. For $\mu \in \mathrm{Eq}(B)$, let $\mu\restriction_A$ denote the *restriction* of μ to A. That is, as an equivalence, $\mu\restriction_A = \mu \cap (A \times A)$. E.g., $\big((\Delta_A)_B^{\mathrm{ext}}\big)\restriction_A = \Delta_A$. Note the obvious rule:

$$(\rho_B^{\mathrm{ext}})\restriction_A = \rho \quad \text{and} \quad (\mu\restriction_A)_B^{\mathrm{ext}} = \mu \wedge (\nabla_A)_B^{\mathrm{ext}} \quad \text{for every } \rho \in \mathrm{Eq}(A) \text{ and } \mu \in \mathrm{Eq}(B). \tag{3.9}$$

Let us agree that, for $x, y \in B$, $\mathrm{at}(x,y)$ is understood as $\mathrm{at}_B(x,y)$ even when $x, y \in A$. We claim that for any $\mu \in \mathrm{Eq}(A)$ and for any $d \in A$,

$$\big(\mu_B^{\mathrm{ext}} \vee \mathrm{at}_B(d,w)\big)\restriction_A = \mu; \quad \text{and, in particular,} \tag{3.10}$$

$$\alpha'\restriction_A = \beta \quad \text{and} \quad \delta'\restriction_A = \gamma. \tag{3.11}$$

The inequality $\big(\mu_B^{\mathrm{ext}} \vee \mathrm{at}_B(d,w)\big)\restriction_A \geq \mu$ is clear. To show the converse inequality, assume that $a \neq b$ and (a,b) belongs to $\big(\mu_B^{\mathrm{ext}} \vee \mathrm{at}_B(d,w)\big)\restriction_A$. Then $a, b \in A$ and, by the description of the join in equivalence lattices, there exists a *shortest* sequence $x_0 = a, x_1, \dots, x_{t-1}, x_t = b$ of elements of B such that, for each $i \in [t]$,

$$\text{either } (x_{i-1}, x_i) \in \mu_B^{\mathrm{ext}} \text{ or } (x_{i-1}, x_i) \in \{(d,w), (w,d)\}. \tag{3.12}$$

Since this sequence is repetition-free, the first alternative in (3.12) means that $(x_{i-1}, x_i) \in \mu$. By way of contradiction, suppose that not all elements of the sequence are in A. Let j be the smallest subscript such that $x_j \notin A$. As $x_0 = a \in A$ and $x_t = b \in A$, we have that $0 < j < t$. By the choice of j, $x_{j-1} \in A$. This rules out that $(x_{j-1}, x_j) = (w,d)$. Since $x_j \notin A$, $(x_{j-1}, x_j) \in \mu$ cannot occur either. Hence, $(x_{j-1}, x_j) = (d,w)$. However, then the only possibility to continue the sequence is that $(x_j, x_{j+1}) = (w,d)$. So d occurs in the sequence at least twice, which contradicts the fact that our sequence is repetition-free. Therefore, all elements of the sequence are in A, whereby the first alternative of (3.12) holds for all i. Thus, $(x_{i-1},x_i)\in\mu$ for $i\in[t]$, and we obtain the required membership $(a,b)=(x_0,x_t)\in\mu$ by transitivity. We have shown (3.10). Letting $(\mu,d):=(\beta,u)$ and $(\mu,d):=(\gamma,v)$, (3.10) implies (3.11).

Next, using the first half of (3.2) (and the fact that f is an embedding), we obtain that $(\nabla_A)_B^{\mathrm{ext}}=(\alpha\vee\delta)_B^{\mathrm{ext}}=\alpha_B^{\mathrm{ext}}\vee\delta_B^{\mathrm{ext}}=\beta'\vee\gamma'$ belongs to S. Hence, so does $\alpha'\wedge(\nabla_A)_B^{\mathrm{ext}}$. By the second half of (3.9) applied to $\mu:=\alpha'$, this equivalence is $(\alpha'\rceil_A)_B^{\mathrm{ext}}$, whence $(\alpha'\rceil_A)_B^{\mathrm{ext}}\in S$. Therefore, applying (3.11), $\beta_B^{\mathrm{ext}}\in S$. As β and γ play a symmetric role, γ_B^{ext} is also in S. By (3.6), S contains $\alpha_B^{\mathrm{ext}}=\beta'$ and $\delta_B^{\mathrm{ext}}=\gamma'$. So $f(\mu)=\mu_B^{\mathrm{ext}}\in S$ for every $\mu\in\{\alpha,\beta,\gamma,\delta\}$. Since f is an embedding and $\{\alpha,\beta,\gamma,\delta\}$ generates $\mathrm{Eq}(A)$, we conclude that $f(\mathrm{Eq}(A))\subseteq S$. In particular, $\mathrm{at}_B(u,v)=f(\mathrm{at}_A(u,v))\in S$. Based on this containment, we claim that

$$\mathrm{at}_B(u,w)=\alpha'\wedge\bigl(\mathrm{at}_B(u,v)\vee\delta'\bigr)\in S. \tag{3.13}$$

As $\mathrm{at}_B(u,v),\alpha',\delta'\in S$, it suffices to show the equality in (3.13). The inequality "$\leq$" in place of the equality is clear by the definition of α' given in (3.6). To show the converse inequality, assume that $a\neq b$ and (a,b) belongs to the right-hand side of the equality in (3.13). Let $\nu:=\mathrm{at}_A(u,v)\vee\gamma$. Observe that

$$(a,b)\in\alpha'\wedge\bigl(\nu_B^{\mathrm{ext}}\vee\mathrm{at}_B(v,w)\bigr), \tag{3.14}$$

since

$$\begin{aligned}\alpha'\wedge\bigl(\nu_B^{\mathrm{ext}}\vee\mathrm{at}_B(v,w)\bigr)&=\alpha'\wedge\Bigl(\bigl(\mathrm{at}_A(u,v)\vee\gamma\bigr)_B^{\mathrm{ext}}\vee\mathrm{at}_B(v,w)\Bigr)\\&=\alpha'\wedge\Bigl(\bigl(\mathrm{at}_A(u,v)\bigr)_B^{\mathrm{ext}}\vee\gamma_B^{\mathrm{ext}}\vee\mathrm{at}_B(v,w)\Bigr) \qquad (3.15)\\&=\alpha'\wedge\bigl(\mathrm{at}_B(u,v)\vee\gamma_B^{\mathrm{ext}}\vee\mathrm{at}_B(v,w)\bigr)\overset{(3.6)}{=}\alpha'\wedge\bigl(\mathrm{at}_B(u,v)\vee\delta'\bigr). \qquad (3.16)\end{aligned}$$

As $a\neq b$ and $|B-A|=|\{w\}|=1$, at least one of a and b is in A. By symmetry, we can assume that $a\in A$. Depending on the position of b, there are two cases.

First, assume that b is also in A. Then $(a,b)\in\alpha'$ and (3.11) give that $(a,b)\in\beta$. As (a,b) is in the second meetand in (3.14) and $a,b\in A$, we have that $(a,b)\in\bigl(\nu_B^{\mathrm{ext}}\vee\mathrm{at}_B(v,w)\bigr)\rceil_A$. Hence, (3.10) applied to $(\mu,d):=(\nu,v)$ yields that $(a,b)\in\nu$. Thus, (a,b) belongs to $\beta\wedge\nu=\beta\wedge\bigl(\mathrm{at}_A(u,v)\vee\gamma\bigr)$, which is Δ_A by (3.3). Since $(a,b)\in\Delta_A$ contradicts the assumption $a\neq b$, the first case cannot occur.

Second, assume that $b\notin A$. Then $(a,w)=(a,b)\in\alpha'\wedge\bigl(\mathrm{at}_B(u,v)\vee\delta'\bigr)$ and $a\in A$. By (3.6), $(w,u)\in\alpha'$. As both (w,v) and (v,u) belong to the second meetand of (3.15), (w,u) belongs to this meetand, too. These facts, (3.15), and (3.16) give that $\alpha'\wedge\bigl(\mathrm{at}_B(u,v)\vee\delta'\bigr)$ contains (w,u). By transitivity, it contains (a,u), too. If we had that $a\neq u$, then (a,u) (with u playing the role of b) would be a contradiction by the first case. Thus, $a=u$, that is, $(a,b)=(u,w)\in\mathrm{at}_B(u,w)$, as required. We have shown the validity of (3.13).

We obtain the following fact analogously; we can derive it also from (3.13) by symmetry, since $(A;\delta,\gamma,\beta,\alpha,v,u)$ is also an eligible system:

$$\mathrm{at}_B(v,w)=\delta'\wedge\bigl(\mathrm{at}_B(u,v)\vee\alpha'\bigr)\in S. \tag{3.17}$$

With $n:=|A|$, list the elements of B as follows: $c_1:=u$, c_2, …, c_{n-1}, $c_n:=v$, $c_{n+1}:=w$. Since $f(\mathrm{Eq}(A))\subseteq S$ and $c_1,\dots,c_n\in A$, we have that $\mathrm{at}_B(c_i,c_{i+1})=f\bigl(\mathrm{at}_A(c_i,c_{i+1})\bigr)\in S$, that is, (c_i,c_{i+1}) is an edge of $G(S)$ for $i\in[n-1]$. So are $(c_n,c_{n+1})=(v,w)$ and $(c_{n+1},c_1)=(w,u)$ by (3.17) and by (3.13), respectively. Therefore, our list is a Hamiltonian cycle, and Lemma 1 implies that $\{\alpha',\beta',\gamma',\delta'\}$ is a generating set of $\mathrm{Eq}(B)$. This set is four-element since $|B|\geq 4$ and so we know from Strietz [8] or [9] that $\mathrm{Eq}(B)$ cannot be generated by less than four elements.

Clearly, $u'=u\in A$ is distinct from $v'=w\in B-A$. Since

$$\begin{aligned}\alpha'\vee\delta'&\overset{(3.6)}{=}\beta_B^{\mathrm{ext}}\vee\mathrm{at}_B(u,w)\vee\gamma_B^{\mathrm{ext}}\vee\mathrm{at}_B(v,w)=\beta_B^{\mathrm{ext}}\vee\gamma_B^{\mathrm{ext}}\vee\mathrm{at}_B(u,v)\vee\mathrm{at}_B(v,w)\\&=\bigl(\beta\vee\gamma\vee\mathrm{at}_A(u,v)\bigr)_B^{\mathrm{ext}}\vee\mathrm{at}_B(v,w)\overset{(3.4)}{=}(\nabla_A)_B^{\mathrm{ext}}\vee\mathrm{at}_B(v,w)=\nabla_B,\end{aligned}$$

$\mathcal{B}$ satisfies the first half of (3.2). To show by way of contradiction that $\mathcal{B}$ fulfills the second half, suppose that $a \neq b$ and $(a,b) \in \alpha' \wedge \delta'$. If $a, b \in A$, then (3.11) leads to $(a,b) \in \beta \wedge \gamma = \Delta_A$, contradicting that $a \neq b$. So one of a and b is w, and we can assume that $a \in A$ and $b = w$. As $(a,w) = (a,b) \in \alpha'$ and $(w,u) \in \alpha'$, we have that $(a,u) \in \alpha'$. Hence, $(a,u) \in \beta$ by (3.11). Similarly, $(a,w),(w,v) \in \delta'$ and (3.11) imply that $(a,v) \in \gamma$. The just-obtained memberships and relations give that

$$(a,u) \in \beta \wedge \big(\gamma \vee \mathrm{at}_A(u,v)\big) \quad \text{and} \quad (a,v) \in \gamma \wedge \big(\beta \vee \mathrm{at}_A(u,v)\big).$$

Combining this with (3.3), we obtain that $a = u$ and $a = v$, contradicting $u \neq v$. So we have proved that $\mathcal{B}$ fulfills (3.2).

By symmetry, to show that $\mathcal{B}$ satisfies (3.3), it suffices to deal with its first half. For the sake of contradiction, suppose that $\beta' \wedge \big(\gamma' \vee \mathrm{at}_B(u',v')\big) \neq \Delta_B$. Then we can pick $a, b \in B$ such that $a \neq b$ and

$$(a,b) \in \beta' \wedge \big(\gamma' \vee \mathrm{at}_B(u',v')\big) \overset{(3.6)}{=} \alpha_B^{\mathrm{ext}} \wedge \big(\delta_B^{\mathrm{ext}} \vee \mathrm{at}_B(u,w)\big). \tag{3.18}$$

The containment $(a,b) \in \alpha_B^{\mathrm{ext}}$ gives that $a, b \in A$. The meet in Eq(B) is the set-theoretic intersection, so it commutes with the restriction map. Hence, applying the first equality of (3.9) with $\rho := \alpha$ and (3.10) with $(\mu, d) := (\delta, u)$ at $\overset{*}{=}$, (3.18) leads to

$$\begin{aligned}(a,b) &\in \Big(\alpha_B^{\mathrm{ext}} \wedge \big(\delta_B^{\mathrm{ext}} \vee \mathrm{at}_B(u,w)\big)\Big)\rceil_A \\ &= \alpha_B^{\mathrm{ext}}\rceil_A \wedge \big(\delta_B^{\mathrm{ext}} \vee \mathrm{at}_B(u,w)\big)\rceil_A \overset{*}{=} \alpha \wedge \delta \overset{(3.2)}{=} \Delta_A \subseteq \Delta_B,\end{aligned}$$

which contradicts the assumption $a \neq b$ and proves that $\mathcal{B}$ satisfies (3.3). Since

$$\begin{aligned}\beta' \vee \gamma' \vee \mathrm{at}_B(u',v') &\overset{(3.6)}{=} \alpha_B^{\mathrm{ext}} \vee \delta_B^{\mathrm{ext}} \vee \mathrm{at}_B(u,w) = (\alpha \vee \delta)_B^{\mathrm{ext}} \vee \mathrm{at}_B(u,w) \\ &\overset{(3.2)}{=} (\nabla_A)_B^{\mathrm{ext}} \vee \mathrm{at}_B(u,w) = \nabla_B,\end{aligned}$$

$\mathcal{B}$ satisfies (3.4), too. We have proved that $\mathcal{B}$ is an eligible system, as required.

For a finite nonempty set H and μ in Eq(H), let NumB(μ) denote the ***num**ber of **b**locks* of μ. For example, if $\mu = \mathrm{eq}(14;25;3) \in \mathrm{Eq}(5)$, then $\mathrm{NumB}(\mu) = 3$. The following folkloric fact is trivial:

$$\text{For any } \mu \in \mathrm{Eq}(H), \quad h(\mu) + \mathrm{NumB}(\mu) = |H|. \tag{3.19}$$

Clearly, (3.6) leads to

$$\begin{aligned}&\mathrm{NumB}(\alpha') = \mathrm{NumB}(\beta), &&\mathrm{NumB}(\beta') = \mathrm{NumB}(\alpha) + 1, \\ &\mathrm{NumB}(\gamma') = \mathrm{NumB}(\delta) + 1, \text{ and} &&\mathrm{NumB}(\delta') = \mathrm{NumB}(\gamma).\end{aligned}$$

These equalities and (3.19) imply (3.8), completing the proof of the Key Lemma. □

Now we are in the position to give a new proof of Strietz's result stating that Eq(n) is four-generated. For those who prefer theoretical arguments rather than long and tedious computations with concrete partitions, the proof below is presumably simpler than the earlier ones.

Corollary 1 ((Strietz [8] and [9])). *For any natural number $n \geq 3$,* Eq(n) *has a four-element generating set.*

Proof. As the case $n = 3$ is trivial, we assume that $n \geq 4$. Let $\mathcal{A}_4$ be the eligible system given in (3.5). For $n > 4$, define $\mathcal{A}_n$ as $\mathrm{ES}(\mathcal{A}_{n-1})$. Then, for each $n \geq 4$, $\mathcal{A}_n$ is an n-element eligible system by Lemmas 3 and (the Key) Lemma 4. Thus, by the definition of eligible systems, Eq(n) is four-generated, completing the proof of Corollary 1. □

4. A tediously provable lemma

The n-th *Bell number* $B(n)$ is defined to be the number of elements of $\mathrm{Eq}(n)$, that is, $B(n) := |\mathrm{Eq}(n)|$. As n grows, $B(n)$ grows very fast; see https://oeis.org/A000110 of N. J. A. Sloan's Online Encyclopedia of Integer Sequences. For example, $|\mathrm{Eq}(6)| = B(6) = 203$, $|\mathrm{Eq}(8)| = 4\,140$, $|\mathrm{Eq}(9)| = 21\,147$, and $|\mathrm{Eq}(20)| = 51\,724\,158\,235\,372 \approx 5.17 \cdot 10^{13}$. These large numbers explain our experience that even when it is feasible to prove that a four-element subset X of $\mathrm{Eq}(n)$ generates $\mathrm{Eq}(n)$, this task requires straightforward but tedious computations in general. Each of Lemmas 5–15 belongs to this category by stating that a subset X of $\mathrm{Eq}(n)$ generates $\mathrm{Eq}(n)$; some of these lemmas state slightly more, but these surpluses are trivial to verify. We offer two ways to verify these lemmas.

