Logic and Computation, & their interactions

Saeed Salehi

University of Tabriz & IPM

http://SaeedSalehi.ir/

Logic is . . .

- From the Greek word LOGOS, translated as "sentence", "discourse", "reason", "rule", and "ratio".
- The study of arguments (Wikipedia) "in the disciplines of philosophy, mathematics, and computer science".
- Logic is for constructing proofs which give us reliable confirmation of the truth of the proven proposition.

Logic is . . .

- From the Greek word LOGOS, translated as "sentence", "discourse", "reason", "rule", and "ratio".
- The study of arguments (Wikipedia) "in the disciplines of philosophy, mathematics, and computer science".
- Logic is for constructing proofs which give us reliable confirmation of the truth of the proven proposition.

Logic is . . .

- From the Greek word LOGOS, translated as "sentence", "discourse", "reason", "rule", and "ratio".
- The study of arguments (Wikipedia) "in the disciplines of philosophy, mathematics, and computer science".
- Logic is for constructing proofs which give us reliable confirmation of the truth of the proven proposition.

SAEED SALEHI, **Logic and Computation, & their interactions**, Arak 2019.

Logic is . . .

- From the Greek word LOGOS, translated as "sentence", "discourse", "reason", "rule", and "ratio".
- The study of arguments (Wikipedia) "in the disciplines of philosophy, mathematics, and computer science".
- Logic is for constructing proofs which give us reliable confirmation of the truth of the proven proposition.

SAEED SALEHI, **Logic and Computation, & their interactions**, Arak 2019.

Mathematical Logic is . . .

- Application of mathematical techniques to logic.
- Mathematical logic (also known as symbolic logic) is a subfield of mathematics with close connections to computer science and philosophical logic. (Wikipedia)
- The field includes both the mathematical study of logic and the applications of formal logic to other areas of mathematics. (Wikipedia)

Mathematical Logic is . . .

- Application of mathematical techniques to logic.

- Mathematical logic (also known as symbolic logic) is a subfield of mathematics with close connections to computer science and philosophical logic. (Wikipedia)

- The field includes both the mathematical study of logic and the applications of formal logic to other areas of mathematics. (Wikipedia)

Mathematical Logic is . . .

- Application of mathematical techniques to logic.
- Mathematical logic (also known as symbolic logic) is a subfield of mathematics with close connections to computer science and philosophical logic. (Wikipedia)
- The field includes both the mathematical study of logic and the applications of formal logic to other areas of mathematics. (Wikipedia)

Mathematical Logic is . . .

- Application of mathematical techniques to logic.
- Mathematical logic (also known as symbolic logic) is a subfield of mathematics with close connections to computer science and philosophical logic. (Wikipedia)
- The field includes both the mathematical study of logic and the applications of formal logic to other areas of mathematics. (Wikipedia)

Hilbert's Entscheidungsproblem = Decision Problem

Finding an ALGORITHM (or AL-KHWARIZMI):
Input: A (Mathematical) Statement.
Output: *YES* (if universally valid) *NO* (if not always valid).

Computably Decidable set A: an algorithm $\mathcal{P}$ decides on any input x whether $x \in A$ (outputs **YES**) or $x \notin A$ (outputs **NO**).

$$\xrightarrow{\texttt{input}:\ x\in\mathscr{M}} \boxed{\text{Algorithm}} \xrightarrow{\texttt{output}:} \begin{cases} \text{YES} & \text{if } x \in A \\ \text{NO} & \text{if } x \notin A \end{cases}$$

Algorithm: single–input, Boolean–output (**1**, **0**)

Hilbert's Entscheidungsproblem = Decision Problem

Finding an ALGORITHM (or AL-KHWARIZMI):

Input: A (Mathematical) Statement.

Output: *YES* (if universally valid) *NO* (if not always valid).

Computably Decidable set A: an algorithm $\mathcal{P}$ decides on any input x whether $x \in A$ (outputs **YES**) or $x \notin A$ (outputs **NO**).

input: $x \in \mathscr{M}$ → Algorithm → **output**: $\begin{cases} \texttt{YES} & \text{if } x \in A \\ \texttt{NO} & \text{if } x \notin A \end{cases}$

Algorithm: single–input, Boolean–output (**1**,**0**)

Hilbert's Entscheidungsproblem = Decision Problem

Finding an ALGORITHM (or AL-KHWARIZMI):

Input: A (Mathematical) Statement.

Output: *YES* (if universally valid) *NO* (if not always valid).

Computably Decidable set A: an algorithm $\mathcal{P}$ decides on any input x whether $x \in A$ (outputs **YES**) or $x \notin A$ (outputs **NO**).

input: $x \in \mathscr{M}$ → Algorithm → **output**: $\begin{cases} \text{YES} & \text{if } x \in A \\ \text{NO} & \text{if } x \notin A \end{cases}$

Algorithm: single–input, Boolean–output (**1**, **0**)

Hilbert's Entscheidungsproblem = Decision Problem

Finding an ALGORITHM (or AL-KHWARIZMI):

Input: A (Mathematical) Statement.

Output: *YES* (if universally valid) *NO* (if not always valid).

Computably Decidable set A: an algorithm $\mathcal{P}$ decides on any input x whether $x \in A$ (outputs `YES`) or $x \notin A$ (outputs `NO`).

`input`: $x \in \mathscr{M}$ ⟶ Algorithm ⟶ `output`: $\begin{cases} \text{YES} & \text{if } x \in A \\ \text{NO} & \text{if } x \notin A \end{cases}$

Algorithm: single–input, Boolean–output (`1`,`0`)

Hilbert's Entscheidungsproblem = Decision Problem

Finding an ALGORITHM (or AL-KHWARIZMI):
Input: A (Mathematical) Statement.
Output: *YES* (if universally valid) *NO* (if not always valid).

Computably Decidable set A: an algorithm $\mathcal{P}$ decides on any input x whether $x \in A$ (outputs **YES**) or $x \notin A$ (outputs **NO**).

Algorithm: single–input, Boolean–output (**1**,**0**)

Hilbert's Entscheidungsproblem = Decision Problem

Finding an ALGORITHM (or AL-KHWARIZMI):

Input: A (Mathematical) Statement.

Output: *YES* (if universally valid) *NO* (if not always valid).

Computably Decidable set A: an algorithm $\mathcal{P}$ decides on any input x whether $x \in A$ (outputs **YES**) or $x \notin A$ (outputs **NO**).

input: $x \in \mathscr{M}$ → Algorithm → **output**: $\begin{cases} \texttt{YES} & \text{if } x \in A \\ \texttt{NO} & \text{if } x \notin A \end{cases}$

Algorithm: single–input, Boolean–output (1, 0)

Hilbert's Entscheidungsproblem = Decision Problem

Finding an ALGORITHM (or AL-KHWARIZMI):
Input: A (Mathematical) Statement.
Output: *YES* (if universally valid) *NO* (if not always valid).