First, one can read their proofs based on Lemma 1. One of these proofs is given in this section. As the rest of these proofs are long without containing a single new idea, the proofs of Lemmas 6–15 are given in Appendix 1

Second, the author has developed three closely related computer programs in Dev-Pascal 1.9.2 under Windows 10. These programs, which are available at https://tinyurl.com/czg-equ2024p or at the author's website[2] http://tinyurl.com/g-czedli/, form a mini-package. The main program and its auxiliary program are also given in Appendices 2 and 3. The third program performs the same tasks as the first one and also uses the auxiliary program. Despite being slower, it is more cross-platform because it requires less computer memory. For $n \le 9$, the auxiliary program lists the elements of $\mathrm{Eq}(n)$; the other two programs rely on this list. In what follows, by a program, we mean the main program. The program can "prove" Lemmas 5–15, and it can also "prove" the $\overset{\mathrm{comp}}{=}$ parts of (2.5)–(2.7). In fact, the program has been designed to perform the following two tasks.

First, the program can take an $n \in \{4, 5, \dots, 9\}$ and a four-element subset X of $\mathrm{Eq}(n)$ as inputs. After enlarging X by adding the join and the meet of any two of its elements as long as the enlargement is proper, the program computes the sublattice S generated by X. Then the program displays the size $|S|$ of S on the screen and tells whether X generates $\mathrm{Eq}(n)$. The program can prove Lemma 8, where $n = 9$, in about fifteen minutes. For Lemma 14, where $n = 8$, 25 seconds suffice. Note that for just one four-element subset X of $\mathrm{Eq}(n)$, it is not worthwhile to create and the program does not create the operation tables of $\mathrm{Eq}(n)$. For this (the first) task, there is no difference between the main program and its slower variant.

Second, for a given $n \in \{4, 5, \dots, 9\}$ and a $k \in [n-1]$ as inputs, the program decides whether $\mathrm{Eq}(n)$ has a four-element horizontal generating set of height k. For $(n, k) = (8, 2)$, this takes about three and a half minutes, provided the program runs on a desktop computer with AMD Ryzen 7 2700X Eight-Core Processor and 3.70 GHz with 16 GB memory. For $(n, k) = (9, 3)$, if $\mathrm{Eq}(9)$ has no four-element horizontal generating set of height 3, which we do not know, the program would need about a month; partially because there is not enough computer memory to store the operation tables of $\mathrm{Eq}(9)$ and also because there are significantly more cases.

The quotation marks around "proved" in a paragraph above indicate that the author believes but cannot prove that the program itself is error-free. The source code of the program and that of its auxiliary program are 24 and 8 kilobytes, respectively, totaling 32 kilobytes. Proving *exactly* that the program is perfect would probably be harder than verifying all proofs in Appendix 1.

[2]This standard "tiny" short link redirects us to the real URL https://www.math.u-szeged.hu/˜czedli/ .

Lemma 5. *With*

$$\alpha := \mathrm{eq}(123; 4; 5), \tag{4.1}$$
$$\beta := \mathrm{eq}(1; 23; 45), \tag{4.2}$$
$$\gamma := \mathrm{eq}(13; 25; 4), \text{ and} \tag{4.3}$$
$$\delta := \mathrm{eq}(15; 2; 34), \tag{4.4}$$

$([5], \alpha, \beta, \gamma, \delta, 1, 4)$ *is an eligible system and* $h(\alpha) = h(\beta) = h(\gamma) = h(\delta) = 2$.

Proof. Let S denote the sublattice of Eq(5) generated by $\{\alpha, \beta, \gamma, \delta\}$. We will list some members of S; each of them belongs to S by earlier containments as indicated.

$$\mathrm{eq}(1; 23; 4; 5) = \mathrm{eq}(123; 4; 5) \wedge \mathrm{eq}(1; 23; 45) \in S \text{ by (4.1) and (4.2)}, \tag{4.5}$$
$$\mathrm{eq}(13; 2; 4; 5) = \mathrm{eq}(123; 4; 5) \wedge \mathrm{eq}(13; 25; 4) \in S \text{ by (4.1) and (4.3)}, \tag{4.6}$$
$$\mathrm{eq}(1235; 4) = \mathrm{eq}(123; 4; 5) \vee \mathrm{eq}(13; 25; 4) \in S \text{ by (4.1) and (4.3)}, \tag{4.7}$$
$$\mathrm{eq}(15; 234) = \mathrm{eq}(15; 2; 34) \vee \mathrm{eq}(1; 23; 4; 5) \in S \text{ by (4.4) and (4.5)}, \tag{4.8}$$
$$\mathrm{eq}(1345; 2) = \mathrm{eq}(15; 2; 34) \vee \mathrm{eq}(13; 2; 4; 5) \in S \text{ by (4.4) and (4.6)}, \tag{4.9}$$
$$\mathrm{eq}(15; 2; 3; 4) = \mathrm{eq}(15; 2; 34) \wedge \mathrm{eq}(1235; 4) \in S \text{ by (4.4) and (4.7)}, \tag{4.10}$$
$$\mathrm{eq}(1; 2; 3; 45) = \mathrm{eq}(1; 23; 45) \wedge \mathrm{eq}(1345; 2) \in S \text{ by (4.2) and (4.9)}, \tag{4.11}$$
$$\mathrm{eq}(13; 245) = \mathrm{eq}(13; 25; 4) \vee \mathrm{eq}(1; 2; 3; 45) \in S \text{ by (4.3) and (4.11)}, \tag{4.12}$$
$$\mathrm{eq}(1; 24; 3; 5) = \mathrm{eq}(15; 234) \wedge \mathrm{eq}(13; 245) \in S \text{ by (4.8) and (4.12)}. \tag{4.13}$$

Let $E(S)$ denote the edge set of the graph $G(S)$; it is defined in the paragraph preceding Lemma 1. Since $(1, 3) \in E(S)$ by (4.6), $(3, 2) \in E(S)$ by (4.5), $(2, 4) \in E(S)$ by (4.13), $(4, 5) \in E(S)$ by (4.11), and $(5, 1) \in E(S)$ by (4.10), the sequence $1, 3, 2, 4, 5$ is a Hamiltonian cycle of $G(S)$. Hence, $\{\alpha, \beta, \gamma, \delta\}$ is a generating set of Eq(5) by Lemma 1. Armed with this fact, now it is a trivial task to verify that $([5], \alpha, \beta, \gamma, \delta, 1, 4)$ satisfies (3.2), (3.3), and (3.4), whereby it is an eligible system. Thus, (3.19) completes the proof Lemma 5. □

5. The rest of tediously provable lemmas

We need the following ten lemmas, too. As indicated in the second paragraph of Section 4, their proofs are given in Appendix 1.

Lemma 6. *With*

$$\alpha := \mathrm{eq}(134; 256; 7), \qquad \beta := \mathrm{eq}(146; 27; 3; 5),$$
$$\gamma := \mathrm{eq}(135; 2; 4; 67), \text{ and} \qquad \delta := \mathrm{eq}(12; 357; 46),$$

$([7], \alpha, \beta, \gamma, \delta, 2, 3)$ *is an eligible system,* $h(\alpha) = h(\delta) = 4$, *and* $h(\beta) = h(\gamma) = 3$.

Lemma 7. *With*

$$\alpha := \mathrm{eq}(134; 258; 67), \qquad \beta := \mathrm{eq}(14; 2; 36; 578),$$
$$\gamma := \mathrm{eq}(17; 25; 348; 6), \text{ and} \qquad \delta := \mathrm{eq}(12; 378; 456),$$

$([8], \alpha, \beta, \gamma, \delta, 2, 6)$ *is an eligible system,* $h(\alpha) = h(\delta) = 5$, *and* $h(\beta) = h(\gamma) = 4$.

Lemma 8. *With*

$$\alpha := \mathrm{eq}(178; 249; 356), \qquad \beta := \mathrm{eq}(19; 26; 378; 45),$$

$$\gamma := \mathrm{eq}(1;28;359;467), \text{ and } \qquad \delta := \mathrm{eq}(169;258;347),$$

$([9], \alpha, \beta, \gamma, \delta, 1, 2)$ *is an eligible system,* $h(\alpha) = h(\delta) = 6$, *and* $h(\beta) = h(\gamma) = 5$.

Lemma 9. *With*

$$\alpha := \mathrm{eq}(134;25), \qquad \beta := \mathrm{eq}(13;245),$$
$$\gamma := \mathrm{eq}(12;345), \text{ and } \qquad \delta := \mathrm{eq}(124;35),$$

$\{\alpha, \beta, \gamma, \delta\}$ *generates* $\mathrm{Eq}(5)$ *and* $h(\alpha) = h(\beta) = h(\gamma) = h(\delta) = 3$.

Lemma 10. *With*

$$\alpha := \mathrm{eq}(12;34;5;6), \qquad \beta := \mathrm{eq}(1;2;35;46),$$
$$\gamma := \mathrm{eq}(1;25;36;4), \text{ and } \qquad \delta := \mathrm{eq}(15;24;3;6),$$

$\{\alpha, \beta, \gamma, \delta\}$ *generates* $\mathrm{Eq}(6)$ *and* $h(\alpha) = h(\beta) = h(\gamma) = h(\delta) = 2$.

Lemma 11. *With*

$$\alpha := \mathrm{eq}(13;256;4), \qquad \beta := \mathrm{eq}(156;2;34),$$
$$\gamma := \mathrm{eq}(12;35;46), \text{ and } \qquad \delta := \mathrm{eq}(13;246;5),$$

$\{\alpha, \beta, \gamma, \delta\}$ *generates* $\mathrm{Eq}(6)$ *and* $h(\alpha) = h(\beta) = h(\gamma) = h(\delta) = 3$.

Lemma 12. *With*

$$\alpha := \mathrm{eq}(1;24;35;6;7), \qquad \beta := \mathrm{eq}(14;26;3;5;7),$$
$$\gamma := \mathrm{eq}(1;2;34;5;67), \text{ and } \qquad \delta := \mathrm{eq}(17;2;3;4;56),$$

$\{\alpha, \beta, \gamma, \delta\}$ *generates* $\mathrm{Eq}(7)$ *and* $h(\alpha) = h(\beta) = h(\gamma) = h(\delta) = 2$.

Lemma 13. *With*

$$\alpha := \mathrm{eq}(13;24;567), \qquad \beta := \mathrm{eq}(125;3;467)$$
$$\gamma := \mathrm{eq}(1357;26;4), \text{ and } \qquad \delta := \mathrm{eq}(126;35;47),$$

$\{\alpha, \beta, \gamma, \delta\}$ *generates* $\mathrm{Eq}(7)$ *and* $h(\alpha) = h(\beta) = h(\gamma) = h(\delta) = 4$.

Lemma 14. *With*

$$\alpha := \mathrm{eq}(18;2;35;4;67), \qquad \beta := \mathrm{eq}(1;24;37;5;68),$$
$$\gamma := \mathrm{eq}(16;2;34;57;8), \text{ and } \qquad \delta := \mathrm{eq}(12;3;45;6;78),$$

$\{\alpha, \beta, \gamma, \delta\}$ *generates* $\mathrm{Eq}(8)$ *and* $h(\alpha) = h(\beta) = h(\gamma) = h(\delta) = 3$.

Lemma 15. *With*

$$\alpha := \mathrm{eq}(137;246;58), \qquad \beta := \mathrm{eq}(146;257;38),$$
$$\gamma := \mathrm{eq}(136;2;4578), \text{ and } \qquad \delta := \mathrm{eq}(1245;37;68),$$

$\{\alpha, \beta, \gamma, \delta\}$ *generates* $\mathrm{Eq}(8)$ *and* $h(\alpha) = h(\beta) = h(\gamma) = h(\delta) = 5$.

6. Proving Theorem 1 and Proposition 1 with our lemmas

Since the proof of Theorem 1 relies on parts of Proposition 1 and the proof of Proposition 1 uses (2.2) from Theorem 1, we present a combined proof of both the theorem and the proposition.

Proof. (Proving Theorem 1 and Proposition 1) First, we deal with (2.2). Assume that $\{\alpha_1, \dots, \alpha_4\}$ is a four-element horizontal generating set of Eq(n) with height k. That is, $k = h(\alpha_i)$ for $i \in [4]$. We need to prove that

$$\lfloor (n-1)/4 \rfloor + 1 \le k \le n - \lceil \sqrt[4]{n}\, \rceil. \tag{6.1}$$

By semimodularity, see Grätzer [5, Theorem IV.2.2 on page 226], the height of $\alpha_1 \vee \dots \vee \alpha_4$ is at most $h(\alpha_1) + \dots + h(\alpha_4) = 4k$. The just-mentioned join is the largest element of the sublattice S generated by $\{\alpha_1, \dots, \alpha_4\}$. But this sublattice is Eq(n), so this join is ∇_n, whereby $h(\nabla_n) \le 4k$. We know from, say, (3.19) that $h(\nabla_n) = n-1$. Thus, the previous inequality turns into $(n-1)/4 \le k$. If $(n-1)/4 < k$, then $\lfloor (n-1)/4 \rfloor < k$ and we obtain the first inequality of (6.1) since k is an integer. Hence, it suffices to exclude that $(n-1)/4 = k$. To obtain a contradiction, suppose that $(n-1)/4 = k$, that is, $n - 1 = h(\nabla_n) = 4k$. Let $i \in [4]$. As $h(\alpha_i) = k$, we can find k atoms $\beta_{k(i-1)+1}$, $\beta_{k(i-1)+2}$, ..., β_{ki} in Eq(n) such that α_i is the join of these atoms; the existence of such atoms is clear in Eq(n) and it is true even in any geometric lattice by Grätzer [5, Theorems IV.2.4 and IV.2.5 on pages 228–229] or [6, Theorems 380 and 381]. As $\{\alpha_1, \dots, \alpha_4\}$ generates Eq(n), $\alpha_1 \vee \dots \vee \alpha_4 = \nabla_n$. Hence,

$$h\Big(\bigvee_{j=1}^{4k} \beta_j\Big) = h(\alpha_1 \vee \dots \vee \alpha_4) = h(\nabla_n) = n - 1 = 4k.$$

Therefore, Grätzer [5, Theorem IV.2.4 on page 228] or [6, Theorem 380] yields that $\{\beta_1, \dots, \beta_{4k}\}$ is an independent set of atoms; this means that $\{\beta_1, \dots, \beta_{4k}\}$ generates a Boolean sublattice T of Eq(n). In particular, T is a distributive. As $\alpha_1, \dots, \alpha_4$ are in T, they generate a sublattice of T, which is distributive, too. This means that Eq(n) is distributive, which contradicts the assumption that $n \ge 4$. Therefore, $(n-1)/4 = k$ cannot occur and we have proved the first inequality in (6.1).

Clearly, $\alpha_1 \wedge \dots \wedge \alpha_4$, which is the smallest element of S, is Δ_n. Let $b :=$ NumB(α_i); by (3.19), $b = n - k$ does not depend on $i \in [4]$. The largest block C_1 of α_1 has at least n/b elements. When we form the meet $\alpha_1 \wedge \alpha_2$, then C_1 splits into at most b blocks of $\alpha_1 \wedge \alpha_2$ and the largest one of these blocks has at least $(n/b)/b$ elements. So $\alpha_1 \wedge \alpha_2$ has a block C_2 with at least n/b^2 elements. And so on; finally, $\Delta_n = \alpha_1 \wedge \dots \wedge \alpha_4$ has a block with at least n/b^4 elements. But Δ_n has only one-element blocks, whereby $n/b^4 \le 1$, that is, $b \ge \sqrt[4]{n}$. Thus $b \ge \lceil \sqrt[4]{n}\, \rceil$, since $b \in \mathbb{N}^+$. Therefore, as we know from (3.19) that $b = n - k$, we obtain that $k \le n - \lceil \sqrt[4]{n}\, \rceil$. This completes the proof of (6.1) and that of (2.2).

Next, assume that $\mathcal{A} = (A, \alpha, \beta, \gamma, \delta, u, v)$. With the "extended system operator" introduced in (3.7), we use the notation $(C, \alpha'', \beta'', \gamma'', \delta'', u'', v'')$ for $\mathrm{ES}^2(\mathcal{A}) := \mathrm{ES}(\mathrm{ES}(\mathcal{A}))$. Clearly, (the Key) Lemma 4 implies the following assertion.

Assertion 1. *If $\mathcal{A} = (A, \alpha, \beta, \gamma, \delta, u, v)$ is an eligible system and*

$$\mathcal{C} = (C, \alpha'', \beta'', \gamma'', \delta'', u'', v'')$$

is $\mathrm{ES}^2(\mathcal{A})$, then $\mathcal{C}$ is also an eligible system, $h(\alpha'') = h(\alpha) + 1$, $h(\beta'') = h(\beta) + 1$, $h(\gamma'') = h(\gamma) + 1$, and $h(\delta'') = h(\delta) + 1$.

Resuming the proof, let us agree that, for any meaningful x, $\mathcal{A}_{\mathrm{L}x}$ denotes the eligible system defined in Lemma x. For example, $\mathcal{A}_{\mathrm{L}5}$ is defined in Lemma 5. W call an eligible system *horizontal* if its four partitions have the same height; this common height is the *height* of the system.

By Lemma 5, $\mathcal{A}_{\mathrm{L}5}$ is a 5-element horizontal eligible system of height 2. Applying Assertion 1 repeatedly, we obtain a 7-element horizontal eligible system, a 9-element horizontal eligible system, etc. of heights

3, 4, ..., respectively. Thus,

$$\text{for } n \geq 5 \text{ odd, } \mathrm{Eq}(n) \text{ has a four-element horizontal generating set of height } \lfloor n/2 \rfloor. \tag{6.2}$$

By Lemma 7 and (the Key) Lemma 4, $\mathrm{ES}(\mathcal{A}_{\mathrm{L}7})$ is a 9-element horizontal eligible system of height 5. Applying Assertion 1 repeatedly, we obtain an 11-element horizontal eligible system, a 13-element horizontal eligible system, etc. of heights 6, 7, ..., respectively. Hence,

$$\text{for } n \geq 9 \text{ odd, } \mathrm{Eq}(n) \text{ has a four-element horizontal generating set of height } \lfloor n/2 \rfloor + 1. \tag{6.3}$$

By Lemma 6 and (the Key) Lemma 4, $\mathrm{ES}(\mathcal{A}_{\mathrm{L}6})$ is an 8-element horizontal eligible system of height 4. Hence, the repeated use of Assertion 1 yields that

$$\text{for } n \geq 8 \text{ even, } \mathrm{Eq}(n) \text{ has a four-element horizontal generating set of height } \lfloor n/2 \rfloor. \tag{6.4}$$

By Lemma 8 and (the Key) Lemma 4, $\mathrm{ES}(\mathcal{A}_{\mathrm{L}8})$ is a 10-element horizontal eligible system of height 6. Hence, the repeated use of Assertion 1 yields that

$$\text{for } n \geq 10 \text{ even, } \mathrm{Eq}(n) \text{ has a four-element horizontal generating set of height } \lfloor n/2 \rfloor + 1. \tag{6.5}$$

We know from Lemma 9 that Eq(5) is generated by a four-element horizontal generating set of height $\lceil 5/2 \rceil + 1$. By Lemma 13, Eq(7) has four-element horizontal generating set of height a $(\lfloor 7/2 \rfloor + 1)$. For Eq(8), a four-element horizontal generating set of height $(\lfloor 8/2 \rfloor + 1)$ is provided by Lemma 15. These three facts, (6.2), (6.3), (6.4), and (6.5) imply (2.1).

In what follows, we will implicitly use that $\mathrm{Eq}(n)$ has no four-element horizontal subset of height 0 or $n-1$. Since there is no four-element subset of height 0 or 3 in Eq(4), Lemma 2 implies (2.3).

Since $\{2,3\} \subseteq \mathrm{HFHGS}(5)$ by (2.2), (2.1) implies (2.4).

We obtain from (2.2) and Lemmas 10–11 that $\{2,3\} \subseteq \mathrm{HFHGS}(6) \subseteq \{2,3,4\}$. As the already mentioned computer program yields that $4 \notin \mathrm{HFHGS}(6)$ in less than a second[3], (2.5) holds.

Lemma 12, (2.1), and (2.2) imply that $\{2,3,4\} \subseteq \mathrm{HFHGS}(7) \subseteq \{2,3,4,5\}$. In 2 seconds, the program excludes that $5 \in \mathrm{HFHGS}(7)$. Thus, we have shown (2.6).

Lemma 14, (2.1) and (2.2) yield that $\{3,4,5\} \subseteq \mathrm{HFHGS}(8) \subseteq \{2,3,4,5,6\}$, as required. The program excludes 2 and 6 from HFHGS(8) in three and a half minutes and in one minute, respectively. Thus, we proved the validity of (2.7) and that of Proposition 1.

Finally, the first sentence of Theorem 1 follows from (2.3), (2.4) or (2.1), the first inclusion in (2.5), and from (2.1). The combined proof of Theorem 1 and Proposition 1 is complete. ☐

7. Appendix 1: the proofs of the technical lemmas stated in Section 5

Proof. (Proof of Lemma 6) It is easy to check that (3.2), (3.3), and (3.4) hold. The equalities for the heights of α, ..., δ are trivial. As in the proof of Lemma 5, let S be the sublattice generated by $\{\alpha, \beta, \gamma, \delta\}$. Then

$$\alpha = \mathrm{eq}(134; 256; 7) \in S, \tag{7.1}$$

$$\beta = \mathrm{eq}(146; 27; 3; 5) \in S, \tag{7.2}$$

$$\gamma = \mathrm{eq}(135; 2; 4; 67) \in S, \tag{7.3}$$

$$\delta = \mathrm{eq}(12; 357; 46) \in S, \tag{7.4}$$

$$\mathrm{eq}(14; 2; 3; 5; 6; 7) = \mathrm{eq}(134; 256; 7) \wedge \mathrm{eq}(146; 27; 3; 5) \in S \text{ by (7.1) and (7.2)}, \tag{7.5}$$

$$\mathrm{eq}(13; 2; 4; 5; 6; 7) = \mathrm{eq}(134; 256; 7) \wedge \mathrm{eq}(135; 2; 4; 67) \in S \text{ by (7.1) and (7.3)}, \tag{7.6}$$

[3] The auxiliary program creates the auxiliary files containing the lists of partitions of $[n]$ for $n \leq 9$ in 4 seconds, but this has to be done only once. Thus, here and later, even though the program needs these files, the just-mentioned 4 seconds are not counted. The time for entering n and k are not counted either.

$$\mathrm{eq}(1;2;3;46;5;7) = \mathrm{eq}(146;27;3;5) \wedge \mathrm{eq}(12;357;46) \in S \text{ by (7.2) and (7.4)}, \tag{7.7}$$
$$\mathrm{eq}(1;2;35;4;6;7) = \mathrm{eq}(12;357;46) \wedge \mathrm{eq}(135;2;4;67) \in S \text{ by (7.4) and (7.3)}, \tag{7.8}$$
$$\mathrm{eq}(1246;357) = \mathrm{eq}(12;357;46) \vee \mathrm{eq}(14;2;3;5;6;7) \in S \text{ by (7.4) and (7.5)}, \tag{7.9}$$
$$\mathrm{eq}(12357;46) = \mathrm{eq}(12;357;46) \vee \mathrm{eq}(13;2;4;5;6;7) \in S \text{ by (7.4) and (7.6)}, \tag{7.10}$$
$$\mathrm{eq}(146;2;3;5;7) = \mathrm{eq}(14;2;3;5;6;7) \vee \mathrm{eq}(1;2;3;46;5;7) \in S \text{ by (7.5) and (7.7)}, \tag{7.11}$$
$$\mathrm{eq}(14;26;3;5;7) = \mathrm{eq}(134;256;7) \wedge \mathrm{eq}(1246;357) \in S \text{ by (7.1) and (7.9)}, \tag{7.12}$$
$$\mathrm{eq}(1;27;3;46;5) = \mathrm{eq}(146;27;3;5) \wedge \mathrm{eq}(12357;46) \in S \text{ by (7.2) and (7.10)}, \tag{7.13}$$
$$\mathrm{eq}(134567;2) = \mathrm{eq}(135;2;4;67) \vee \mathrm{eq}(146;2;3;5;7) \in S \text{ by (7.3) and (7.11)}, \tag{7.14}$$
$$\mathrm{eq}(12467;3;5) = \mathrm{eq}(146;27;3;5) \vee \mathrm{eq}(14;26;3;5;7) \in S \text{ by (7.2) and (7.12)}, \tag{7.15}$$
$$\mathrm{eq}(1;2;357;46) = \mathrm{eq}(12;357;46) \wedge \mathrm{eq}(134567;2) \in S \text{ by (7.4) and (7.14)}, \tag{7.16}$$
$$\mathrm{eq}(1345;267) = \mathrm{eq}(135;2;4;67) \vee \mathrm{eq}(14;26;3;5;7) \in S \text{ by (7.3) and (7.12)}, \tag{7.17}$$
$$\mathrm{eq}(1;2;3;4;5;67) = \mathrm{eq}(135;2;4;67) \wedge \mathrm{eq}(12467;3;5) \in S \text{ by (7.3) and (7.15)}, \tag{7.18}$$
$$\mathrm{eq}(1;2357;46) = \mathrm{eq}(1;27;3;46;5) \vee \mathrm{eq}(1;2;357;46) \in S \text{ by (7.13) and (7.16)}, \tag{7.19}$$
$$\mathrm{eq}(1;27;3;4;5;6) = \mathrm{eq}(1;27;3;46;5) \wedge \mathrm{eq}(1345;267) \in S \text{ by (7.13) and (7.17)}, \tag{7.20}$$
$$\mathrm{eq}(1;25;3;4;6;7) = \mathrm{eq}(134;256;7) \wedge \mathrm{eq}(1;2357;46) \in S \text{ by (7.1) and (7.19)}. \tag{7.21}$$

Hence, the graph $G(S)$ of S contains the following edges: $(1,3)$ by (7.6), $(3,5)$ by (7.8), $(5,2)$ by (7.21), $(2,7)$ by (7.20), $(7,6)$ by (7.18), $(6,4)$ by (7.7), and $(4,1)$ by (7.5). So 1, 3, 5, 2, 7, 6, 4 is a Hamiltonian cycle in $G(S)$. Hence, like in the proof of Lemma 5, a reference to Lemma 1 completes the proof of Lemma 6. ☐

Proof. (Proof of Lemma 7) The validity of (3.2), (3.3), and (3.4) is trivial. The equalities for the heights of α, …, δ are trivial, too. As in the earlier proofs, S denotes the sublattice generated by $\{\alpha,\beta,\gamma,\delta\}$. The following partitions belong to S:

$$\alpha = \mathrm{eq}(134;258;67), \tag{7.22}$$
$$\beta = \mathrm{eq}(14;2;36;578), \tag{7.23}$$
$$\gamma = \mathrm{eq}(17;25;348;6), \tag{7.24}$$
$$\delta = \mathrm{eq}(12;378;456), \tag{7.25}$$
$$\mathrm{eq}(14;2;3;58;6;7) = \mathrm{eq}(134;258;67) \wedge \mathrm{eq}(14;2;36;578) \text{ by (7.22) and (7.23)}, \tag{7.26}$$
$$\mathrm{eq}(1;25;34;6;7;8) = \mathrm{eq}(134;258;67) \wedge \mathrm{eq}(17;25;348;6) \text{ by (7.22) and (7.24)}, \tag{7.27}$$
$$\mathrm{eq}(1;2;3;4;5;6;78) = \mathrm{eq}(14;2;36;578) \wedge \mathrm{eq}(12;378;456) \text{ by (7.23) and (7.25)}, \tag{7.28}$$
$$\mathrm{eq}(1;2;38;4;5;6;7) = \mathrm{eq}(12;378;456) \wedge \mathrm{eq}(17;25;348;6) \text{ by (7.25) and (7.24)}, \tag{7.29}$$
$$\mathrm{eq}(134;25678) = \mathrm{eq}(134;258;67) \vee \mathrm{eq}(1;2;3;4;5;6;78) \text{ by (7.22) and (7.28)}, \tag{7.30}$$
$$\mathrm{eq}(123458;67) = \mathrm{eq}(134;258;67) \vee \mathrm{eq}(1;2;38;4;5;6;7) \text{ by (7.22) and (7.29)}, \tag{7.31}$$
$$\mathrm{eq}(1346;2578) = \mathrm{eq}(14;2;36;578) \vee \mathrm{eq}(1;25;34;6;7;8) \text{ by (7.23) and (7.27)}, \tag{7.32}$$
$$\mathrm{eq}(14;2;35678) = \mathrm{eq}(14;2;36;578) \vee \mathrm{eq}(1;2;38;4;5;6;7) \text{ by (7.23) and (7.29)}, \tag{7.33}$$
$$\mathrm{eq}(13478;25;6) = \mathrm{eq}(17;25;348;6) \vee \mathrm{eq}(1;2;3;4;5;6;78) \text{ by (7.24) and (7.28)}, \tag{7.34}$$
$$\mathrm{eq}(14;2;358;6;7) = \mathrm{eq}(14;2;3;58;6;7) \vee \mathrm{eq}(1;2;38;4;5;6;7) \text{ by (7.26) and (7.29)}, \tag{7.35}$$
$$\mathrm{eq}(1;25;348;6;7) = \mathrm{eq}(1;25;34;6;7;8) \vee \mathrm{eq}(1;2;38;4;5;6;7) \text{ by (7.27) and (7.29)}, \tag{7.36}$$
$$\mathrm{eq}(14;2;3;58;67) = \mathrm{eq}(134;258;67) \wedge \mathrm{eq}(14;2;35678) \text{ by (7.22) and (7.33)}, \tag{7.37}$$

$$\mathrm{eq}(134;25;6;7;8)=\mathrm{eq}(134;258;67)\wedge\mathrm{eq}(13478;25;6) \text{ by (7.22) and (7.34)}, \tag{7.38}$$
$$\mathrm{eq}(14;2;3;5;6;78)=\mathrm{eq}(14;2;36;578)\wedge\mathrm{eq}(13478;25;6) \text{ by (7.23) and (7.34)}, \tag{7.39}$$
$$\mathrm{eq}(1;2;3;4;56;78)=\mathrm{eq}(12;378;456)\wedge\mathrm{eq}(134;25678) \text{ by (7.25) and (7.30)}, \tag{7.40}$$
$$\mathrm{eq}(12;38;45;6;7)=\mathrm{eq}(12;378;456)\wedge\mathrm{eq}(123458;67) \text{ by (7.25) and (7.31)}, \tag{7.41}$$
$$\mathrm{eq}(1;2;3;46;5;78)=\mathrm{eq}(12;378;456)\wedge\mathrm{eq}(1346;2578) \text{ by (7.25) and (7.32)}, \tag{7.42}$$
$$\mathrm{eq}(14;2;3;5;6;7;8)=\mathrm{eq}(14;2;3;58;6;7)\wedge\mathrm{eq}(13478;25;6) \text{ by (7.26) and (7.34)}, \tag{7.43}$$
$$\mathrm{eq}(12456;378)=\mathrm{eq}(12;378;456)\vee\mathrm{eq}(14;2;3;5;6;78) \text{ by (7.25) and (7.39)}, \tag{7.44}$$
$$\mathrm{eq}(1;256;34;78)=\mathrm{eq}(1;25;34;6;7;8)\vee\mathrm{eq}(1;2;3;4;56;78) \text{ by (7.27) and (7.40)}, \tag{7.45}$$
$$\mathrm{eq}(1;25;34678)=\mathrm{eq}(1;25;348;6;7)\vee\mathrm{eq}(1;2;3;46;5;78) \text{ by (7.36) and (7.42)}, \tag{7.46}$$
$$\mathrm{eq}(145678;2;3)=\mathrm{eq}(14;2;3;58;67)\vee\mathrm{eq}(1;2;3;46;5;78) \text{ by (7.37) and (7.42)}, \tag{7.47}$$
$$\mathrm{eq}(1;2;3;456;78)=\mathrm{eq}(1;2;3;4;56;78)\vee\mathrm{eq}(1;2;3;46;5;78) \text{ by (7.40) and (7.42)}, \tag{7.48}$$
$$\mathrm{eq}(17;2;3;48;5;6)=\mathrm{eq}(17;25;348;6)\wedge\mathrm{eq}(145678;2;3) \text{ by (7.24) and (7.47)}, \tag{7.49}$$
$$\mathrm{eq}(1;25;3;4;6;7;8)=\mathrm{eq}(1;25;34;6;7;8)\wedge\mathrm{eq}(12456;378) \text{ by (7.27) and (7.44)}, \tag{7.50}$$
$$\mathrm{eq}(1;23456;78)=\mathrm{eq}(1;25;34;6;7;8)\vee\mathrm{eq}(1;2;3;456;78) \text{ by (7.27) and (7.48)}, \tag{7.51}$$
$$\mathrm{eq}(1;2;3;48;5;6;7)=\mathrm{eq}(1;25;348;6;7)\wedge\mathrm{eq}(145678;2;3) \text{ by (7.36) and (7.47)}, \tag{7.52}$$
$$\mathrm{eq}(1;2;3;4;5;67;8)=\mathrm{eq}(14;2;3;58;67)\wedge\mathrm{eq}(1;25;34678) \text{ by (7.37) and (7.46)}, \tag{7.53}$$
$$\mathrm{eq}(123456;78)=\mathrm{eq}(134;25;6;7;8)\vee\mathrm{eq}(1;2;3;456;78) \text{ by (7.38) and (7.48)}, \tag{7.54}$$
$$\mathrm{eq}(1;2;35;4;6;7;8)=\mathrm{eq}(14;2;358;6;7)\wedge\mathrm{eq}(1;23456;78) \text{ by (7.35) and (7.51)}, \tag{7.55}$$
$$\mathrm{eq}(12;3;45;6;7;8)=\mathrm{eq}(12;38;45;6;7)\wedge\mathrm{eq}(123456;78) \text{ by (7.41) and (7.54)}, \tag{7.56}$$
$$\mathrm{eq}(167;2;3;48;5)=\mathrm{eq}(17;2;3;48;5;6)\vee\mathrm{eq}(1;2;3;4;5;67;8) \text{ by (7.49) and (7.53)}, \tag{7.57}$$
$$\mathrm{eq}(16;2;3;4;5;7;8)=\mathrm{eq}(1346;2578)\wedge\mathrm{eq}(167;2;3;48;5) \text{ by (7.32) and (7.57)}, \tag{7.58}$$
$$\mathrm{eq}(1267;3;458)=\mathrm{eq}(12;3;45;6;7;8)\vee\mathrm{eq}(167;2;3;48;5) \text{ by (7.56) and (7.57)}, \tag{7.59}$$
$$\mathrm{eq}(1467;2;3;58)=\mathrm{eq}(14;2;3;58;67)\vee\mathrm{eq}(16;2;3;4;5;7;8) \text{ by (7.37) and (7.58)}, \tag{7.60}$$
$$\mathrm{eq}(1;26;3;4;5;7;8)=\mathrm{eq}(1;256;34;78)\wedge\mathrm{eq}(1267;3;458) \text{ by (7.45) and (7.59)}, \tag{7.61}$$
$$\mathrm{eq}(17;2;3;4;5;6;8)=\mathrm{eq}(17;25;348;6)\wedge\mathrm{eq}(1467;2;3;58) \text{ by (7.24) and (7.60)}. \tag{7.62}$$