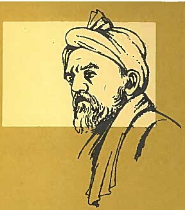
Computably Decidable set A: an algorithm $\mathcal{P}$ decides on any input x whether $x \in A$ (outputs **YES**) or $x \notin A$ (outputs **NO**).

$$\xrightarrow{\texttt{input}:\quad x\in\mathscr{M}} \boxed{\text{Algorithm}} \xrightarrow{\texttt{output}:} \begin{cases} \texttt{YES} & \text{if } x \in A \\ \texttt{NO} & \text{if } x \notin A \end{cases}$$

Algorithm: single–input, Boolean–output (**1**, **0**)

کتاب

جبر و مقابله



خوارزمی و میراث علمی وی

کتاب جبر و مقابله نخستین اثر علمی بر جای مانده است از محمد بن موسی خوارزمی، ریاضیدان بزرگ ایرانی، در علم «حساب و جبر و مقابله». کتابی که در آغاز قرن سوم هجری ـ حدود یکهزار و دویست سال پیش از این ـ به همت و ابتکار این ریاضیدان ایرانی تبار به زبان عربی تصنیف شد و به فرهنگ رو به گسترش اسلامی رونقی تازه بخشید. استقبال گرم و کم سابقه‌ای که در محافل علمی روزگار خوارزمی، و مآخذ قرنهای پس از وی، از این کتاب ریاضی به عمل آمد مهر تأییدی شد که بر نفاست کتاب و کفایت نویسنده‌اش نقش بست و زمینهٔ نیکنامی و جاودانگی نویسنده و نوشتهٔ او را در سراسر گیتی فراهم ساخت.

خوارزمی کار نگارش این اثر ماندنی خویش را در سال ۲۱۵ هجری به پایان رسانیده است. اثری که پس از انتشار در قلمرو جهان اسلام پیوسته استادان را مفید بوده و دانشجویان را کلید.

کتاب جبر خوارزمی در سال ۵۴۰ هجری (۱۱۴۵ میلادی) به همت «رابرت جستری» به لاتین ترجمه شد، و این ترجمه را می توان آغاز رواج علم جبر در اروپا دانست و از آنجا به سراسر گیتی.

کتاب

جبر و مقابله

نوشتهٔ

محمدبن موسی خوارزمی

ترجمهٔ

حسین خدیوجم

انتشارات اطلاعات

۱۳۶۳

می کنی، می‌شود : شش درهم، و حاصل آن یك مال و یك جذر است که برابر است با شش درهم . آنگاه جـذر را پس از نصف‌کردن ، درمانند خودش ضرب کن ، می‌شود : یك چهارم، آن را برشش بیفزا، و جذر حاصل جمع را بگیر، و نصف جذری را که در مانند خودش ضرب کرده بودی ـ وعبارت است از نصف ـ از آن کم کن ، باقیمانده عبارت است از تعداد مردان نوبت اول که دراین مسئله دومرد است .

۲۹ـ اگر کسی بگوید: مالی است که چون آن را در دوسومش ضرب کنی پنج[1] می‌شود .

راه حل آن چنین است : اگر آن‌را در مانند خودش ضرب کنی هفت و نیم‌می‌شود.پس‌می گویی: آن‌مال‌جذرهفت ونیم‌است که باید در دوسوم جذر هفت‌و نیم‌ضرب شود، آنگاه دوسوم‌را دردوسوم ضرب می کنی می‌شود چهار نهم، و چهارنهم‌ضرب درهفت‌ونیم می‌شودسه‌و یك سوم،پس‌جذر سه ویك‌سوم‌عبارت‌است از دوسوم جذر هفت ونیم، آنگاه سه و یك سوم را درهفت و نیم ضرب می کنی می‌شود بیست و پنج ، جذر آن را می‌گیری پنج می‌شود .

۳۰ـ اگر کسی بگوید : مالی است که چون درسه جذر خودش ضرب شود پنج برابر مال اول می‌شود .

راه حل آن چنین‌است: چنان است که‌گفته باشد مالی رادرجذرش ضرب کردم به اندازهٔ یك مال و دوسوم مال اول شد ، پس مقدار جذر این مال یك درهم ودوسوم‌درهم است، و اصل مال دودرهم وهفت نهم درهم خواهد بود .

۳۱ـ اگر کسی بگوید: مالی است که چون یك سوم آن را کم

[1] خوارزمی این مسئله را با اندکی تفصیل تکرار کرده‌است . یعنی شکل دیگری از مسئلهٔ شمارهٔ ۱۴ است.

کنی وباقیمانده را درسه جذر آن مال ضرب کنی مقدار مال اول بدست می آید .

راه حل آن چـنین است : اگر تمام مال اول را ، پیش از کسر یك‌سوم، درسه‌عدد جذرخودش ضرب کنی‌می‌شود یك مال ونیم؛ زیرا دو سوم آن ضرب در سه جذر خودش می‌شود یك مال ، پس تمام آن ضرب در سه جذرش می‌شود یك مال ونیم ، وچون تمام آن‌را در یك جذر ضرب کنی می‌شود نصف مال، بنابراین جذر این مال نصف است واصل آن یك چهارم است ، پس دو سوم مال برابر است با یك ششم، وسه جذر مال یك درهم ونیم است، بنابراین هنگامی که یك‌ششم را در یك‌ونیم‌ضرب کنی‌یك‌چهارم به‌دست می آید وآن مقدارمال‌است.

۳۲ ـ اگر کسی بگوید : مالی است که چون چهار جذر آن را کنار بگذاری وسپس یك سوم باقیمانده را برداری، این یك‌سوم برابر است باچهار جذر مال.

راه حل آن چنین است : می‌دانی که یك سوم باقیـمانده برابر است با چهار جذر مال ، پس تمام باقیمانده برابر است با دوازده جذر آن . و چون چهار جذری را که کنار گذاشتی برآن بیفزایی می‌شود : شانزده‌جذر، و این تعدادجذر های‌مال است، و مقدار این‌مال‌دویست‌و پنجاه‌وشش‌است.

۳۳ ـ اگر کسی بگوید: مالی است که چون یك جذر آن‌را کنار بگذاری وجذر باقیمانده را برجذر آن بیفزایی دو درهم می‌شود[1] .