Now 1, 4, 8, 3, 5, 2, 6, 7 is a Hamiltonian cycle, since $G(S)$ contains the following edges: $(1,4)$ by (7.43), $(4,8)$ by (7.52), $(8,3)$ by (7.29), $(3,5)$ by (7.55), $(5,2)$ by (7.50) , $(2,6)$ by (7.61) , $(6,7)$ by (7.53), and $(7,1)$ by (7.62). Hence, Lemma 1 applies, completing the proof of Lemma 7. □

Proof. (Proof of Lemma 8) Again, (3.2), (3.3), (3.4), and the equalities for the heights are trivial. Let S denote the sublattice generated by $\{\alpha,\beta,\gamma,\delta\}$; it contains the following partitions:

$$\alpha=\mathrm{eq}(178;249;356), \tag{7.63}$$
$$\beta=\mathrm{eq}(19;26;378;45), \tag{7.64}$$
$$\gamma=\mathrm{eq}(1;28;359;467), \tag{7.65}$$
$$\delta=\mathrm{eq}(169;258;347), \tag{7.66}$$
$$\mathrm{eq}(1;2;3;4;5;6;78;9)=\mathrm{eq}(178;249;356)\wedge\mathrm{eq}(19;26;378;45) \text{ by (7.63) and (7.64)}, \tag{7.67}$$
$$\mathrm{eq}(19;2;37;4;5;6;8)=\mathrm{eq}(19;26;378;45)\wedge\mathrm{eq}(169;258;347) \text{ by (7.64) and (7.66)}, \tag{7.68}$$
$$\mathrm{eq}(1;28;3;47;5;6;9)=\mathrm{eq}(169;258;347)\wedge\mathrm{eq}(1;28;359;467) \text{ by (7.66) and (7.65)}, \tag{7.69}$$

$$\mathrm{eq}(124789;356)=\mathrm{eq}(178;249;356)\vee\mathrm{eq}(1;28;3;47;5;6;9) \text{ by (7.63) and (7.69),} \tag{7.70}$$
$$\mathrm{eq}(19;2345678)=\mathrm{eq}(19;26;378;45)\vee\mathrm{eq}(1;28;3;47;5;6;9) \text{ by (7.64) and (7.69),} \tag{7.71}$$
$$\mathrm{eq}(1;24678;359)=\mathrm{eq}(1;28;359;467)\vee\mathrm{eq}(1;2;3;4;5;6;78;9) \text{ by (7.65) and (7.67),} \tag{7.72}$$
$$\mathrm{eq}(1345679;28)=\mathrm{eq}(1;28;359;467)\vee\mathrm{eq}(19;2;37;4;5;6;8) \text{ by (7.65) and (7.68),} \tag{7.73}$$
$$\mathrm{eq}(1;2478;3;5;6;9)=\mathrm{eq}(1;2;3;4;5;6;78;9)\vee\mathrm{eq}(1;28;3;47;5;6;9) \text{ by (7.67), (7.69),} \tag{7.74}$$
$$\mathrm{eq}(19;28;347;5;6)=\mathrm{eq}(19;2;37;4;5;6;8)\vee\mathrm{eq}(1;28;3;47;5;6;9) \text{ by (7.68) and (7.69),} \tag{7.75}$$
$$\mathrm{eq}(1;24;356;78;9)=\mathrm{eq}(178;249;356)\wedge\mathrm{eq}(19;2345678) \text{ by (7.63) and (7.71),} \tag{7.76}$$
$$\mathrm{eq}(17;2;356;49;8)=\mathrm{eq}(178;249;356)\wedge\mathrm{eq}(1345679;28) \text{ by (7.63) and (7.73),} \tag{7.77}$$
$$\mathrm{eq}(1;24;3;5;6;78;9)=\mathrm{eq}(178;249;356)\wedge\mathrm{eq}(1;2478;3;5;6;9) \text{ by (7.63) and (7.74),} \tag{7.78}$$
$$\mathrm{eq}(1;26;3;4;5;78;9)=\mathrm{eq}(19;26;378;45)\wedge\mathrm{eq}(1;24678;359) \text{ by (7.64) and (7.72),} \tag{7.79}$$
$$\mathrm{eq}(19;2;37;45;6;8)=\mathrm{eq}(19;26;378;45)\wedge\mathrm{eq}(1345679;28) \text{ by (7.64) and (7.73),} \tag{7.80}$$
$$\mathrm{eq}(19;28;3;47;5;6)=\mathrm{eq}(169;258;347)\wedge\mathrm{eq}(124789;356) \text{ by (7.66) and (7.70),} \tag{7.81}$$
$$\mathrm{eq}(169;28;347;5)=\mathrm{eq}(169;258;347)\wedge\mathrm{eq}(1345679;28) \text{ by (7.66) and (7.73),} \tag{7.82}$$
$$\mathrm{eq}(19;2;3;4;5;6;7;8)=\mathrm{eq}(19;2;37;4;5;6;8)\wedge\mathrm{eq}(124789;356) \text{ by (7.68) and (7.70),} \tag{7.83}$$
$$\mathrm{eq}(178;234569)=\mathrm{eq}(178;249;356)\vee\mathrm{eq}(1;26;3;4;5;78;9) \text{ by (7.63) and (7.79),} \tag{7.84}$$
$$\mathrm{eq}(19;2456;378)=\mathrm{eq}(19;26;378;45)\vee\mathrm{eq}(1;24;3;5;6;78;9) \text{ by (7.64) and (7.78),} \tag{7.85}$$
$$\mathrm{eq}(19;24;35678)=\mathrm{eq}(19;2;37;4;5;6;8)\vee\mathrm{eq}(1;24;356;78;9) \text{ by (7.68) and (7.76),} \tag{7.86}$$
$$\mathrm{eq}(19;234678;5)=\mathrm{eq}(19;28;347;5;6)\vee\mathrm{eq}(1;26;3;4;5;78;9) \text{ by (7.75) and (7.79),} \tag{7.87}$$
$$\mathrm{eq}(178;2356;49)=\mathrm{eq}(17;2;356;49;8)\vee\mathrm{eq}(1;26;3;4;5;78;9) \text{ by (7.77) and (7.79),} \tag{7.88}$$
$$\mathrm{eq}(19;24678;3;5)=\mathrm{eq}(1;26;3;4;5;78;9)\vee\mathrm{eq}(19;28;3;47;5;6) \text{ by (7.79) and (7.81),} \tag{7.89}$$
$$\mathrm{eq}(1;25;34;69;7;8)=\mathrm{eq}(169;258;347)\wedge\mathrm{eq}(178;234569) \text{ by (7.66) and (7.84),} \tag{7.90}$$
$$\mathrm{eq}(19;25;37;4;6;8)=\mathrm{eq}(169;258;347)\wedge\mathrm{eq}(19;2456;378) \text{ by (7.66) and (7.85),} \tag{7.91}$$
$$\mathrm{eq}(1;25;3;4;6;7;8;9)=\mathrm{eq}(169;258;347)\wedge\mathrm{eq}(178;2356;49) \text{ by (7.66) and (7.88),} \tag{7.92}$$
$$\mathrm{eq}(1;2;36;4;5;7;8;9)=\mathrm{eq}(17;2;356;49;8)\wedge\mathrm{eq}(19;234678;5) \text{ by (7.77) and (7.87),} \tag{7.93}$$
$$\mathrm{eq}(1;2;3;45;6;7;8;9)=\mathrm{eq}(19;2;37;45;6;8)\wedge\mathrm{eq}(178;234569) \text{ by (7.80) and (7.84),} \tag{7.94}$$
$$\mathrm{eq}(1;2;34;5;69;7;8)=\mathrm{eq}(169;28;347;5)\wedge\mathrm{eq}(178;234569) \text{ by (7.82) and (7.84),} \tag{7.95}$$
$$\mathrm{eq}(19;24;3;5;678)=\mathrm{eq}(19;24;35678)\wedge\mathrm{eq}(19;24678;3;5) \text{ by (7.86) and (7.89),} \tag{7.96}$$
$$\mathrm{eq}(1;23589;467)=\mathrm{eq}(1;28;359;467)\vee\mathrm{eq}(1;25;3;4;6;7;8;9) \text{ by (7.65) and (7.92),} \tag{7.97}$$
$$\mathrm{eq}(17;2356;49;8)=\mathrm{eq}(17;2;356;49;8)\vee\mathrm{eq}(1;25;3;4;6;7;8;9) \text{ by (7.77) and (7.92),} \tag{7.98}$$
$$\mathrm{eq}(1;2569;34;78)=\mathrm{eq}(1;26;3;4;5;78;9)\vee\mathrm{eq}(1;25;34;69;7;8) \text{ by (7.79) and (7.90),} \tag{7.99}$$
$$\mathrm{eq}(19;245;37;6;8)=\mathrm{eq}(19;2;37;45;6;8)\vee\mathrm{eq}(19;25;37;4;6;8) \text{ by (7.80) and (7.91),} \tag{7.100}$$
$$\mathrm{eq}(16789;234;5)=\mathrm{eq}(1;2;34;5;69;7;8)\vee\mathrm{eq}(19;24;3;5;678) \text{ by (7.95) and (7.96),} \tag{7.101}$$
$$\mathrm{eq}(1;24;3;5;6;7;8;9)=\mathrm{eq}(178;249;356)\wedge\mathrm{eq}(19;245;37;6;8) \text{ by (7.63) and (7.100),} \tag{7.102}$$
$$\mathrm{eq}(1;2;38;4;5;6;7;9)=\mathrm{eq}(19;26;378;45)\wedge\mathrm{eq}(1;23589;467) \text{ by (7.64) and (7.97),} \tag{7.103}$$
$$\mathrm{eq}(1;26;3;4;5;7;8;9)=\mathrm{eq}(19;26;378;45)\wedge\mathrm{eq}(17;2356;49;8) \text{ by (7.64) and (7.98),} \tag{7.104}$$
$$\mathrm{eq}(1;2;3;4;59;6;7;8)=\mathrm{eq}(1;28;359;467)\wedge\mathrm{eq}(1;2569;34;78) \text{ by (7.65) and (7.99),} \tag{7.105}$$
$$\mathrm{eq}(17;2;3;4;5;6;8;9)=\mathrm{eq}(17;2;356;49;8)\wedge\mathrm{eq}(16789;234;5) \text{ by (7.77) and (7.101).} \tag{7.106}$$

Since $G(S)$ contains the following edges: $(1,7)$ by (7.106), $(7,8)$ by (7.67), $(8,3)$ by (7.103), $(3,6)$ by (7.93) , $(6,2)$ by (7.104), $(2,4)$ by (7.102), $(4,5)$ by (7.94) , $(5,9)$ by (7.105), and $(9,1)$ by (7.83), Lemma 1 is applicable and completes the proof of Lemma 8. □

P r o o f. (Proof of Lemma 9) Let S be the sublattice generated by $\{\alpha,\beta,\gamma,\delta\}$. Then the following partitions belong to S:

$$\alpha = \mathrm{eq}(134;25), \tag{7.107}$$
$$\beta = \mathrm{eq}(13;245), \tag{7.108}$$
$$\gamma = \mathrm{eq}(12;345), \tag{7.109}$$
$$\delta = \mathrm{eq}(124;35), \tag{7.110}$$
$$\mathrm{eq}(1;2;34;5) = \mathrm{eq}(134;25) \wedge \mathrm{eq}(12;345) \text{ by (7.107) and (7.109)}, \tag{7.111}$$
$$\mathrm{eq}(14;2;3;5) = \mathrm{eq}(134;25) \wedge \mathrm{eq}(124;35) \text{ by (7.107) and (7.110)}, \tag{7.112}$$
$$\mathrm{eq}(1;2;3;45) = \mathrm{eq}(13;245) \wedge \mathrm{eq}(12;345) \text{ by (7.108) and (7.109)}, \tag{7.113}$$
$$\mathrm{eq}(1;24;3;5) = \mathrm{eq}(13;245) \wedge \mathrm{eq}(124;35) \text{ by (7.108) and (7.110)}, \tag{7.114}$$
$$\mathrm{eq}(134;2;5) = \mathrm{eq}(1;2;34;5) \vee \mathrm{eq}(14;2;3;5) \text{ by (7.111) and (7.112)}, \tag{7.115}$$
$$\mathrm{eq}(124;3;5) = \mathrm{eq}(14;2;3;5) \vee \mathrm{eq}(1;24;3;5) \text{ by (7.112) and (7.114)}, \tag{7.116}$$
$$\mathrm{eq}(1;245;3) = \mathrm{eq}(1;2;3;45) \vee \mathrm{eq}(1;24;3;5) \text{ by (7.113) and (7.114)}, \tag{7.117}$$
$$\mathrm{eq}(1;25;3;4) = \mathrm{eq}(134;25) \wedge \mathrm{eq}(1;245;3) \text{ by (7.107) and (7.117)}, \tag{7.118}$$
$$\mathrm{eq}(13;2;4;5) = \mathrm{eq}(13;245) \wedge \mathrm{eq}(134;2;5) \text{ by (7.108) and (7.115)}, \tag{7.119}$$
$$\mathrm{eq}(12;3;4;5) = \mathrm{eq}(12;345) \wedge \mathrm{eq}(124;3;5) \text{ by (7.109) and (7.116)}. \tag{7.120}$$

Hence, the graph $G(S)$ of S contains the following edges: $(1,2)$ by (7.120), $(2,5)$ by (7.118), $(5,4)$ by (7.113), $(4,3)$ by (7.111), and $(3,1)$ by (7.119). Thus, a reference to Lemma 1 completes the proof of Lemma 9. □

P r o o f. (Proof of Lemma 10) Now the sublattice S generated by $\{\alpha,\beta,\gamma,\delta\}$ contains

$$\alpha = \mathrm{eq}(12;34;5;6), \tag{7.121}$$
$$\beta = \mathrm{eq}(1;2;35;46), \tag{7.122}$$
$$\gamma = \mathrm{eq}(1;25;36;4), \tag{7.123}$$
$$\delta = \mathrm{eq}(15;24;3;6), \tag{7.124}$$
$$\mathrm{eq}(12;3456) = \mathrm{eq}(12;34;5;6) \vee \mathrm{eq}(1;2;35;46) \text{ by (7.121) and (7.122)}, \tag{7.125}$$
$$\mathrm{eq}(125;346) = \mathrm{eq}(12;34;5;6) \vee \mathrm{eq}(1;25;36;4) \text{ by (7.121) and (7.123)}, \tag{7.126}$$
$$\mathrm{eq}(12345;6) = \mathrm{eq}(12;34;5;6) \vee \mathrm{eq}(15;24;3;6) \text{ by (7.121) and (7.124)}, \tag{7.127}$$
$$\mathrm{eq}(1;23456) = \mathrm{eq}(1;2;35;46) \vee \mathrm{eq}(1;25;36;4) \text{ by (7.122) and (7.123)}, \tag{7.128}$$
$$\mathrm{eq}(1245;36) = \mathrm{eq}(1;25;36;4) \vee \mathrm{eq}(15;24;3;6) \text{ by (7.123) and (7.124)}, \tag{7.129}$$
$$\mathrm{eq}(12;3;4;5;6) = \mathrm{eq}(12;34;5;6) \wedge \mathrm{eq}(1245;36) \text{ by (7.121) and (7.129)}, \tag{7.130}$$
$$\mathrm{eq}(1;2;3;46;5) = \mathrm{eq}(1;2;35;46) \wedge \mathrm{eq}(125;346) \text{ by (7.122) and (7.126)}, \tag{7.131}$$
$$\mathrm{eq}(1;2;35;4;6) = \mathrm{eq}(1;2;35;46) \wedge \mathrm{eq}(12345;6) \text{ by (7.122) and (7.127)}, \tag{7.132}$$
$$\mathrm{eq}(1;2;36;4;5) = \mathrm{eq}(1;25;36;4) \wedge \mathrm{eq}(12;3456) \text{ by (7.123) and (7.125)}, \tag{7.133}$$
$$\mathrm{eq}(15;2;3;4;6) = \mathrm{eq}(15;24;3;6) \wedge \mathrm{eq}(125;346) \text{ by (7.124) and (7.126)}, \tag{7.134}$$

$$\mathrm{eq}(1;24;3;5;6) = \mathrm{eq}(15;24;3;6) \wedge \mathrm{eq}(1;23456) \text{ by (7.124) and (7.128).} \tag{7.135}$$

Since $G(S)$ contains the edges $(1,2)$ by (7.130), $(2,4)$ by (7.135), $(4,6)$ by (7.131), $(6,3)$ by (7.133), $(3,5)$ by (7.132), and $(5,1)$ by (7.134), Lemma 1 is applicable and completes the proof of Lemma 10. □

P r o o f. (Proof of Lemma 11) Now the sublattice S generated by $\{\alpha,\beta,\gamma,\delta\}$ contains

$$\alpha = \mathrm{eq}(13;256;4), \tag{7.136}$$
$$\beta = \mathrm{eq}(156;2;34), \tag{7.137}$$
$$\gamma = \mathrm{eq}(12;35;46), \tag{7.138}$$
$$\delta = \mathrm{eq}(13;246;5), \tag{7.139}$$
$$\mathrm{eq}(1;2;3;4;56) = \mathrm{eq}(13;256;4) \wedge \mathrm{eq}(156;2;34) \text{ by (7.136) and (7.137),} \tag{7.140}$$
$$\mathrm{eq}(13;26;4;5) = \mathrm{eq}(13;256;4) \wedge \mathrm{eq}(13;246;5) \text{ by (7.136) and (7.139),} \tag{7.141}$$
$$\mathrm{eq}(1;2;3;46;5) = \mathrm{eq}(12;35;46) \wedge \mathrm{eq}(13;246;5) \text{ by (7.138) and (7.139),} \tag{7.142}$$
$$\mathrm{eq}(13456;2) = \mathrm{eq}(156;2;34) \vee \mathrm{eq}(1;2;3;46;5) \text{ by (7.137) and (7.142),} \tag{7.143}$$
$$\mathrm{eq}(12;3456) = \mathrm{eq}(12;35;46) \vee \mathrm{eq}(1;2;3;4;56) \text{ by (7.138) and (7.140),} \tag{7.144}$$
$$\mathrm{eq}(1;2;3;456) = \mathrm{eq}(1;2;3;4;56) \vee \mathrm{eq}(1;2;3;46;5) \text{ by (7.140) and (7.142),} \tag{7.145}$$
$$\mathrm{eq}(13;2;4;56) = \mathrm{eq}(13;256;4) \wedge \mathrm{eq}(13456;2) \text{ by (7.136) and (7.143),} \tag{7.146}$$
$$\mathrm{eq}(1;2;34;56) = \mathrm{eq}(156;2;34) \wedge \mathrm{eq}(12;3456) \text{ by (7.137) and (7.144),} \tag{7.147}$$
$$\mathrm{eq}(13;2;46;5) = \mathrm{eq}(13;246;5) \wedge \mathrm{eq}(13456;2) \text{ by (7.139) and (7.143),} \tag{7.148}$$
$$\mathrm{eq}(13;2;4;5;6) = \mathrm{eq}(13;26;4;5) \wedge \mathrm{eq}(13456;2) \text{ by (7.141) and (7.143),} \tag{7.149}$$
$$\mathrm{eq}(134;256) = \mathrm{eq}(13;256;4) \vee \mathrm{eq}(1;2;34;56) \text{ by (7.136) and (7.147),} \tag{7.150}$$
$$\mathrm{eq}(1235;46) = \mathrm{eq}(12;35;46) \vee \mathrm{eq}(13;2;46;5) \text{ by (7.138) and (7.148),} \tag{7.151}$$
$$\mathrm{eq}(134;2;56) = \mathrm{eq}(13;2;4;56) \vee \mathrm{eq}(1;2;34;56) \text{ by (7.146) and (7.147),} \tag{7.152}$$
$$\mathrm{eq}(15;2;3;4;6) = \mathrm{eq}(156;2;34) \wedge \mathrm{eq}(1235;46) \text{ by (7.137) and (7.151),} \tag{7.153}$$
$$\mathrm{eq}(12356;4) = \mathrm{eq}(13;256;4) \vee \mathrm{eq}(15;2;3;4;6) \text{ by (7.136) and (7.153),} \tag{7.154}$$
$$\mathrm{eq}(135;26;4) = \mathrm{eq}(13;26;4;5) \vee \mathrm{eq}(15;2;3;4;6) \text{ by (7.141) and (7.153),} \tag{7.155}$$
$$\mathrm{eq}(1456;2;3) = \mathrm{eq}(1;2;3;456) \vee \mathrm{eq}(15;2;3;4;6) \text{ by (7.145) and (7.153),} \tag{7.156}$$
$$\mathrm{eq}(12;35;4;6) = \mathrm{eq}(12;35;46) \wedge \mathrm{eq}(12356;4) \text{ by (7.138) and (7.154),} \tag{7.157}$$
$$\mathrm{eq}(1;2;35;4;6) = \mathrm{eq}(12;35;46) \wedge \mathrm{eq}(135;26;4) \text{ by (7.138) and (7.155),} \tag{7.158}$$
$$\mathrm{eq}(14;2;3;56) = \mathrm{eq}(134;256) \wedge \mathrm{eq}(1456;2;3) \text{ by (7.150) and (7.156),} \tag{7.159}$$
$$\mathrm{eq}(124;356) = \mathrm{eq}(12;35;4;6) \vee \mathrm{eq}(14;2;3;56) \text{ by (7.157) and (7.159),} \tag{7.160}$$
$$\mathrm{eq}(1;24;3;5;6) = \mathrm{eq}(13;246;5) \wedge \mathrm{eq}(124;356) \text{ by (7.139) and (7.160),} \tag{7.161}$$
$$\mathrm{eq}(1234;56) = \mathrm{eq}(134;2;56) \vee \mathrm{eq}(1;24;3;5;6) \text{ by (7.152) and (7.161),} \tag{7.162}$$
$$\mathrm{eq}(12;3;4;5;6) = \mathrm{eq}(12;35;46) \wedge \mathrm{eq}(1234;56) \text{ by (7.138) and (7.162).} \tag{7.163}$$

Since $G(S)$ contains the edges $(1,2)$ by (7.163), $(2,4)$ by (7.161), $(4,6)$ by (7.142), $(6,5)$ by (7.140), $(5,3)$ by (7.158), and $(3,1)$ by (7.149), Lemma 1 is applicable and completes the proof of Lemma 11. □

P r o o f. (Proof of Lemma 12) Now the sublattice S generated by $\{\alpha,\beta,\gamma,\delta\}$ contains

$$\alpha = \mathrm{eq}(1;24;35;6;7), \tag{7.164}$$

$$\beta = \text{eq}(14;26;3;5;7), \tag{7.165}$$
$$\gamma = \text{eq}(1;2;34;5;67), \tag{7.166}$$
$$\delta = \text{eq}(17;2;3;4;56), \tag{7.167}$$
$$\text{eq}(1246;35;7) = \text{eq}(1;24;35;6;7) \vee \text{eq}(14;26;3;5;7) \text{ by (7.164) and (7.165)}, \tag{7.168}$$
$$\text{eq}(1;2345;67) = \text{eq}(1;24;35;6;7) \vee \text{eq}(1;2;34;5;67) \text{ by (7.164) and (7.166)}, \tag{7.169}$$
$$\text{eq}(17;24;356) = \text{eq}(1;24;35;6;7) \vee \text{eq}(17;2;3;4;56) \text{ by (7.164) and (7.167)}, \tag{7.170}$$
$$\text{eq}(134;267;5) = \text{eq}(14;26;3;5;7) \vee \text{eq}(1;2;34;5;67) \text{ by (7.165) and (7.166)}, \tag{7.171}$$
$$\text{eq}(147;256;3) = \text{eq}(14;26;3;5;7) \vee \text{eq}(17;2;3;4;56) \text{ by (7.165) and (7.167)}, \tag{7.172}$$
$$\text{eq}(1567;2;34) = \text{eq}(1;2;34;5;67) \vee \text{eq}(17;2;3;4;56) \text{ by (7.166) and (7.167)}, \tag{7.173}$$
$$\text{eq}(16;2;3;4;5;7) = \text{eq}(1246;35;7) \wedge \text{eq}(1567;2;34) \text{ by (7.168) and (7.173)}, \tag{7.174}$$
$$\text{eq}(1;25;3;4;6;7) = \text{eq}(1;2345;67) \wedge \text{eq}(147;256;3) \text{ by (7.169) and (7.172)}, \tag{7.175}$$
$$\text{eq}(1;2345;6;7) = \text{eq}(1;24;35;6;7) \vee \text{eq}(1;25;3;4;6;7) \text{ by (7.164) and (7.175)}, \tag{7.176}$$
$$\text{eq}(1246;3;5;7) = \text{eq}(14;26;3;5;7) \vee \text{eq}(16;2;3;4;5;7) \text{ by (7.165) and (7.174)}, \tag{7.177}$$
$$\text{eq}(14;256;3;7) = \text{eq}(14;26;3;5;7) \vee \text{eq}(1;25;3;4;6;7) \text{ by (7.165) and (7.175)}, \tag{7.178}$$
$$\text{eq}(167;2;34;5) = \text{eq}(1;2;34;5;67) \vee \text{eq}(16;2;3;4;5;7) \text{ by (7.166) and (7.174)}, \tag{7.179}$$
$$\text{eq}(1567;2;3;4) = \text{eq}(17;2;3;4;56) \vee \text{eq}(16;2;3;4;5;7) \text{ by (7.167) and (7.174)}, \tag{7.180}$$
$$\text{eq}(123456;7) = \text{eq}(1246;35;7) \vee \text{eq}(1;25;3;4;6;7) \text{ by (7.168) and (7.175)}, \tag{7.181}$$
$$\text{eq}(13567;24) = \text{eq}(17;24;356) \vee \text{eq}(16;2;3;4;5;7) \text{ by (7.170) and (7.174)}, \tag{7.182}$$
$$\text{eq}(1;24;3;5;6;7) = \text{eq}(1;24;35;6;7) \wedge \text{eq}(1246;3;5;7) \text{ by (7.164) and (7.177)}, \tag{7.183}$$
$$\text{eq}(1;2;34;5;6;7) = \text{eq}(1;2;34;5;67) \wedge \text{eq}(1;2345;6;7) \text{ by (7.166) and (7.176)}, \tag{7.184}$$
$$\text{eq}(1;2;3;4;5;67) = \text{eq}(1;2;34;5;67) \wedge \text{eq}(1567;2;3;4) \text{ by (7.166) and (7.180)}, \tag{7.185}$$
$$\text{eq}(1;2;3;4;56;7) = \text{eq}(17;2;3;4;56) \wedge \text{eq}(14;256;3;7) \text{ by (7.167) and (7.178)}, \tag{7.186}$$
$$\text{eq}(17;2;3;4;5;6) = \text{eq}(17;2;3;4;56) \wedge \text{eq}(167;2;34;5) \text{ by (7.167) and (7.179)}, \tag{7.187}$$
$$\text{eq}(1356;24;7) = \text{eq}(123456;7) \wedge \text{eq}(13567;24) \text{ by (7.181) and (7.182)}, \tag{7.188}$$
$$\text{eq}(13;2;4;5;6;7) = \text{eq}(134;267;5) \wedge \text{eq}(1356;24;7) \text{ by (7.171) and (7.188)}. \tag{7.189}$$

Since $G(S)$ contains the edges $(1,3)$ by (7.189), $(3,4)$ by (7.184), $(4,2)$ by (7.183), $(2,5)$ by (7.175), $(5,6)$ by (7.186), $(6,7)$ by (7.185), $(7,1)$ by (7.187), Lemma 1 is applicable and completes the proof of Lemma 12. □

P r o o f. (Proof of Lemma 13) Now the sublattice S generated by $\{\alpha, \beta, \gamma, \delta\}$ contains

$$\alpha = \text{eq}(13;24;567), \tag{7.190}$$
$$\beta = \text{eq}(125;3;467), \tag{7.191}$$
$$\gamma = \text{eq}(1357;26;4), \tag{7.192}$$
$$\delta = \text{eq}(126;35;47), \tag{7.193}$$
$$\text{eq}(1;2;3;4;5;67) = \text{eq}(13;24;567) \wedge \text{eq}(125;3;467) \text{ by (7.190) and (7.191)}, \tag{7.194}$$
$$\text{eq}(13;2;4;57;6) = \text{eq}(13;24;567) \wedge \text{eq}(1357;26;4) \text{ by (7.190) and (7.192)}, \tag{7.195}$$
$$\text{eq}(15;2;3;4;6;7) = \text{eq}(125;3;467) \wedge \text{eq}(1357;26;4) \text{ by (7.191) and (7.192)}, \tag{7.196}$$
$$\text{eq}(12;3;47;5;6) = \text{eq}(125;3;467) \wedge \text{eq}(126;35;47) \text{ by (7.191) and (7.193)}, \tag{7.197}$$

$$\mathrm{eq}(1;26;35;4;7)=\mathrm{eq}(1357;26;4)\wedge\mathrm{eq}(126;35;47) \text{ by (7.192) and (7.193)}, \tag{7.198}$$
$$\mathrm{eq}(123567;4)=\mathrm{eq}(1357;26;4)\vee\mathrm{eq}(1;2;3;4;5;67) \text{ by (7.192) and (7.194)}, \tag{7.199}$$
$$\mathrm{eq}(12467;35)=\mathrm{eq}(126;35;47)\vee\mathrm{eq}(1;2;3;4;5;67) \text{ by (7.193) and (7.194)}, \tag{7.200}$$
$$\mathrm{eq}(12356;47)=\mathrm{eq}(126;35;47)\vee\mathrm{eq}(15;2;3;4;6;7) \text{ by (7.193) and (7.196)}, \tag{7.201}$$
$$\mathrm{eq}(1357;2;4;6)=\mathrm{eq}(13;2;4;57;6)\vee\mathrm{eq}(15;2;3;4;6;7) \text{ by (7.195) and (7.196)}, \tag{7.202}$$
$$\mathrm{eq}(125;3;47;6)=\mathrm{eq}(15;2;3;4;6;7)\vee\mathrm{eq}(12;3;47;5;6) \text{ by (7.196) and (7.197)}, \tag{7.203}$$
$$\mathrm{eq}(135;26;4;7)=\mathrm{eq}(15;2;3;4;6;7)\vee\mathrm{eq}(1;26;35;4;7) \text{ by (7.196) and (7.198)}, \tag{7.204}$$
$$\mathrm{eq}(1;24;3;5;67)=\mathrm{eq}(13;24;567)\wedge\mathrm{eq}(12467;35) \text{ by (7.190) and (7.200)}, \tag{7.205}$$
$$\mathrm{eq}(13;2;4;56;7)=\mathrm{eq}(13;24;567)\wedge\mathrm{eq}(12356;47) \text{ by (7.190) and (7.201)}, \tag{7.206}$$
$$\mathrm{eq}(13;2;4;5;6;7)=\mathrm{eq}(13;24;567)\wedge\mathrm{eq}(135;26;4;7) \text{ by (7.190) and (7.204)}, \tag{7.207}$$
$$\mathrm{eq}(1;2;35;4;6;7)=\mathrm{eq}(126;35;47)\wedge\mathrm{eq}(1357;2;4;6) \text{ by (7.193) and (7.202)}, \tag{7.208}$$
$$\mathrm{eq}(123457;6)=\mathrm{eq}(13;2;4;57;6)\vee\mathrm{eq}(125;3;47;6) \text{ by (7.195) and (7.203)}, \tag{7.209}$$
$$\mathrm{eq}(12;3;4;5;6;7)=\mathrm{eq}(12;3;47;5;6)\wedge\mathrm{eq}(123567;4) \text{ by (7.197) and (7.199)}, \tag{7.210}$$
$$\mathrm{eq}(124567;3)=\mathrm{eq}(125;3;467)\vee\mathrm{eq}(1;24;3;5;67) \text{ by (7.191) and (7.205)}, \tag{7.211}$$
$$\mathrm{eq}(1;2467;35)=\mathrm{eq}(1;26;35;4;7)\vee\mathrm{eq}(1;24;3;5;67) \text{ by (7.198) and (7.205)}, \tag{7.212}$$
$$\mathrm{eq}(1;24;3;5;6;7)=\mathrm{eq}(1;24;3;5;67)\wedge\mathrm{eq}(123457;6) \text{ by (7.205) and (7.209)}, \tag{7.213}$$
$$\mathrm{eq}(1;2;3;47;5;6)=\mathrm{eq}(12;3;47;5;6)\wedge\mathrm{eq}(1;2467;35) \text{ by (7.197) and (7.212)}, \tag{7.214}$$
$$\mathrm{eq}(1;2;3;4;56;7)=\mathrm{eq}(13;2;4;56;7)\wedge\mathrm{eq}(124567;3) \text{ by (7.206) and (7.211)}. \tag{7.215}$$

Since $G(S)$ contains the edges $(1,2)$ by (7.210), $(2,4)$ by (7.213), $(4,7)$ by (7.214), $(7,6)$ by (7.194), $(6,5)$ by (7.215), $(5,3)$ by (7.208), $(3,1)$ by (7.207), Lemma 1 applies and completes the proof of Lemma 13. □