راه حل آن چنین است : این معادله بدین صورت در می‌آید : جذر مال ، به‌اضافهٔ جذر مال ، منهای یك جذر برابر است بادو درهم ، آنگاه‌یك‌جذر مال از آن ویك جذر مال از دو درهم کم می کنی ، معادله

[1] $x+\sqrt{x^{2}-x}=2$ بنابراین $x^{2}-x=(2-x)^{2}$ تا آخر

SAEED SALEHI, **Logic and Computation, & their interactions**, Arak 2019.

ROBERT OF CHESTER'S

LATIN TRANSLATION

OF THE

ALGEBRA OF AL-KHOWARIZMI

WITH AN INTRODUCTION, CRITICAL NOTES AND AN ENGLISH VERSION

BY

LOUIS CHARLES KARPINSKI
UNIVERSITY OF MICHIGAN

Muhammad ibn Mūsa, al-Khuwarazmi

New York
THE MACMILLAN COMPANY
LONDON: MACMILLAN AND COMPANY LIMITED
1915

C

equal to 6 units. I take one-half of the roots and I multiply the half by itself. I add the product to 6, and of this sum I take the root. The remainder obtained after subtracting one-half of the roots will designate the first number of girls, and this is two.

Fifteenth Problem

If from a square I subtract four of its roots and then take one-third of the remainder, finding this equal to four of the roots, the square will be 256.[1]

Explanation. Since one-third of the remainder is equal to four roots, you know that the remainder itself will equal 12 roots. Therefore add this to the four, giving 16 roots. This (16) is the root of the square.

Sixteenth Problem

From a square I subtract three of its roots and multiply the remainder by itself; the sum total of this multiplication equals the square.[2]

Explanation. It is evident that the remainder is equal to the root, which amounts to four. The square is 16.

These now are the sixteen problems which are seen to arise from the former ones, as we have explained. Hence by means of those things which have been set forth you will easily carry through any multiplication that you may wish to attempt in accordance with the art of restoration and opposition.

CHAPTER ON MERCANTILE TRANSACTIONS[3]

Mercantile transactions and all things pertaining thereto involve two ideas and four numbers.[4] Of these numbers the first is called by the Arabs Almuzahar and is the first one proposed. The second is called Alszian, and recognized as second by means of the first. The third, Almuhen, is unknown. The fourth, Alchemon, is obtained by means of the first and second. Further, these four numbers are so related that the first of them, the measure, is inversely proportional to the last, which is cost. Moreover, three of these numbers are always given or known and the fourth is unknown, and this

[1] Rosen, p. 66; Libri, p. 296. $\frac{1}{3}(x^2 - 4x) = 4x$.
In the Arabic text these two problems precede: $x^2 \cdot 3x = 5x^2$ and $(x^2 - \frac{1}{3}x^2) \cdot 3x = x^2$.

[2] Rosen, p. 67; Libri, p. 296. $(x^2 - 3x)^2 = x^2$, whence $x^2 - 3x = x$.
The problem, $x + \sqrt{x^2 - x} = 2$, precedes. This is one of two problems given in the German excerpt of 1461 from the algebra of Al-Khowarizmi (Gerhardt, *Monatsbericht d. Königl. Akad. der Wissenschaften zu Berlin*, 1870, pp. 142–143).

[3] The famous 'rule of three' is the subject of discussion in this chapter.

[4] The two ideas appear to be the notions of quantity and cost; the four numbers represent unit of measure and price per unit, quantity desired and cost of the same. These four technical terms are *al-musa' 'ir*, *al-si'r*, *al-thaman*, and *al-muthammin*, and further *al-maqūl*; see p. 44.

SAEED SALEHI, **Logic and Computation, & their interactions**, Arak 2019.

Coding Mathematics

How to write (code) mathematical statements (as input strings)?

Example from Al-Khwarizmi: If from a square, I subtract four of its roots and then take one-third of the remainder, finding this equal to four of the roots, the square will be 256.

Modern Notation: If I have $\frac{1}{3}(x^2 - 4x) = 4x$, then $x^2 = 256$.

More Modern: $\forall x[\frac{1}{3}(x^2 - 4x) = 4x \longrightarrow x^2 = 256]$.

This holds in the domain $\mathbb{N} - \{0\} = \{1, 2, 3, \cdots\}$ (but not in $\mathbb{N}$).

Indeed, $\mathbb{N} \models \forall x[\frac{1}{3}(x^2 - 4x) = 4x \longrightarrow x = 16 \vee x = 0]$.

Coding Mathematics

How to write (code) mathematical statements (as input strings)?

Example from Al-Khwarizmi: If from a square, I subtract four of its roots and then take one-third of the remainder, finding this equal to four of the roots, the square will be 256.

Modern Notation: If I have $\frac{1}{3}(x^2 - 4x) = 4x$, then $x^2 = 256$.

More Modern: $\forall x[\frac{1}{3}(x^2 - 4x) = 4x \longrightarrow x^2 = 256]$.

This holds in the domain $\mathbb{N} - \{0\} = \{1, 2, 3, \cdots\}$ (but not in $\mathbb{N}$).

Indeed, $\mathbb{N} \models \forall x[\frac{1}{3}(x^2 - 4x) = 4x \longrightarrow x = 16 \vee x = 0]$.

Coding Mathematics

How to write (code) mathematical statements (as input strings)?

Example from Al-Khwarizmi: If from a square, I subtract four of its roots and then take one-third of the remainder, finding this equal to four of the roots, the square will be 256.

Modern Notation: If I have $\frac{1}{3}(x^2 - 4x) = 4x$, then $x^2 = 256$.

More Modern: $\forall x[\frac{1}{3}(x^2 - 4x) = 4x \longrightarrow x^2 = 256]$.

This holds in the domain $\mathbb{N} - \{0\} = \{1, 2, 3, \cdots\}$ (but not in $\mathbb{N}$).

Indeed, $\mathbb{N} \models \forall x[\frac{1}{3}(x^2 - 4x) = 4x \longrightarrow x = 16 \vee x = 0]$.

Coding Mathematics

How to write (code) mathematical statements (as input strings)?

Example from Al-Khwarizmi: If from a square, I subtract four of its roots and then take one-third of the remainder, finding this equal to four of the roots, the square will be 256.

Modern Notation: If I have $\frac{1}{3}(x^2 - 4x) = 4x$, then $x^2 = 256$.

More Modern: $\forall x[\frac{1}{3}(x^2 - 4x) = 4x \longrightarrow x^2 = 256]$.

This holds in the domain $\mathbb{N} - \{0\} = \{1, 2, 3, \cdots\}$ (but not in $\mathbb{N}$).

Indeed, $\mathbb{N} \models \forall x[\frac{1}{3}(x^2 - 4x) = 4x \longrightarrow x = 16 \vee x = 0]$.

INSTITUTE FOR RESEARCH IN FUNDAMENTAL SCIENCES

Coding Mathematics

How to write (code) mathematical statements (as input strings)?