P r o o f. (Proof of Lemma 14) Now the sublattice S generated by $\{\alpha,\beta,\gamma,\delta\}$ contains

$$\alpha=\mathrm{eq}(18;2;35;4;67), \tag{7.216}$$
$$\beta=\mathrm{eq}(1;24;37;5;68), \tag{7.217}$$
$$\gamma=\mathrm{eq}(16;2;34;57;8), \tag{7.218}$$
$$\delta=\mathrm{eq}(12;3;45;6;78), \tag{7.219}$$
$$\mathrm{eq}(135678;24)=\mathrm{eq}(18;2;35;4;67)\vee\mathrm{eq}(1;24;37;5;68) \text{ by (7.216) and (7.217)}, \tag{7.220}$$
$$\mathrm{eq}(12678;345)=\mathrm{eq}(18;2;35;4;67)\vee\mathrm{eq}(12;3;45;6;78) \text{ by (7.216) and (7.219)}, \tag{7.221}$$
$$\mathrm{eq}(168;23457)=\mathrm{eq}(1;24;37;5;68)\vee\mathrm{eq}(16;2;34;57;8) \text{ by (7.217) and (7.218)}, \tag{7.222}$$
$$\mathrm{eq}(1245;3678)=\mathrm{eq}(1;24;37;5;68)\vee\mathrm{eq}(12;3;45;6;78) \text{ by (7.217) and (7.219)}, \tag{7.223}$$
$$\mathrm{eq}(126;34578)=\mathrm{eq}(16;2;34;57;8)\vee\mathrm{eq}(12;3;45;6;78) \text{ by (7.218) and (7.219)}, \tag{7.224}$$
$$\mathrm{eq}(18;2;35;4;6;7)=\mathrm{eq}(18;2;35;4;67)\wedge\mathrm{eq}(168;23457) \text{ by (7.216) and (7.222)}, \tag{7.225}$$
$$\mathrm{eq}(1;2;3;4;5;67;8)=\mathrm{eq}(18;2;35;4;67)\wedge\mathrm{eq}(1245;3678) \text{ by (7.216) and (7.223)}, \tag{7.226}$$
$$\mathrm{eq}(1;2;35;4;6;7;8)=\mathrm{eq}(18;2;35;4;67)\wedge\mathrm{eq}(126;34578) \text{ by (7.216) and (7.224)}, \tag{7.227}$$
$$\mathrm{eq}(1;2;3;4;5;68;7)=\mathrm{eq}(1;24;37;5;68)\wedge\mathrm{eq}(12678;345) \text{ by (7.217) and (7.221)}, \tag{7.228}$$
$$\mathrm{eq}(16;2;3;4;57;8)=\mathrm{eq}(16;2;34;57;8)\wedge\mathrm{eq}(135678;24) \text{ by (7.218) and (7.220)}, \tag{7.229}$$
$$\mathrm{eq}(16;2;34;5;7;8)=\mathrm{eq}(16;2;34;57;8)\wedge\mathrm{eq}(12678;345) \text{ by (7.218) and (7.221)}, \tag{7.230}$$

$$\mathrm{eq}(1;2;3;45;6;7;8) = \mathrm{eq}(12;3;45;6;78) \wedge \mathrm{eq}(168;23457) \text{ by (7.219) and (7.222)}, \tag{7.231}$$
$$\mathrm{eq}(12;3;45;678) = \mathrm{eq}(12678;345) \wedge \mathrm{eq}(1245;3678) \text{ by (7.221) and (7.223)}, \tag{7.232}$$
$$\mathrm{eq}(1;245;37;68) = \mathrm{eq}(168;23457) \wedge \mathrm{eq}(1245;3678) \text{ by (7.222) and (7.223)}, \tag{7.233}$$
$$\mathrm{eq}(12;378;45;6) = \mathrm{eq}(1245;3678) \wedge \mathrm{eq}(126;34578) \text{ by (7.223) and (7.224)}, \tag{7.234}$$
$$\mathrm{eq}(18;2;345;67) = \mathrm{eq}(18;2;35;4;67) \vee \mathrm{eq}(1;2;3;45;6;7;8) \text{ by (7.216) and (7.231)}, \tag{7.235}$$
$$\mathrm{eq}(168;2347;5) = \mathrm{eq}(1;24;37;5;68) \vee \mathrm{eq}(16;2;34;5;7;8) \text{ by (7.217) and (7.230)}, \tag{7.236}$$
$$\mathrm{eq}(1567;2;34;8) = \mathrm{eq}(16;2;34;57;8) \vee \mathrm{eq}(1;2;3;4;5;67;8) \text{ by (7.218) and (7.226)}, \tag{7.237}$$
$$\mathrm{eq}(1278;345;6) = \mathrm{eq}(12;3;45;6;78) \vee \mathrm{eq}(18;2;35;4;6;7) \text{ by (7.219) and (7.225)}, \tag{7.238}$$
$$\mathrm{eq}(1234578;6) = \mathrm{eq}(18;2;35;4;6;7) \vee \mathrm{eq}(12;378;45;6) \text{ by (7.225) and (7.234)}, \tag{7.239}$$
$$\mathrm{eq}(1;23457;68) = \mathrm{eq}(1;2;35;4;6;7;8) \vee \mathrm{eq}(1;245;37;68) \text{ by (7.227) and (7.233)}, \tag{7.240}$$
$$\mathrm{eq}(1245678;3) = \mathrm{eq}(16;2;3;4;57;8) \vee \mathrm{eq}(12;3;45;678) \text{ by (7.229) and (7.232)}, \tag{7.241}$$
$$\mathrm{eq}(18;2;3;4;5;6;7) = \mathrm{eq}(18;2;35;4;67) \wedge \mathrm{eq}(168;2347;5) \text{ by (7.216) and (7.236)}, \tag{7.242}$$
$$\mathrm{eq}(1;2;34;5;6;7;8) = \mathrm{eq}(16;2;34;57;8) \wedge \mathrm{eq}(18;2;345;67) \text{ by (7.218) and (7.235)}, \tag{7.243}$$
$$\mathrm{eq}(18;27;34;5;6) = \mathrm{eq}(168;2347;5) \wedge \mathrm{eq}(1278;345;6) \text{ by (7.236) and (7.238)}, \tag{7.244}$$
$$\mathrm{eq}(157;2;34;6;8) = \mathrm{eq}(1567;2;34;8) \wedge \mathrm{eq}(1234578;6) \text{ by (7.237) and (7.239)}, \tag{7.245}$$
$$\mathrm{eq}(124578;3;6) = \mathrm{eq}(1234578;6) \wedge \mathrm{eq}(1245678;3) \text{ by (7.239) and (7.241)}, \tag{7.246}$$
$$\mathrm{eq}(1;2457;3;68) = \mathrm{eq}(1;23457;68) \wedge \mathrm{eq}(1245678;3) \text{ by (7.240) and (7.241)}, \tag{7.247}$$
$$\mathrm{eq}(1;24;3;5;6;7;8) = \mathrm{eq}(1;24;37;5;68) \wedge \mathrm{eq}(124578;3;6) \text{ by (7.217) and (7.246)}, \tag{7.248}$$
$$\mathrm{eq}(15;2;3;4;6;7;8) = \mathrm{eq}(1245;3678) \wedge \mathrm{eq}(157;2;34;6;8) \text{ by (7.223) and (7.245)}, \tag{7.249}$$
$$\mathrm{eq}(1;27;3;4;5;6;8) = \mathrm{eq}(18;27;34;5;6) \wedge \mathrm{eq}(1;2457;3;68) \text{ by (7.244) and (7.247)}. \tag{7.250}$$

Since $G(S)$ contains the edges $(1,8)$ by (7.242), $(8,6)$ by (7.228), $(6,7)$ by (7.226), $(7,2)$ by (7.250), $(2,4)$ by (7.248), $(4,3)$ by (7.243), $(3,5)$ by (7.227), $(5,1)$ by (7.249), Lemma 1 applies and completes the proof of Lemma 14. □

Proof. (Proof of Lemma 15) Now the sublattice S generated by $\{\alpha,\beta,\gamma,\delta\}$ contains

$$\alpha = \mathrm{eq}(137;246;58), \tag{7.251}$$
$$\beta = \mathrm{eq}(146;257;38), \tag{7.252}$$
$$\gamma = \mathrm{eq}(136;2;4578), \tag{7.253}$$
$$\delta = \mathrm{eq}(1245;37;68), \tag{7.254}$$
$$\mathrm{eq}(1;2;3;46;5;7;8) = \mathrm{eq}(137;246;58) \wedge \mathrm{eq}(146;257;38) \text{ by (7.251) and (7.252)}, \tag{7.255}$$
$$\mathrm{eq}(13;2;4;58;6;7) = \mathrm{eq}(137;246;58) \wedge \mathrm{eq}(136;2;4578) \text{ by (7.251) and (7.253)}, \tag{7.256}$$
$$\mathrm{eq}(1;24;37;5;6;8) = \mathrm{eq}(137;246;58) \wedge \mathrm{eq}(1245;37;68) \text{ by (7.251) and (7.254)}, \tag{7.257}$$
$$\mathrm{eq}(16;2;3;4;57;8) = \mathrm{eq}(146;257;38) \wedge \mathrm{eq}(136;2;4578) \text{ by (7.252) and (7.253)}, \tag{7.258}$$
$$\mathrm{eq}(14;25;3;6;7;8) = \mathrm{eq}(146;257;38) \wedge \mathrm{eq}(1245;37;68) \text{ by (7.252) and (7.254)}, \tag{7.259}$$
$$\mathrm{eq}(1;2;3;45;6;7;8) = \mathrm{eq}(136;2;4578) \wedge \mathrm{eq}(1245;37;68) \text{ by (7.253) and (7.254)}, \tag{7.260}$$
$$\mathrm{eq}(124567;38) = \mathrm{eq}(146;257;38) \vee \mathrm{eq}(1;2;3;45;6;7;8) \text{ by (7.252) and (7.260)}, \tag{7.261}$$
$$\mathrm{eq}(1345678;2) = \mathrm{eq}(136;2;4578) \vee \mathrm{eq}(1;2;3;46;5;7;8) \text{ by (7.253) and (7.255)}, \tag{7.262}$$
$$\mathrm{eq}(124568;37) = \mathrm{eq}(1245;37;68) \vee \mathrm{eq}(1;2;3;46;5;7;8) \text{ by (7.254) and (7.255)}, \tag{7.263}$$

$$\mathrm{eq}(146;25;3;7;8)=\mathrm{eq}(1;2;3;46;5;7;8)\vee\mathrm{eq}(14;25;3;6;7;8) \text{ by (7.255) and (7.259)}, \tag{7.264}$$
$$\mathrm{eq}(137;24;58;6)=\mathrm{eq}(13;2;4;58;6;7)\vee\mathrm{eq}(1;24;37;5;6;8) \text{ by (7.256) and (7.257)}, \tag{7.265}$$
$$\mathrm{eq}(16;24;357;8)=\mathrm{eq}(1;24;37;5;6;8)\vee\mathrm{eq}(16;2;3;4;57;8) \text{ by (7.257) and (7.258)}, \tag{7.266}$$
$$\mathrm{eq}(1;245;37;6;8)=\mathrm{eq}(1;24;37;5;6;8)\vee\mathrm{eq}(1;2;3;45;6;7;8) \text{ by (7.257) and (7.260)}, \tag{7.267}$$
$$\mathrm{eq}(1245;3;6;7;8)=\mathrm{eq}(14;25;3;6;7;8)\vee\mathrm{eq}(1;2;3;45;6;7;8) \text{ by (7.259) and (7.260)}, \tag{7.268}$$
$$\mathrm{eq}(1;24;3;5;6;7;8)=\mathrm{eq}(137;246;58)\wedge\mathrm{eq}(1245;3;6;7;8) \text{ by (7.251) and (7.268)}, \tag{7.269}$$
$$\mathrm{eq}(1;25;3;4;6;7;8)=\mathrm{eq}(146;257;38)\wedge\mathrm{eq}(1;245;37;6;8) \text{ by (7.252) and (7.267)}, \tag{7.270}$$
$$\mathrm{eq}(16;2;3;4;5;7;8)=\mathrm{eq}(136;2;4578)\wedge\mathrm{eq}(146;25;3;7;8) \text{ by (7.253) and (7.264)}, \tag{7.271}$$
$$\mathrm{eq}(1;2;3;4;58;6;7)=\mathrm{eq}(13;2;4;58;6;7)\wedge\mathrm{eq}(124568;37) \text{ by (7.256) and (7.263)}, \tag{7.272}$$
$$\mathrm{eq}(135678;24)=\mathrm{eq}(13;2;4;58;6;7)\vee\mathrm{eq}(16;24;357;8) \text{ by (7.256) and (7.266)}, \tag{7.273}$$
$$\mathrm{eq}(123458;6;7)=\mathrm{eq}(13;2;4;58;6;7)\vee\mathrm{eq}(1245;3;6;7;8) \text{ by (7.256) and (7.268)}, \tag{7.274}$$
$$\mathrm{eq}(1;2;37;4;5;6;8)=\mathrm{eq}(1;24;37;5;6;8)\wedge\mathrm{eq}(1345678;2) \text{ by (7.257) and (7.262)}, \tag{7.275}$$
$$\mathrm{eq}(137;2;4;58;6)=\mathrm{eq}(1345678;2)\wedge\mathrm{eq}(137;24;58;6) \text{ by (7.262) and (7.265)}, \tag{7.276}$$
$$\mathrm{eq}(17;2;3;4;5;6;8)=\mathrm{eq}(124567;38)\wedge\mathrm{eq}(137;2;4;58;6) \text{ by (7.261) and (7.276)}, \tag{7.277}$$
$$\mathrm{eq}(1358;24;6;7)=\mathrm{eq}(135678;24)\wedge\mathrm{eq}(123458;6;7) \text{ by (7.273) and (7.274)}, \tag{7.278}$$
$$\mathrm{eq}(1;2;38;4;5;6;7)=\mathrm{eq}(146;257;38)\wedge\mathrm{eq}(1358;24;6;7) \text{ by (7.252) and (7.278)}. \tag{7.279}$$

Since $G(S)$ contains the edges $(1,6)$ by (7.271), $(6,4)$ by (7.255), $(4,2)$ by (7.269), $(2,5)$ by (7.270), $(5,8)$ by (7.272), $(8,3)$ by (7.279), $(3,7)$ by (7.275), $(7,1)$ by (7.277), Lemma 1 is applicable and completes the proof of Lemma 15. □