Example from Al-Khwarizmi: If from a square, I subtract four of its roots and then take one-third of the remainder, finding this equal to four of the roots, the square will be 256.

Modern Notation: If I have $\frac{1}{3}(x^2 - 4x) = 4x$, then $x^2 = 256$.

More Modern: $\forall x[\frac{1}{3}(x^2 - 4x) = 4x \longrightarrow x^2 = 256]$.

This holds in the domain $\mathbb{N} - \{0\} = \{1, 2, 3, \cdots\}$ (but not in $\mathbb{N}$).

Indeed, $\mathbb{N} \models \forall x[\frac{1}{3}(x^2 - 4x) = 4x \longrightarrow x = 16 \vee x = 0]$.

Coding Mathematics

How to write (code) mathematical statements (as input strings)?

Example from Al-Khwarizmi: If from a square, I subtract four of its roots and then take one-third of the remainder, finding this equal to four of the roots, the square will be 256.

Modern Notation: If I have $\frac{1}{3}(x^2 - 4x) = 4x$, then $x^2 = 256$.

More Modern: $\forall x[\frac{1}{3}(x^2 - 4x) = 4x \longrightarrow x^2 = 256]$.

This holds in the domain $\mathbb{N} - \{0\} = \{1, 2, 3, \cdots\}$ (but not in $\mathbb{N}$).

Indeed, $\mathbb{N} \models \forall x[\frac{1}{3}(x^2 - 4x) = 4x \longrightarrow x = 16 \vee x = 0]$.

Coding Mathematics

How to write (code) mathematical statements (as input strings)?

Example from Al-Khwarizmi: If from a square, I subtract four of its roots and then take one-third of the remainder, finding this equal to four of the roots, the square will be 256.

Modern Notation: If I have $\frac{1}{3}(x^2 - 4x) = 4x$, then $x^2 = 256$.

More Modern: $\forall x[\frac{1}{3}(x^2 - 4x) = 4x \longrightarrow x^2 = 256]$.

This holds in the domain $\mathbb{N} - \{0\} = \{1, 2, 3, \cdots\}$ (but not in $\mathbb{N}$).

Indeed, $\mathbb{N} \models \forall x[\frac{1}{3}(x^2 - 4x) = 4x \longrightarrow x = 16 \vee x = 0]$.

Computing the Solution

Khwarizmi's Explanation:

Since one-third of the remainder is equal to four roots, you know that the remainder itself will equal 12 roots. Therefore, add this to the four, giving 16 roots. This (16) is the root of the square.

Modern Notation: Since $\frac{1}{3}(x^2 - 4x) = 4x$, then $x^2 - 4x = 12x$.
Therefore, $x^2 = 16x$. Thus, $x = 16$.

In fact, $\mathcal{A}\mathfrak{rithmetic} \vdash \forall x \big[\frac{1}{3}(x^2 - 4x) = 4x \ (\& x \neq 0) \longrightarrow x = 16\big]$.

Computing the Solution

Khwarizmi's Explanation:

Since one-third of the remainder is equal to four roots, you know that the remainder itself will equal 12 roots. Therefore, add this to the four, giving 16 roots. This (16) is the root of the square.

Modern Notation: Since $\frac{1}{3}(x^2-4x)=4x$, then $x^2-4x=12x$.
Therefore, $x^2=16x$. Thus, $x=16$.

In fact, $\mathfrak{Arithmetic}\vdash\forall x\big[\frac{1}{3}(x^2-4x)=4x\ (\&x\neq 0)\longrightarrow x=16\big]$.

Computing the Solution

Khwarizmi's Explanation:

Since one-third of the remainder is equal to four roots, you know that the remainder itself will equal 12 roots. Therefore, add this to the four, giving 16 roots. This (16) is the root of the square.

Modern Notation: Since $\frac{1}{3}(x^2 - 4x) = 4x$, then $x^2 - 4x = 12x$.
Therefore, $x^2 = 16x$. Thus, $x = 16$.

In fact, $\mathcal{A}rithmetic \vdash \forall x \left[\frac{1}{3}(x^2 - 4x) = 4x \ (\& x \neq 0) \longrightarrow x = 16\right]$.

Computing the Solution

Khwarizmi's Explanation:

Since one-third of the remainder is equal to four roots, you know that the remainder itself will equal 12 roots. Therefore, add this to the four, giving 16 roots. This (16) is the root of the square.

Modern Notation: Since $\frac{1}{3}(x^2 - 4x) = 4x$, then $x^2 - 4x = 12x$.

Therefore, $x^2 = 16x$. Thus, $x = 16$.

In fact, $\mathcal{A}rithmetic \vdash \forall x\big[\frac{1}{3}(x^2 - 4x) = 4x \ (\& x \neq 0) \longrightarrow x = 16\big]$.

Computing the Solution

Khwarizmi's Explanation:

Since one-third of the remainder is equal to four roots, you know that the remainder itself will equal 12 roots. Therefore, add this to the four, giving 16 roots. This (16) is the root of the square.

Modern Notation: Since $\frac{1}{3}(x^2 - 4x) = 4x$, then $x^2 - 4x = 12x$. Therefore, $x^2 = 16x$. Thus, $x = 16$.

In fact, $\mathfrak{Arithmetic} \vdash \forall x \left[\frac{1}{3}(x^2 - 4x) = 4x \; (\& x \neq 0) \longrightarrow x = 16\right]$.

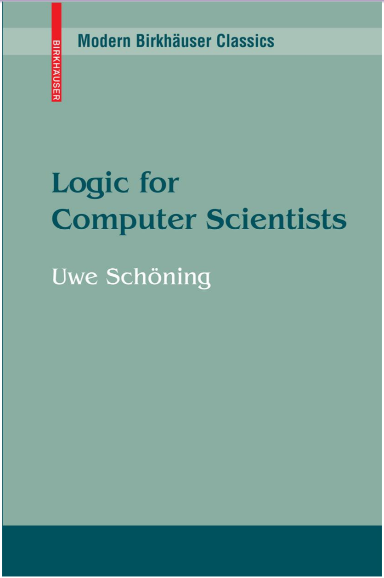
Computing the Solution

Khwarizmi's Explanation:

Since one-third of the remainder is equal to four roots, you know that the remainder itself will equal 12 roots. Therefore, add this to the four, giving 16 roots. This (16) is the root of the square.

Modern Notation: Since $\frac{1}{3}(x^2 - 4x) = 4x$, then $x^2 - 4x = 12x$. Therefore, $x^2 = 16x$. Thus, $x = 16$.

In fact, $\mathcal{A}\mathfrak{rithmetic} \vdash \forall x\big[\frac{1}{3}(x^2 - 4x) = 4x \ (\& x \neq 0) \longrightarrow x = 16\big]$.



SAEED SALEHI, **Logic and Computation, & their interactions**, Arak 2019.

Logic for Computer Scientists

Uwe Schöning

Reprint of the 1989 Edition

Birkhäuser
Boston • Basel • Berlin

Uwe Schöning
Abt. Theoretische Informatik
Universität Ulm
Oberer Eselsberg
D-89069 Ulm
Germany

English hardcover edition originally published as Volume 8 in the series *Progress in Computer Science and Applied Logic.*

German edition was published in 1987 as *Logik für Informatiker* by Wissenschaftsverlag, Mannheim • Vienna • Zürich.

ISBN-13: 978-0-8176-4762-9 e-ISBN-13: 978-0-8176-4763-6
DOI: 10.1007/978-0-8176-4763-6

Library of Congress Control Number: 2007940259

Cover design by Alex Gerasev.

Printed on acid-free paper.

9 8 7 6 5 4 3 2 1

www.birkhauser.com

Uwe Schöning

Logic for Computer Scientists

With 34 Illustrations

1989

Birkhäuser
Boston · Basel · Berlin

Uwe Schöning
Abt. Theoretische Informatik
Universität Ulm
Oberer Eselsberg
D-89069 Ulm
Germany

Library of Congress Cataloging-in-Publication Data
Schöning, Uwe, 1955-
Logic for computer scientists / Uwe Schöning
p. cm. — (Progress in computer science and applied logic ; v.8)
Includes bibliographical references.
ISBN 0-8176-3453-0 (alk. paper).
1. Logic, Symbolic and mathematical 2. Logic programming.
I. Title.
QA9.S363 1989 89-17864
511.3—dc20 CIP

Logic for Computer Scientists was originally published in 1987
as *Logik für Informatiker* by Wissenschaftsverlag, Mannheim • Vienna • Zürich.

Printed on acid-free paper.

Third printing 1999

Birkhäuser

ISBN 0-8176-3453-3
ISBN 3-7643-3453-3

Typeset by the author using LaTeX.
Printed and bound by Quinn-Woodbine, Woodbine, NJ.
Printed in the United States of America.

9 8 7 6 5 4 3

Preface

By the development of new fields and applications, such as Automated Theorem Proving and Logic Programming, Logic has obtained a new and important role in Computer Science. The traditional mathematical way of dealing with Logic is in some respect not tailored for Computer Science applications. This book emphasizes such Computer Science aspects in Logic. It arose from a series of lectures in 1986 and 1987 on Computer Science Logic at the EWH University in Koblenz, Germany. The goal of this lecture series was to give the undergraduate student an early and theoretically well-founded access to modern applications of Logic in Computer Science.

A minimal mathematical basis is required, such as an understanding of the set theoretic notation and knowledge about the basic mathematical proof techniques (like induction). More sophisticated mathematical knowledge is not a precondition to read this book. Acquaintance with some conventional programming language, like PASCAL, is assumed.

Several people helped in various ways in the preparation process of the original German version of this book: Johannes Köbler, Eveline and Rainer Schuler, and Hermann Engesser from B.I. Wissenschaftsverlag.

Regarding the English version, I want to express my deep gratitude to Prof. Ronald Book. Without him, this translated version of the book would not have been possible.

Koblenz, June 1989 U. Schöning

Contents

SAEED SALEHI, **Logic and Computation, & their interactions**, Arak 2019.

Introduction

Formal Logic investigates how assertions are combined and connected, how theorems formally can be deduced from certain axioms, and what kind of object a proof is. In Logic there is a consequent separation of syntactical notions (formulas, proofs) – these are essentially strings of symbols built up according to certain rules – and semantical notions (truth values, models) – these are "interpretations", assignments of "meanings" to the syntactical objects.

Because of its development from philosophy, the questions investigated in Logic were originally of a more fundamental character, like: What is truth? What theories are axiomatizable? What is a model of a certain axiom system?, and so on. In general, it can be said that traditional Logic is oriented to fundamental questions, whereas Computer Science is interested in what is programmable. This book provides some unification of both aspects.

Computer Science has utilized many subfields of Logic in areas such as program verification, semantics of programming languages, automated theorem proving, and logic programming. This book concentrates on those aspects of Logic which have applications in Computer Science, especially theorem proving and logic programming. From the very beginning, education in Computer Science supports the idea of strict separation between syntax and semantics (of programming languages). Also, recursive definitions, equations and programs are a familiar thing to a first year Computer Science student. This book is oriented in its style of presentation to this style.

In the first Chapter, propositional logic is introduced with emphasis on the resolution calculus and Horn formulas (which have their particular Computer Science applications in later sections). The second Chapter introduces the predicate logic. Again, Computer Science aspects are emphasized, like undecidability and semi-decidability of predicate logic, Herbrand's the-

ory, and building upon this, the resolution calculus (and its refinements) for predicate logic is discussed. Most modern theorem proving programs are based on resolution refinements as discussed in Section 2.6.

The third Chapter leads to the special version of resolution (SLD-resolution) used in logic programming systems, as realized in the logic programming language PROLOG (= *Pro*gramming in *Log*ic). The idea of this book, though, is not to be a programmer's manual for PROLOG. Rather, the aim is to give the theoretical foundations for an understanding of logic programming in general.

Exercise 1: "What is the secret of your long life?" a centenarian was asked. "I strictly follow my diet: If I don't drink beer for dinner, then I always have fish. Any time I have both beer and fish for dinner, then I do without ice cream. If I have ice cream or don't have beer, then I never eat fish." The questioner found this answer rather confusing. Can you simplify it?

Find out which formal methods (diagrams, graphs, tables, etc.) you used to solve this Exercise. You have done your own first steps to develop a Formal Logic!

SAEED SALEHI, **Logic and Computation, & their interactions**, Arak 2019.

Proving or Computing?

Exercise 1: "What is the secret of your long life?"
a centenarian was asked.

"I strictly follow my diet:
If I don't drink beer for dinner, then I always have fish.
If I have both beer and fish for dinner, then I do without ice cream.
If I have ice cream or don't have beer, then I never eat fish."

The questioner found this answer rather confusing.
Can you simplify it?

Proving or Computing?

Exercise 1: "What is the secret of your long life?"
a centenarian was asked.

"I strictly follow my diet:
If I don't drink beer for dinner, then I always have fish.
If I have both beer and fish for dinner, then I do without ice cream.
If I have ice cream or don't have beer, then I never eat fish."

The questioner found this answer rather confusing.
Can you simplify it?

Proving or Computing?

Exercise 1: "What is the secret of your long life?"
a centenarian was asked.