8. Appendix 2: the source code of the main program

As indicated in Section 4, here we present the Dev-Pascal 1.9.2 source code of the main computer program. Note that there are two ways to include comments in the program. First, after `//` (two forward slashes), the rest of a line is a comment. Second, any text between `{` and `}` (two curly brackets) is a comment; in this case, the comment can expand to several lines but it cannot contain curly brackets.

```
program equp2024ot;   uses sysutils, crt;
 const  created='August 20, 2024';  createdate =
 'Program equ2024ot version '+created+',          (C) Gabor Czedli, 2024.';
//*********************************************************************
// Some comment lines can contain misprints or can be ungrammatical; sorry.
const bellnos: array[1..10] of integer =
   (      1,        2,          5,       15,        52,
        203,      877,       4140,    21147,    115975 ); {Bell numbers}
 nmax=9; bnnmx= 21147 {=Bell(nmax)};  tnmx=2*nmax+1;
 // nmax=8; bnnmx= 4140 {=Bell(nmax)};  tnmx=2*nmax+1;
  freqdetail=60;{After how many dots to give details}
  freqdotarray:array[4..nmax] of integer=
   (5000, 2000, 1200, 600, 300,200);{After how many steps to display a dot}
  layermax= 7770{max_x(stirling2(nmax,x))};
  nnul=1; nmo=0; {"new 0", "new -1", increased}

type partt=array[1..tnmx] of byte; { Each entry is increased by 1 !
```

```
    That is, the program computes  with and stores 2,3,...,ne+1  (bytes)
    but inputs 1,2,...,ne. E.g., for ne=6, if
     1  0  2  3  6  0  4  5  0 -1  is the input, then the vector
     2  1  2  4  7  1  5  6  1  0  is stored. So, in computations,
     nnul=1 separates the blocks and nmo=0 terminates the partition!! }
  PSett= {"set of partitions" type; with reference to the variable A0,
     see later, its members are given in two ways: a "subset" of A0 given
     by bits, and by listing the A0-indices of the members of a PSett }
    record es: array[1..bnnmx] of Boolean;{which partitions of A0}
      ssize: integer; {how many partitions are in PSett}
      wh: array[1..bnnmx] of integer;//PSett={A0[wh[1]],...,A0[wh[ssize]]}
    end;  {PSett}
  layert=record prtpnt: array[1..layermax] of integer; {pointers to A0}
            lynum,diffpat: integer;  {Layer=horizontal subset (element
             of a common height).   The layer consists of partitions
             A0[prtpnt[1]],...,A0[prtpnt[lynum]]. If ordered, then the
             first diffpat of these are different patterns, and they
             represent all patterns. }
            ordered: boolean;
         end; {layert}
  layersett=array[0..nmax-1] of layert;
  blockstructt=array[1..nmax] of integer;
   {its i-th entry is the number of i-element blocks}

var   ne: integer {size of the base set};  bn: integer {:=bell(ne)};
  h: integer; {will stand for a given height}
  tne1: integer; {=2*ne+1} nep1: integer {:=ne+1};
  A0:array[1..bnnmx] of partt; {Set of all partition of [ne];
    each of its bn=Bell(ne) members in given in string form}
  MJt:array[1..bnnmx,1..bnnmx] of integer; {operation table:
    for i<j, M[i,j]=i meet j,  for i>j, M[i,j]=i join j}
  gps {general progress counter}, freqdot , dotsnumbr: longint;
  f: text;   rtop: longint; useots:Boolean; {do we use operation tables?}
  fulllayers:layersett; {All layers in it will be full} X: PSett;
    hour,minute,second,millisecond,hour0,minute0,second0,millisecond0:word;
//*********************************************************************
 procedure fopen; forward;  {Opens one of 4.txt, 5.txt, ..., 8.txt}
 procedure inputdata; forward; {Inputs Eq(ne), initializes A0, fulllayers}
 procedure dmistake(s: string); forward; {Halts with error message}
 procedure readfline(var p: partt); forward; {Reads a partition from f}
 procedure sToPart(var s:string; var p: partt);forward; {converts s into p}
 procedure makefulllayers; forward; {Computes the layers of Eq(ne)}
 function heightof(var x:partt):integer;forward;{:=height(x)}
 function whattodo: char;forward;{Prompts for choosing action}
 procedure readX(var X: PSett); forward; {reads into X \subseteq Eq(ne)}
 procedure generate(var X:PSett); forward;
   {X:=the sublattice X generates, without operation table}
     {old name: joinmeetclose}
```

```
 procedure join(var x,y,z: partt); forward; {z:=x+y} {nnul,nmo are used}
 procedure meet(var x,y,z: partt); forward; {z:=x*y} {nnul,nmo are used}
 function placeInSet(var y:partt):integer; forward; {y’s place in A0}
 function compare(var x,y: partt):integer; forward;
   {If x<y, then :=1; if x>y, then :=2; if x=y, then :=0}
 procedure putInSet(var y:partt; var X: PSett); forward;{inserts y into X}
 function  isinset(var y:partt; var X: PSett):boolean;forward;{y in X ?}
 procedure orderfullilayer(i:integer); forward; {Turns fulllayers[i]
   ordered so that it starts with different patterns, if it was unordered}
 procedure itspattern(var p: partt; var u: blockstructt); forward;
  {u:= the pattern of p; p ~ 2 1  2  4  7  1  5  6  1  0}
 function samepattern(var u, v: blockstructt): boolean; forward;
 procedure handledots(c:char); forward;
  {increments gps and writes s to screen at every freqdot step}
 procedure timing(start:boolean); forward; {displays system time}
 procedure filloptable; forward; {Sets MJt}
 procedure ngenerate(var X:PSett);forward;
   {X:=the sublattice X generates, "n" from "new"}
 procedure setuseots; forward;
  {sets useots, to control the use of operation tables}
 procedure SetWriteScreen(var X: PSett); forward; {writes X to the screen}
 procedure PWriteScreen(var x:partt); forward; {writes x to the screen}
 procedure PSetCopy(var X,Y: PSett); forward; {Y:=X}
//************************************************************************
procedure PSetCopy(var X,Y: PSett); {Y:=X}
 var i,sz: integer;
 begin sz:= X.ssize; Y.ssize:=sz; for i:=1 to bn do Y.es[i]:=X.es[i];
   for i:=1 to sz do Y.wh[i]:=X.wh[i];
 end;

procedure PWriteScreen(var x:partt); {forward} {writes x to the screen}
  var i: integer;
 begin //writeln(’number of elements = ’,ne);
   for i:=1 to tne1 do
     begin write(x[i]-1,’ ’);
        if i mod 20 =0 then writeln;
     end;  writeln;
 end;

procedure SetWriteScreen(var X: PSett); {writes X to the screen}
 var i: integer;
 begin with X do for i:=1 to ssize do PWriteScreen( A0[wh[i]] )
 end;

procedure setuseots; {forward;}
  {sets useots, to control the use of operation tables}
 var s: string; done: integer;
 begin done:=0;
```

```
  repeat readln(s);
   if (pos('y',s)>0) or (pos('Y',s)>0) then done:=1;
   if done=0 then if (pos('n',s)>0) or (pos('N',s)>0) then done:=2;
  until done>0;
  if done=1 then useots:=true else useots:=false;
 end;

procedure filloptable; {forward;} {Sets JMt}
   var i,j,bnne:integer; z:partt; cnt,rcnt,sbnne,pdt:longint;
   const stpp=1000000; inarow=50;
 begin bnne:=bellnos[ne]; sbnne:=bnne*(bnne-1) div 2; cnt:=0; rcnt:=0;
       pdt:=round(sbnne/stpp); if pdt=0 then pdt:=1;
  writeln('Computing the ',sbnne,'-element operation table. Each *,');
  writeln('if any, indicates ', 2*stpp,
     ' new entries, i.e, the fulfillment of ~ 1/',pdt,
     ' part of this task;');   writeln('   note that ',inarow,
    ' *s make a row and <ctrl-break> quits from the program.');
  for i:=1 to bnne do MJt[i,i]:=i;
  for i:=1 to bnne-1 do for j:=i+1 to bnne do {now i<j}
  begin join(A0[i],A0[j],z); MJt[j,i]:=placeInSet(z);
        meet(A0[i],A0[j],z); MJt[i,j]:=placeInSet(z); inc(cnt);
        if cnt mod stpp =0 then
        begin write('*'); inc(rcnt); if rcnt mod inarow=0 then writeln;
        end;
  end; writeln; writeln('The operation table is filled up.');
 end;

procedure timing(start:boolean); {forward;} {displays system time}
 begin if start then
  begin
   decodetime(time,hour0,minute0,second0,millisecond0);
   writeln(
   '   The computation below starts at  (hour:min:second.millisec)  '
                  ,hour0,':',minute0,':',second0,'.',millisecond0,' .')
  end   else
  begin  decodetime(time,hour,minute,second,millisecond);
   writeln(
   '   The computation above started at (hour:min:second.millisec)  '
                  ,hour0,':',minute0,':',second0,'.',millisecond0);
   writeln(
   '               and terminated at (hour:min:seconc.millisecond)  '
                  ,hour,':',minute,':',second,'.',millisecond,' .')
  end;
 end;

procedure handledots(c:char);
 {forward;} {increments gps and writes s to screen at every freqdot step}
 begin inc(gps);
```

```
   if gps mod freqdot = 0
   then
   begin write(c);  inc(dotsnumbr);
     if dotsnumbr mod freqdetail = 0 then
       begin write('(',gps,'-th ');
         if rtop>0 then write(' out of ',rtop,') ') else write(') ');
       end;
   end;
 end;

function samepattern(var u, v: blockstructt): boolean;{forward;}
  var i: integer;
 begin for i:=1 to ne do if u[i]<>v[i] then
   begin samepattern:=false; exit
   end;
   samepattern:=true;
 end;

procedure itspattern(var p: partt; var u: blockstructt); {forward;}
  {u:= the pattern of p; p ~ 2 1  2  4  7  1  5  6  1  0}
  var i,ic: integer;
 begin for i:=1 to ne do u[i]:=0 {at the start, 0 i-element blocks};
   i:=1; ic:=0 {the size of the actual block};
   while p[i]<>0 do
   begin{while p[i]<>0} if p[i]<>1 then inc(ic)
     else begin{now p[i]=1} u[ic]:=u[ic]+1; {one more ic-element block}
           ic:=0
          end;             inc(i);
   end {while u[i]<>0}
 end;

procedure orderfullilayer(i:integer); {forward;} {Turns fulllayers[i]
   ordered so that it starts with different patterns, if it was unordered}
  var j1,j2,chng: integer; isnew: boolean;  patj1,patj2: blockstructt;
 begin with fulllayers[i] do if not ordered then
  begin  ordered:=true; diffpat:=1;
   for j1:=2 to lynum do
   begin isnew:=true; {is A0[prtpnt[j1]] of a new pattern?}
    for j2:=1 to diffpat do  if isnew then
    begin
      itspattern(A0[prtpnt[j1]],patj1); itspattern(A0[prtpnt[j2]],patj2);
      if samepattern (patj1,patj2) then isnew:=false;
    end {for j2};
    if isnew then
    begin inc(diffpat);  chng:=prtpnt[diffpat]; {for changing}
          prtpnt[diffpat]:=prtpnt[j1]; prtpnt[j1]:=chng;
    end; {if isnew}
   end {for j1};
```

```
  end;
 end;

procedure testC(h:integer); {forward;} {We can enter a height}
  var i, i1,i2,i3,i4, cnt,nX: integer; longlnm:longint; X,oldX:PSett;
   function countcases:longint; {counts the cases up to automorphism}
              var j4,sm:integer;
      begin orderfullilayer(h);sm:=0; with fulllayers[h] do
       begin  for j4:=1 to diffpat do
         sm:=sm+(lynum-j4)*(lynum-j4-1)*(lynum-j4-2) div 6;
       end; {with fulllayers[h]}
       countcases:= sm;
      end;{function countcases}
 begin {testC} nX:=4; cnt:=0; gps:=0; dotsnumbr:=0;
  longlnm:=fulllayers[h].lynum;  writeln;
  if longlnm<nX  then dmistake('Too small layer, halting');
  orderfullilayer(h); rtop:=countcases;
  writeLn('Up to automorphism, there are at most ',rtop,
             ' 4-element sets of height=',h);
  writeLn('   (The program will write a new dot on the screen at every ',
              freqdot,'-th set, if any.)'); writeln;
  with fulllayers[h] do
  begin {with fulllayers[h]}
   for i1:=1 to diffpat do for i2:=i1+1 to lynum-2 do
   for i3:=i2+1 to lynum-1 do for i4:=i3+1 to lynum do
   begin with X do begin ssize:=0; for i:=1 to  bn do es[i]:=false;
                   end;
    putInSet(A0[prtpnt[i1]], X); putInSet(A0[prtpnt[i2]], X);
    putInSet(A0[prtpnt[i3]], X); putInSet(A0[prtpnt[i4]], X);
    PSetCopy(X,oldX);
    if useots then ngenerate(X) else generate(X);
    if X.ssize=bn then
     begin inc(cnt);   writeln;   writeln;
      write('YES, Eq(',ne,
            ') has a 4-element horizontal generating set of');
      writeln(' height ',h,'. '); write('     (Such a generating');
      writeln(' set was found at the ',gps,'th trial.)');
       writeln('               The generating set found is this:');
      SetWriteScreen(oldX);
      writeln('                 Hit <enter> to abandon.');
      timing(false);  exit;
     end;
    handledots('.');
   end;{four-fold for} writeln;  writeln; writeln('NO, Eq(',ne,
   ') has no 4-element horizontal generating set of height ',h,'.');
   writeln('      (',rtop,'   4-element subsets have been checked.)');
   timing(false);
  end; {with fulllayers[h]}
```

```
 end; {testC}

function  isinset(var y:partt; var X: PSett):boolean;{forward;}{y in X ?}
 begin isinset:=X.es[placeInSet(y)]
 end;

procedure putInSet(var y:partt; var X: PSett);{forward;}{inserts y into X}
  var i: integer;
  begin i:=placeInSet(y); if i=0 then dmistake('Internal Error/putInSet');
   with X do if not isinset(y,X) then
             begin inc(ssize); es[i]:=true; wh[ssize]:=i;
             end;
  end;

function compare(var x,y: partt):integer; {forward;}
  {If x<y, then :=1; if x>y, then :=2; if x=y, then :=0}
   var which, i: integer;
 begin which:=0; i:=0;
  repeat inc(i);
    if x[i]<y[i] then which:=1;  if x[i]>y[i] then which:=2;
  until (which<>0) or (i>=tne1);
  compare:=which;
 end; {arePartsEqual}

function placeInSet(var y:partt):integer; {forward;} {y's place in A0}
  var left,right,middle, place, ii,flip,cnt,tillbn: integer;
 begin   left:=1; right:=bn; place:=0; flip:=0;cnt:=0; tillbn:=bn+5;
   while (place=0) and (cnt<tillbn) do
   begin
    middle:= flip + ((left+right) div 2);
      flip:=1-flip; ii:=compare(y, A0[middle]);
      if ii=0 then begin place:=middle; right:=left
                     {to get out from the while loop}
                  end;
      if ii=1 then right:=middle;
      if ii=2 then left:=middle;      inc(cnt)
   end; {while}
   placeInSet:=place
 end;

function arecollapsed(ie,je:byte; var x: partt):boolean;
   {is ie=je modulo x ? Here ie and je are the enlarged bytes }
  var u,v: integer; b: byte; are:boolean;
 begin if ie>je then begin b:=ie; ie:=je; je:=b end; are:=false;
  u:=0;    repeat inc(u) until x[u]=ie;  v:=u-1;
  repeat inc(v); if x[v]=je then are:=true;
  until are or (x[v]=nnul);          arecollapsed:=are;
 end;
```

```
procedure segmentsort(u,v:integer; var z: partt); {sorts z[u]--zg[v] }
   var i,j: integer; swapped:Boolean; b: byte;
 begin if not ( (1<= u) and (u<v) and (v<=tnmx) ) then dmistake('sort');
   i:=v;
   repeat swapped:=false;
     for j:=u to i-1 do if z[j]>z[j+1] then
         begin {if} b:=z[j]; z[j]:=z[j+1]; z[j+1]:=b; swapped:=true;
         end {if} ;
     i:=i-1;
   until (not swapped) or (i<=u)
 end; {segmentsort}

procedure join(var x,y,z: partt);{forward;} {z:=x+y} {nnul,nmo are used}
  var todo:array[2..nmax+1] of boolean;  dne,i,j,iz,ibs,jz,fi,ti: integer;
 begin
  for i:=1 to tnmx do z[i]:=nmo;
  dne:=0; for i:=2 to nep1 do todo[i]:=true;
  iz:=0; {last place where we put an element into  z}
  for i:=2 to nep1 do if todo[i] then
  begin
    inc(iz); z[iz]:=i;  ibs:=iz {block starts};
    fi:=iz+1; inc(dne);  todo[i]:=false;
    for j:=i+1 to nep1 do
    if todo[j] and (arecollapsed(i,j,x) or arecollapsed(i,j,y)) then
     begin  todo[j]:=false; inc(dne); inc(iz); z[iz]:=j;
     end; {for j; the x\cup y -related elements are put in the block of i}
    if iz>ibs  then {i is not in a singleton block:}
     begin ti:=iz; {now z[fi--ti] is an ordered segment of the block of
        i, and we need to find its neighbors as long as we find new}
      repeat
        {Looking for the neighbors of the ORDERED segmeng z[fi]...z[ti].}
        for jz:=fi to ti do {jz is used to walk in segment z[fi]...z[ti]}
        for j:=i+1 to nep1 do
         begin   if todo[j] and (arecollapsed(z[jz],j,x)
                                or arecollapsed(z[jz],j,y)) then
          begin inc(iz); z[iz]:=j; todo[j]:=false;  inc(dne);
          end;
         end;
        fi:=ti+1; ti:=iz ;
      until (ti<fi) or (dne>=ne);  {No more neighbor, segment, element}
     end; {if iz>ibs}
     if ibs+1< iz then begin segmentsort(ibs+1,iz,z);
                       end;
     inc(iz); z[iz]:=nnul; {end of the block of i}
  end; {for i} {all blocks are ready}
  inc(iz); z[iz]:=nmo;
  for j:=iz+1 to tnmx do z[j]:=nmo;
```

```
 end; {join}

procedure meet(var x,y,z: partt);{forward;} {z:=x*y} {nnul,nmo are used}
   var todo:array[2..nmax+1] of boolean;  i,j,iz, i2: integer;
 begin for i:=2 to nep1 do todo[i]:=true;
  iz:=0; {next place in z}
  for i:=2 to nep1 do if todo[i] then
  begin inc(iz); z[iz]:=i; todo[i]:=false;
   for j:=i+1 to nep1 do
     if todo[j] and arecollapsed(i,j,x) and arecollapsed(i,j,y) then
     begin todo[j]:=false;  inc(iz); z[iz]:=j {enlarged by 1 !}
     end; {for j}
   inc(iz); z[iz]:=nnul;
  end; {for i}   inc(iz); z[iz]:=nmo;
  for i2:=iz+1 to tnmx do z[i2]:=nmo;
 end; {meet}

procedure ngenerate(var X:PSett);{forward;}
  {X:=the sublattice X generates, "n" from "new"}
 var oldsize, i,j,k,doneUpTo: integer;
 begin doneUpTo:=0;   {The purpose of doneUpTo: we not to check those
  that were checked in the previous round.}
  with X do
  begin
    repeat oldsize:=ssize;
      for i:=1 to oldsize do
      begin{for i}
        for j:= doneUpTo+1 to oldsize do
        begin{for j}  k:=MJt[wh[i],wh[j]]; {join or meet}
         if not es[k] then
          begin es[k]:=true; inc(ssize); wh[ssize]:=k;
          end;  k:=MJt[wh[j],wh[i]];
             {meet or join, the opposite of the above}
         if not es[k] then
          begin es[k]:=true; inc(ssize); wh[ssize]:=k;
          end;
        end;{for j}
      end;{for i}
      doneUpTo:=oldsize;
    until ssize=oldsize;
  end; {with X}
 end; {ngenerate}

procedure generate(var X:PSett);{forward;}
     {X:=the sublattice X generates, without operation table}
  var oldsize, i,j,k,doneUpTo: integer; z: partt;
 begin doneUpTo:=0;   {The purpose of doneUpTo: we not to check those
  that were checked in the previous round.}
```

```
  with X do
  begin
    repeat oldsize:=ssize;
      for i:=1 to oldsize do
      begin{for i}
        for j:= doneUpTo+1 to oldsize do
        begin{for j}  meet(A0[wh[i]],A0[wh[j]],z); k:=placeInSet(z);
         if not es[k] then
          begin es[k]:=true; inc(ssize); wh[ssize]:=k;
          end;        join(A0[wh[i]],A0[wh[j]],z); k:=placeInSet(z);
         if not es[k] then
          begin es[k]:=true; inc(ssize); wh[ssize]:=k;
          end;
        end;{for j}
      end;{for i}
      doneUpTo:=oldsize;
    until ssize=oldsize;
  end; {with X}
 end; {generate}

procedure readX(var X: PSett);{forward;} {reads into X \subseteq Eq(ne)}
      var  s: string; p:partt;  i: integer;
 begin with X do
  begin {with X} ssize:=0; for i:=1 to  bn do es[i]:=false;{: X is empty}
   while ssize<4 do
   begin{with X while X.ssize<4}
    writeln('Enter the ',ssize+1,
     '-st/nd/rd/th partition; syntax: the same as in ',ne,'.txt .');
    readLn(s); sToPart(s,p);
    if placeInSet(p)>0 then putInSet(p,X)
        else writeln('   Invalid partition. Mind the syntax.');
   end{while X.ssize<4}
  end {with X};
  writeln('Computing [X] has been started, please wait ...');
 end;

function whattodo: char;{forward;}{Prompts for choosing action}
   var c: char; s: string;
 begin c:=' ';
  repeat  writeLn(
   'Type "a" or "b" (followed by <enter>) to choose from:'); writeln(
   '(a) does a given 4-element set X of partitions generate Eq(n) or');
   writeln('(b) is there a 4-element horizontal generating set',
    ' of a given height?');
   readln(s); if (pos('a',s)>0) or (pos('A',s)>0) then c:='a';
   if (pos('b',s)>0) or (pos('B',s)>0) then c:='b';
  until (c='a') or (c='b');
  whattodo:=c;
```

```
 end; {whattodo}

procedure dmistake(s: string);  {forward;} {Halts with error message}
 begin writeLn('Error '+s); writeLn('Hit <enter> to quit'); readLn; halt;
 end;

function heightof(var x:partt):integer;{forward;} {:=height(x)}
   var i,j,h: integer;
 begin  h:=0; i:=1; j:=0;
  while x[i]<>0 do
   begin if x[i]<>1 then inc(j) else begin h:=h+(j-1); j:=0
                                       end;            inc(i);
   end;
   heightof:=h;
 end;

procedure makefulllayers; {forward;} {Computes the layers of Eq(ne)}
     var i,j: integer;
 begin {first, we empty fulllayers:}  for i:=0 to ne-1 do
    begin
      fulllayers[i].lynum:=0; fulllayers[i].ordered:=false;
    end;
    for i:=1 to bn do
    begin  j:=heightof(A0[i]);
      with fulllayers[j] do
      begin lynum:=lynum+1; prtpnt[lynum]:=i;
          if lynum>layermax then dmistake('Internal error 1');
      end;
    end;
 end;

procedure sToPart(var s:string; var p: partt);{forward, converts s into p}
      var i,j,b2: integer; s1: string;  code: word; b: byte;
 begin
  while (length(s)>0) and (s[length(s)]=' ') do delete(s,length(s),1);
  s:=s+' '; i:=0;
  while length(s)>0 do
    begin i:=i+1; while (length(s)>0) and (s[1]=' ') do delete(s,1,1);
     s1:=copy(s,1,pos(' ',s)-1); val(s1,b2,code);
     if code<>0 then  dmistake('Invalid character');
     b:=b2+1; p[i]:=b;{bytes are increased by 1!}
     delete(s,1,pos(' ',s));
    end;
  for j:=i+1 to tnmx do p[j]:=nmo;
 end; {sToPart}

procedure readfline(var p: partt);{forward;}{Reads a partition from f to p}
      var s: string;
```

```
 begin readLn(f,s); sToPart(s,p)
 end; {readline}

procedure fOpen; {forward;}  {Opens one of 4.txt, 5.txt, ..., 8.txt}
   var i: integer;
 begin writeln;  writeln(createdate);
  writeln('  Topic/purpose:  4-element generating sets of Eq(n).');
  write('What is n (the size of the base set) ? ');
  readln(ne);     if (ne<3) or (ne>nmax)
  then begin  writeln('     Error! Only 3<n<',nmax+1,
        ' is allowed. The program halts after <enter>');readln; halt
      end;
  assign(f,IntToStr(ne)+'.txt'); {$I-} reset(f); {$I+}
  if ioresult >0 then  {file opening was unsuccessful}
   begin       writeln; writeln('            ERROR!'); writeln;
     writeln('   IMPORTANT: 1.txt, ..., 9.txt should be perfect and');
     writeln('   they should be in the current folder!  If not so,');
     writeln('   ten run partitions.exe in the current folder ');
     writeln('   to create these auxiliary files.'); writeln;
     writeln('Now the program will halt after <enter>'); readln;
   end;
 end; {procedure fOpen}

procedure inputdata; {forward;} {Inputs Eq(ne), initializes A0, fulllayers}
   var i: integer; s2: string;           // ii:integer;
 begin  bn:=0;{counter; at the end: Bell(ne)}
  for i:=1 to 2 do
   begin if eof(f) then dmistake('Bad first two lines in the input file');
         readln(f,s2);
         if eof(f) then dmistake('Bad first two lines in the input file');
   end;
  while not eof(f) do
    begin inc(bn);
           readfline(A0[bn]);
           if eof(f) then dmistake('The file should not end here');
           readLn(f,s2);
    end;                                              writeln;
  if bn<1 then dmistake('No partition is given in the input file');
  ne:=0; for i:=1 to tnmx do if A0[1,i]>ne then ne:=A0[1,i];  nep1:=ne;
  ne:=ne-1; {since bytes in A0 are enlarged} tne1:=2*ne+1; close(f);
  freqdot:=freqdotarray[ne];
 end{inputdata};

begin {main} fOpen;  inputdata;
 if whattodo='a' then
 begin
   readX(X);
   generate(X);
```

```
   write('[X] consists of ',X.ssize,' elements, so X ');
   if X.ssize<bn then
     writeln('does NOT generate Eq(',ne,')')  else
     writeln('GENERATES Eq(',ne,')')
 end
 else
 begin  makefulllayers;
   writeln('The program is going to decide whether there is a horizontal');
   writeln('generating set of a given height h, 1 <= h <= ',ne-2,'.');
   if ne <= 8 then
    begin
     writeln('Should we create operation tables (takes time but ');
     write(' accelerates the computation later)? Enter y or n: ');
     setuseots;
    end  else   useots:=false;
   if useots then
    begin  writeln(' Filling up the operation tables ... ');
      timing(true);  filloptable; writeln;  timing(false);
    end;
   repeat write(
     'Enter the height (of the 4-element generating set to be found): ');
      readln(h);
   until (0<h) and (h<ne-1);
   timing(true); testC(h);
 end;
 write('   Hit <enter> to abandon.'); readln;
end. {main}
```