"I strictly follow my diet:
If I don't drink beer for dinner, then I always have fish.
If I have both beer and fish for dinner, then I do without ice cream.
If I have ice cream or don't have beer, then I never eat fish."

The questioner found this answer rather confusing.
Can you simplify it?

Proving or Computing?

Exercise 1: "What is the secret of your long life?"
a centenarian was asked.

"I strictly follow my diet:
If I don't drink beer for dinner, then I always have fish.
If I have both beer and fish for dinner, then I do without ice cream.
If I have ice cream or don't have beer, then I never eat fish."

The questioner found this answer rather confusing.
Can you simplify it?

Proving or Computing?

$B =$ beer $\quad F =$ fish $\quad I =$ ice cream

If I don't drink beer for dinner, then I always have fish.

$$\neg B \rightarrow F$$

Any time I have both beer and fish for dinner, then I do without ice cream.

$$B \wedge F \rightarrow \neg I$$

If I have ice cream or don't have beer, then I never eat fish.

$$I \vee \neg B \rightarrow \neg F$$

Proving or Computing?

$B =$ beer $F =$ fish $I =$ ice cream

If I don't drink beer for dinner, then I always have fish.

$\neg B \to F$

Any time I have both beer and fish for dinner, then I do without ice cream.

$B \wedge F \to \neg I$

If I have ice cream or don't have beer, then I never eat fish.

$I \vee \neg B \to \neg F$

Proving or Computing?

$B =$ beer $F =$ fish $I =$ ice cream

If I don't drink beer for dinner, then I always have fish.

$\neg B \to F$

Any time I have both beer and fish for dinner, then I do without ice cream.

$B \wedge F \to \neg I$

If I have ice cream or don't have beer, then I never eat fish.

$I \vee \neg B \to \neg F$

Proving or Computing?

$B =$ beer $\qquad F =$ fish $\qquad I =$ ice cream

If I don't drink beer for dinner, then I always have fish.

$$\neg B \to F$$

Any time I have both beer and fish for dinner, then I do without ice cream.

$$B \wedge F \to \neg I$$

If I have ice cream or don't have beer, then I never eat fish.

$$I \vee \neg B \to \neg F$$

Proving or Computing?

$B =$ beer $\quad F =$ fish $\quad I =$ ice cream

If I don't drink beer for dinner, then I always have fish.

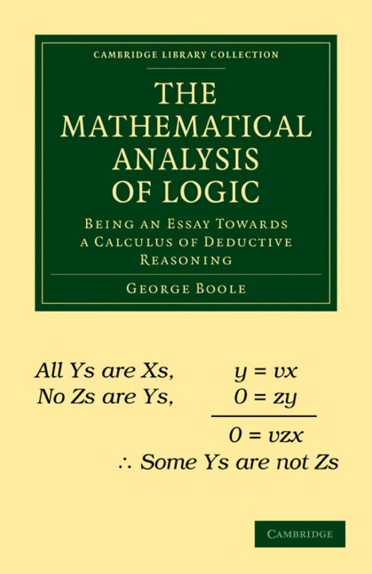
$$\neg B \rightarrow F$$

Any time I have both beer and fish for dinner, then I do without ice cream.

$$B \wedge F \rightarrow \neg I$$

If I have ice cream or don't have beer, then I never eat fish.

$$I \vee \neg B \rightarrow \neg F$$



The Mathematical Analysis of Logic

Being an Essay Towards a Calculus of Deductive Reasoning

GEORGE BOOLE

CAMBRIDGE UNIVERSITY PRESS

CAMBRIDGE UNIVERSITY PRESS

Cambridge, New York, Melbourne, Madrid, Cape Town, Singapore,
São Paolo, Delhi, Dubai, Tokyo

Published in the United States of America by Cambridge University Press, New York

www.cambridge.org
Information on this title: www.cambridge.org/9781108001014

This edition first published 1847
This digitally printed version 2009

ISBN 978-1-108-00101-4 Paperback

THE MATHEMATICAL ANALYSIS OF LOGIC,

BEING AN ESSAY TOWARDS A CALCULUS OF DEDUCTIVE REASONING.

BY GEORGE BOOLE.

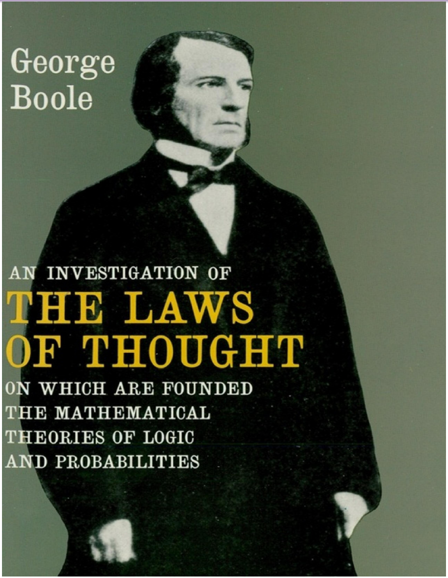
Ἐπικοινωνοῦσι δὲ πᾶσαι αἱ ἐπιστῆμαι ἀλλήλαις κατὰ τὰ κοινά. Κοινὰ δὲ λέγω, οἷς χρῶνται ὡς ἐκ τούτων ἀποδεικνύντες· ἀλλ᾽ οὐ περὶ ὧν δεικνύουσιν, οὐδὲ ὃ δεικνύουσι.

ARISTOTLE, *Anal. Post.*, lib. I. cap. XI.

CAMBRIDGE:
MACMILLAN, BARCLAY, & MACMILLAN;
LONDON: GEORGE BELL.

1847

SAEED SALEHI, **Logic and Computation, & their interactions**, Arak 2019.


George
Boole
AN INVESTIGATION OF
THE LAWS
OF THOUGHT
ON WHICH ARE FOUNDED
THE MATHEMATICAL
THEORIES OF LOGIC
AND PROBABILITIES

AN INVESTIGATION

OF

THE LAWS OF THOUGHT

ON WHICH ARE FOUNDED

THE MATHEMATICAL THEORIES OF LOGIC
AND PROBABILITIES

BY

GEORGE BOOLE, L.L.D.

DOVER PUBLICATIONS, INC., NEW YORK

Propositional Logic

- Connectives $\wedge$, $\vee$, $\neg$, $\rightarrow$, $\leftrightarrow$
- Atomic Propositions (without a truth value) $P, Q, R, \cdots$
- More Complex Propositions and Truth Tables

Propositional Logic

- Connectives $\wedge$, $\vee$, $\neg$, $\rightarrow$, $\leftrightarrow$
- Atomic Propositions (without a truth value) $P, Q, R, \cdots$
- More Complex Propositions and Truth Tables

SAEED SALEHI, **Logic and Computation, & their interactions**, Arak 2019.

Propositional Logic

- Connectives $\wedge$, $\vee$, $\neg$, $\rightarrow$, $\leftrightarrow$
- Atomic Propositions (without a truth value) $P, Q, R, \cdots$
- More Complex Propositions and Truth Tables

Saeed Salehi, **Logic and Computation, & their interactions**, Arak 2019.

Propositional Logic

- Connectives $\wedge$, $\vee$, $\neg$, $\rightarrow$, $\leftrightarrow$
- Atomic Propositions (without a truth value) $P, Q, R, \cdots$
- More Complex Propositions and Truth Tables

Proving or Computing?

$(\neg B \to F), (B \wedge F \to \neg I), (I \vee \neg B \to \neg F)$

$\varphi = (\neg B \to F) \wedge (B \wedge F \to \neg I) \wedge (I \vee \neg B \to \neg F)$

B	F	I	$\neg B \to F$	$B \wedge F \to \neg I$	$I \vee \neg B \to \neg F$	φ
0	0	0	0	1	1	0
0	0	1	0	1	1	0
0	1	0	1	1	0	0
0	1	1	1	1	0	0
1	0	0	1	1	1	1
1	0	1	1	1	1	1
1	1	0	1	1	1	1
1	1	1	1	0	0	0

https://web.stanford.edu/class/cs103/tools/truth-table-tool/

Proving or Computing?

$$(\neg B \to F),\ (B \wedge F \to \neg I),\ (I \vee \neg B \to \neg F)$$

$$\varphi = (\neg B \to F) \wedge (B \wedge F \to \neg I) \wedge (I \vee \neg B \to \neg F)$$

B	F	I	$\neg B \to F$	$B \wedge F \to \neg I$	$I \vee \neg B \to \neg F$	φ
0	0	0	0	1	1	0
0	0	1	0	1	1	0
0	1	0	1	1	0	0
0	1	1	1	1	0	0
1	0	0	1	1	1	1
1	0	1	1	1	1	1
1	1	0	1	1	1	1
1	1	1	1	0	0	0

https://web.stanford.edu/class/cs103/tools/truth-table-tool/

Proving or Computing?

$$(\neg B \to F),\ (B \wedge F \to \neg I),\ (I \vee \neg B \to \neg F)$$

$$\varphi = (\neg B \to F) \wedge (B \wedge F \to \neg I) \wedge (I \vee \neg B \to \neg F)$$

B	F	I	$\neg B \to F$	$B \wedge F \to \neg I$	$I \vee \neg B \to \neg F$	φ
0	0	0	0	1	1	0
0	0	1	0	1	1	0
0	1	0	1	1	0	0
0	1	1	1	1	0	0
1	0	0	1	1	1	1
1	0	1	1	1	1	1
1	1	0	1	1	1	1
1	1	1	1	0	0	0

https://web.stanford.edu/class/cs103/tools/truth-table-tool/

Proving or Computing?

$$(\neg B \to F),\ (B \wedge F \to \neg I),\ (I \vee \neg B \to \neg F)$$
$$\varphi = (\neg B \to F) \wedge (B \wedge F \to \neg I) \wedge (I \vee \neg B \to \neg F)$$

B	F	I	$\neg B \to F$	$B \wedge F \to \neg I$	$I \vee \neg B \to \neg F$	φ
0	0	0	0	1	1	0
0	0	1	0	1	1	0
0	1	0	1	1	0	0
0	1	1	1	1	0	0
1	0	0	1	1	1	1
1	0	1	1	1	1	1
1	1	0	1	1	1	1
1	1	1	1	0	0	0

https://web.stanford.edu/class/cs103/tools/truth-table-tool/

Proving or Computing?

$$(\neg B \to F), (B \wedge F \to \neg I), (I \vee \neg B \to \neg F)$$

$$\varphi = (\neg B \to F) \wedge (B \wedge F \to \neg I) \wedge (I \vee \neg B \to \neg F)$$

B	F	I	$\neg B \to F$	$B \wedge F \to \neg I$	$I \vee \neg B \to \neg F$	φ
0	0	0	0	1	1	0
0	0	1	0	1	1	0
0	1	0	1	1	0	0
0	1	1	1	1	0	0
1	0	0	1	1	1	1
1	0	1	1	1	1	1
1	1	0	1	1	1	1
1	1	1	1	0	0	0

https://web.stanford.edu/class/cs103/tools/truth-table-tool/

Proving or Computing?

$$(\neg B \to F),\ (B \wedge F \to \neg I),\ (I \vee \neg B \to \neg F)$$

$$\varphi = (\neg B \to F) \wedge (B \wedge F \to \neg I) \wedge (I \vee \neg B \to \neg F)$$

B	F	I	$\neg B \to F$	$B \wedge F \to \neg I$	$I \vee \neg B \to \neg F$	φ
0	0	0	0	1	1	0
0	0	1	0	1	1	0
0	1	0	1	1	0	0
0	1	1	1	1	0	0
1	0	0	1	1	1	1
1	0	1	1	1	1	1
1	1	0	1	1	1	1
1	1	1	1	0	0	0

https://web.stanford.edu/class/cs103/tools/truth-table-tool/

Proving or Computing?

$$(\neg B \to F),\ (B \wedge F \to \neg I),\ (I \vee \neg B \to \neg F)$$
$$\varphi = (\neg B \to F) \wedge (B \wedge F \to \neg I) \wedge (I \vee \neg B \to \neg F)$$

B	F	I	$\neg B \to F$	$B \wedge F \to \neg I$	$I \vee \neg B \to \neg F$	φ
0	0	0	0	1	1	0
0	0	1	0	1	1	0
0	1	0	1	1	0	0
0	1	1	1	1	0	0
1	0	0	1	1	1	1
1	0	1	1	1	1	1
1	1	0	1	1	1	1
1	1	1	1	0	0	0

https://web.stanford.edu/class/cs103/tools/truth-table-tool/

Proving or Computing?

$$(\neg B \to F), (B \wedge F \to \neg I), (I \vee \neg B \to \neg F)$$

$$\varphi = (\neg B \to F) \wedge (B \wedge F \to \neg I) \wedge (I \vee \neg B \to \neg F)$$

B	F	I	$\neg B \to F$	$B \wedge F \to \neg I$	$I \vee \neg B \to \neg F$	φ
0	0	0	0	1	1	0
0	0	1	0	1	1	0
0	1	0	1	1	0	0
0	1	1	1	1	0	0
1	0	0	1	1	1	1
1	0	1	1	1	1	1
1	1	0	1	1	1	1
1	1	1	1	0	0	0

https://web.stanford.edu/class/cs103/tools/truth-table-tool/

Proving or Computing?