9. Appendix 3: the source code of the auxiliary program

Before running the main program, the following auxiliary program should create the necessary files.

```
program partitions;   uses sysutils, crt;
 const created='August 17, 2024';   createdate =
 'Program partitions ver. '+created+',          (C) Gabor Czedli, 2024.';
const n=9; BellNumber=30000; {>=Bell1+...+Bell(n)}  n2p1=2*n+1;

type partt=array[1..n2p1] of integer; {partition-type}

var i,k,po,pn,nb: integer;
 A:array[1..BellNumber, 1..n2p1] of integer;
 {Each row of A is a partition on some k, 2 <=k <= n in the form, say,
  1,3,0,2,4,0,-1,-1,-1,...; 0 separates the blocks, -1 is the end symbol.
  The elements of a block are in increasing order. The blocks are ordered
  lexicographically. In the example above, k=4.}
 pti,ptt: array[1..n] of integer; {pointer_initial and pointer_terminal;
  the partitions of [k] are the pti[k], pti[k]+1,...,ptt[k] -th rows of A.}
  hour,minute,second,millisecond,hour0,minute0,second0,millisecond0:word;

procedure initA; forward; {For k=1, puts Eq(k) into A, pti, ptt.}
```

```
function numbL(k:integer):integer; forward;
 {The number of blocks of partition A[k]}
procedure copypartition(op,np:integer);forward; {Copies A[op,-] into
  A[np,-]}
procedure insOBL(i1,j1,k: integer); forward;
 {inserts i1 to the j1-st block of A[k,-] if this block exists}
procedure insNBL(i1,pn:integer); forward;
 {inserts i1 to a new block in A[pn,-]; i1 > earlier elements}
procedure pts2file(k:integer); forward;
 {Saves the partitions forming Eq(k) into k.txt}
procedure swap(i,j:integer);forward; {swaps partitions A[i,-] and A[j,-]}
function whichless(i,j:integer):integer; forward;  {Gives 1,2,0
  if A[i,-] < A[j,-], 2 if >, and 0 if =, respectively.}
function lessthanp(i:integer; var p: partt): integer; forward; {gives 1,
 0, 2 if A[i,-] < p (first, 0: A[i,-] = p (none) 2: p < A[i,-] (second)}
procedure qsort(sta,top:integer); forward;
      {sorts A from A[sta,-] to A[top,-]}
 procedure timing(start:boolean); forward; {displays system time}

procedure timing(start:boolean); {forward;} {displays system time}
 begin if start then
  begin
   decodetime(time,hour0,minute0,second0,millisecond0);
   writeln(
   '   The computation below starts at  (hour:min:second.millisec)  '
                 ,hour0,':',minute0,':',second0,'.',millisecond0,' .')
  end   else
  begin  decodetime(time,hour,minute,second,millisecond);
   writeln(
   '   The computation above started at (hour:min:second.millisec)  '
                 ,hour0,':',minute0,':',second0,'.',millisecond0);
   writeln(
   '              and terminated at (hour:min:seconc.millisecond)  '
                 ,hour,':',minute,':',second,'.',millisecond,' .')
  end;
 end;

procedure qsort(sta,top:integer); {forward;}
      {sorts A from A[sta,-] to A[top,-]}
   var  i,j,i5,spli: integer; p: partt;
 begin if (sta+1=top)  and (whichless(sta,top)=2) then swap(sta,top);
  if sta+1 < top then
  begin
    for i5:=1 to n2p1 do p[i5]:= A[sta,i5]; {p=pivot}
    i:=sta+1; j:=top;
    while i <= j do
     begin while (i<=top) and (lessthanp(i,p)=1) do inc(i);
       while (j>sta) and ((lessthanp(j,p)=0) or (lessthanp(j,p)=2))
```

```
              do j:=j-1;
       if  i<j then swap(i,j);
     end{while i<=j};
    swap(sta,j); spli:=j;
    qsort(sta,spli-1);
    qsort(spli+1,top)
  end;
 end;

function lessthanp(i:integer; var p: partt): integer; {forward;} {gives 1,
 0, 2 if A[i,-] < p (first, 0: A[i,-] = p (none) 2: p < A[i,-] (second)}
  var u,w: integer;
 begin w:=0; u:=1; while (w=0) and (u<n2p1) do
  begin u:=u+1; if A[i,u]<p[u] then w:=1; if A[i,u]>p[u] then w:=2;
  end{while};
  lessthanp:=w;
 end;

function whichless(i,j:integer):integer; {forward;}{Gives 1,2,0
  if A[i,-] < A[j,-], 2 if >, and 0 if =, respectively.}
   var u,w: integer;
  begin  w:=0; u:=1; while (w=0) and (u<n2p1) do
    begin {if equal so far then they terminate simultaneously!}
      u:=u+1; if A[i,u]<A[j,u] then w:=1; if A[i,u]>A[j,u] then w:=2;
    end;      whichless:=w;
  end; {whichless}

procedure swap(i,j:integer);{forward;} {swaps partitions A[i,-] and A[j,-]}
  var u,v: integer;
 begin for u:=1 to n2p1 do begin v:=A[i,u]; A[i,u]:=A[j,u]; A[j,u]:=v
                           end;
 end;

procedure pts2file(k:integer); {forward;}
 {Saves the partitions forming Eq(k) into k.txt}
 var f: text; s:string; i,j,m: integer;
 begin s:=IntToStr(k)+'.txt'; assign(f,s); {$I-} reset(f); {$I+}
  if ioresult=0 then {already exist!}
   begin write('The file ',s,' already exists! Remove the files 2.txt,');
    writeln('...,9.txt from'); writeln(' the current folder and restart',
    ' the program again. Now hit <enter> to quit.'); readln; halt;
   end;
   {$I-} rewrite(f); {$I+} if ioresult<>0 then
      begin writeln('Something is wrong. No disk space? Hit <enter>');
        readln; halt;
      end;
  writeln(f,'§ The list of partitions on {1,...,',k,'} begins here');
  m:=0;
```

```
  for j:= pti[k] to ptt[k] do
  begin inc(m); write(f,'§    The '); write(f,m);
   writeln(f,' -st/nd/rd/th partition is the following:');
   i:=1; write(f,' ');
   while A[j,i]<>-1 do begin write(f,' ',A[j,i],' '); inc(i);
                       end{while}; writeln(f,'-1');
  end{for j};
  writeln(f,'§ This was the last partition on {1,...,',k,'}');
  close(f);
 end;

procedure insNBL(i1,pn:integer); {forward;}
 {inserts i1 to a new block in A[pn,-]; i1 > earlier elements}
 var i,j:integer;
 begin  i:=1;  while A[pn,i] <> -1 do i:=i+1;
   A[pn,i]:=i1; A[pn,i+1]:=0;
   for j:=i+2 to n2p1 do  A[pn,j]:=-1;
 end;

procedure insOBL(i1,j1,k: integer); {forward;}
 {inserts i1 to the j1-st block of A[k,-] if this block exists.}
 var f1,f2,cnt,i: integer;
 begin  f1:=0; cnt:=1; i:=0;
  {Goal: A[k,f1] should be the 1st element of the j1-st block}
  while f1=0 do
   begin  i:=i+1; if cnt=j1 then f1:=i; if A[k,i]=0 then cnt:=cnt+1;
   end; {Now A[k,f1] is the first element of the j1-st block.}
   f2:=f1;     while A[k,f2]<> 0 do f2:=f2+1;
   {Now A[k,f2] is the 1st zero after A[k,f1]}
   for i:=1 to n2p1-f2 do A[k,n2p1-(i-1)]:=A[k,n2p1-i]; {shift to right}
   A[k,f2]:=i1; {The insertion, at last. Note that i1 must be bigger
    than the earlier elements.}
 end;

procedure copypartition(op,np:integer); {forward;} {Copies A[op,-] into
  A[np,-]}
  var i: integer;
 begin for i:=1 to n2p1 do A[np,i]:=A[op,i];
 end;

function numbL(k:integer):integer; {forward;}
 {The number of blocks of partition A[k]}
  var count,i: integer; stillSearchForMin1: boolean;
 begin count:=0; stillSearchForMin1:=true; i:=0;
  while stillSearchForMin1 do
  begin i:=i+1; if A[k,i]=0 then count:=count+1;
     if A[k,i]=-1 then stillSearchForMin1:=false;
  end;      numbL:=count;
```

```
 end;

procedure initA; {forward;} {For k=1, puts Eq(k) into A, pti, ptt.}
  var i: integer;
begin pti[1]:=1; ptt[1]:=1; A[1,1]:=1; A[1,2]:=0;
   for i:=3 to n2p1 do A[1,i]:=-1;
end;

begin {main} writeln(createdate);
 writeln('For k=2,...,',n,', the program lists the partitions of');
 writeln('    the set {1,2,...,k}, and prints them into ',k,'.txt');
 timing(true);initA;
 for k:=2 to n do
  begin {creating all partitions on [k]}
   pti[k]:=ptt[k-1]+1; ptt[k]:=ptt[k-1];
   for po:=pti[k-1] to ptt[k-1] do  {po: partition-old}
   begin {for po,
    constructing partitions from old A[po,-] on [k-1] }
    nb:=numbL(po); {number of old blocks}
    for i:=1 to nb do {adding k to the i-th block in new place}
     begin ptt[k]:=ptt[k]+1; pn:=ptt[k];
      copypartition(po,pn); insOBL(k,i,pn);
     end;{for i} {next, we add k to a new block}
    ptt[k]:=ptt[k]+1; pn:=ptt[k]; copypartition(po,pn);
    insNBL(k,pn); {The descendants of A[po,-] have been created}
   end;{for po; all partitions on [k] have been created}
  end; {for k}
 {for all k<=n, all partitions on [k] have been created}
 for k:=2 to n do
 begin
   qsort(pti[k],ptt[k]);  pts2file(k)
 end;             timing(false);
 writeln(
 'The required auxiliary files are ready. Hit <enter> to quit');
 readln;
end.{main}
```

References

[1] Czédli G. Lattices embeddable in three-generated lattices. *Acta Sci. Math. (Szeged),* 2016. Vol. 82. P 361–382. DOI: 10.14232/actasm-015-586-2

[2] Czédli G. Generating Boolean lattices by few elements and exchanging session keys. arXiv:2303.10790

[3] Czédli, G, Kurusa, Á: A convex combinatorial property of compact sets in the plane and its roots in lattice theory. *Categories and General Algebraic Structures with Applications,* 2019. Vol. 11. P 57–92. DOI: 10.29252/CGASA.11.1.57

[4] Czédli, G, Oluoch, L. Four-element generating sets of partition lattices and their direct products. *Acta Sci. Math. (Szeged),* 2020. Vol. 86. P 405–448. DOI: 10.14232/actasm-020-126-7

[5] Grätzer G. *General Lattice Theory.* 2nd. ed. Basel–Boston–Berlin: Birkhäuser, 1998. XX+663 p. ISBN 978-3-7643-6996-5.

[6] Grätzer G. *Lattice Theory: Foundation.* Basel: Birkhäuser, 2011. XXX 614 p. DOI: 10.1007/978-3-0348-0018-1

[7] P. Pudlák and J. Tůma: Every finite lattice can be embedded in a finite partition lattice; Algebra Universalis **10**, 74–95 (1980).

[8] Strietz, H. Finite partition lattices are four-generated. In: Proc. Lattice Th. Conf. Ulm, 1975, pp. 257–259.
[9] Strietz, H. Über Erzeugendenmengen endlicher Partitionenverbände. (German) Studia Sci. Math. Hungar. 12 (1977), 1–17 (1980)
[10] Whitman P. M. Lattices, equivalence relations, and subgroups. *Bull. Amer. Math. Soc.*, 1946. Vol. 2. P. 507–522.
[11] Zádori L. Generation of finite partition lattices. In: *Lectures in universal algebra: Proc. Colloq. Szeged, 1983. Colloq. Math. Soc. János Bolyai,* vol. 43. Amsterdam: North-Holland Publishing, 1986. P. 573–586.

Email address: czedli@math.u-szeged.hu
URL: http://www.math.u-szeged.hu/~czedli/

University of Szeged, Bolyai Institute. Szeged, Aradi vértanúk tere 1, HUNGARY 6720