$$(\neg B \to F), (B \wedge F \to \neg I), (I \vee \neg B \to \neg F)$$

$$\varphi = (\neg B \to F) \wedge (B \wedge F \to \neg I) \wedge (I \vee \neg B \to \neg F)$$

B	F	I	$\neg B \to F$	$B \wedge F \to \neg I$	$I \vee \neg B \to \neg F$	φ
0	0	0	0	1	1	0
0	0	1	0	1	1	0
0	1	0	1	1	0	0
0	1	1	1	1	0	0
1	0	0	1	1	1	1
1	0	1	1	1	1	1
1	1	0	1	1	1	1
1	1	1	1	0	0	0

https://web.stanford.edu/class/cs103/tools/truth-table-tool/

Proving or Computing?

$$(\neg B \to F),\ (B \wedge F \to \neg I),\ (I \vee \neg B \to \neg F)$$

$$\varphi = (\neg B \to F) \wedge (B \wedge F \to \neg I) \wedge (I \vee \neg B \to \neg F)$$

B	F	I	$\neg B \to F$	$B \wedge F \to \neg I$	$I \vee \neg B \to \neg F$	φ
0	0	0	0	1	1	0
0	0	1	0	1	1	0
0	1	0	1	1	0	0
0	1	1	1	1	0	0
1	0	0	1	1	1	1
1	0	1	1	1	1	1
1	1	0	1	1	1	1
1	1	1	1	0	0	0

https://web.stanford.edu/class/cs103/tools/truth-table-tool/

Proving or Computing?

$$(\neg B \to F), (B \wedge F \to \neg I), (I \vee \neg B \to \neg F)$$

$$\varphi = (\neg B \to F) \wedge (B \wedge F \to \neg I) \wedge (I \vee \neg B \to \neg F)$$

B	F	I	$\neg B \to F$	$B \wedge F \to \neg I$	$I \vee \neg B \to \neg F$	φ
0	0	0	0	1	1	0
0	0	1	0	1	1	0
0	1	0	1	1	0	0
0	1	1	1	1	0	0
1	0	0	1	1	1	1
1	0	1	1	1	1	1
1	1	0	1	1	1	1
1	1	1	1	0	0	0

https://web.stanford.edu/class/cs103/tools/truth-table-tool/

Proving or Computing?

$$(\neg B \to F),\ (B \wedge F \to \neg I),\ (I \vee \neg B \to \neg F)$$

$$\varphi = (\neg B \to F) \wedge (B \wedge F \to \neg I) \wedge (I \vee \neg B \to \neg F)$$

B	F	I	$\neg B \to F$	$B \wedge F \to \neg I$	$I \vee \neg B \to \neg F$	φ
0	0	0	0	1	1	0
0	0	1	0	1	1	0
0	1	0	1	1	0	0
0	1	1	1	1	0	0
1	0	0	1	1	1	1
1	0	1	1	1	1	1
1	1	0	1	1	1	1
1	1	1	1	0	0	0

https://web.stanford.edu/class/cs103/tools/truth-table-tool/

Proving or Computing?

$$(\neg B \rightarrow F), (B \wedge F \rightarrow \neg I), (I \vee \neg B \rightarrow \neg F)$$

$$\varphi = (\neg B \rightarrow F) \wedge (B \wedge F \rightarrow \neg I) \wedge (I \vee \neg B \rightarrow \neg F)$$

B	F	I	$\neg B \rightarrow F$	$B \wedge F \rightarrow \neg I$	$I \vee \neg B \rightarrow \neg F$	φ
0	0	0	0	1	1	0
0	0	1	0	1	1	0
0	1	0	1	1	0	0
0	1	1	1	1	0	0
1	0	0	1	1	1	1
1	0	1	1	1	1	1
1	1	0	1	1	1	1
1	1	1	1	0	0	0

https://web.stanford.edu/class/cs103/tools/truth-table-tool/

Proving or Computing?

B	F	I	$\neg B \to F$	$B \wedge F \to \neg I$	$I \vee \neg B \to \neg F$	φ
1	0	0	1	1	1	1
1	0	1	1	1	1	1
1	1	0	1	1	1	1

$\varphi \equiv$
$(B \wedge \neg F \wedge \neg I) \vee$
$(B \wedge \neg F \wedge I) \vee$
$(B \wedge F \wedge \neg I)$

$$\equiv (B \wedge \neg F) \vee (B \wedge F \wedge \neg I) \equiv B \wedge (\neg F \vee [F \wedge \neg I]) \equiv B \wedge (\neg F \vee \neg I)$$

$$\varphi \equiv B \wedge \neg (F \wedge I)$$

Proving or Computing?

B	F	I	$\neg B \to F$	$B \wedge F \to \neg I$	$I \vee \neg B \to \neg F$	φ
1	0	0	1	1	1	1
1	0	1	1	1	1	1
1	1	0	1	1	1	1

$\varphi \equiv$

$(B \wedge \neg F \wedge \neg I) \vee$

$(B \wedge \neg F \wedge I) \vee$

$(B \wedge F \wedge \neg I)$

$\equiv (B \wedge \neg F) \vee (B \wedge F \wedge \neg I) \equiv B \wedge (\neg F \vee [F \wedge \neg I]) \equiv B \wedge (\neg F \vee \neg I)$

$\varphi \equiv B \wedge \neg (F \wedge I)$

Proving or Computing?

B	F	I	$\neg B \to F$	$B \wedge F \to \neg I$	$I \vee \neg B \to \neg F$	φ
1	0	0	1	1	1	1
1	0	1	1	1	1	1
1	1	0	1	1	1	1

$\varphi \equiv$
$(B \wedge \neg F \wedge \neg I) \vee$
$(B \wedge \neg F \wedge I) \vee$
$(B \wedge F \wedge \neg I)$

$$\equiv (B \wedge \neg F) \vee (B \wedge F \wedge \neg I) \equiv B \wedge (\neg F \vee [F \wedge \neg I]) \equiv B \wedge (\neg F \vee \neg I)$$

$$\varphi \equiv B \wedge \neg (F \wedge I)$$

Proving or Computing?

B	F	I	$\neg B \to F$	$B \wedge F \to \neg I$	$I \vee \neg B \to \neg F$	φ
1	0	0	1	1	1	1
1	0	1	1	1	1	1
1	1	0	1	1	1	1

$\varphi \equiv$
$(B \wedge \neg F \wedge \neg I) \vee$
$(B \wedge \neg F \wedge I) \vee$
$(B \wedge F \wedge \neg I)$
$\equiv (B \wedge \neg F) \vee (B \wedge F \wedge \neg I) \equiv B \wedge (\neg F \vee [F \wedge \neg I]) \equiv B \wedge (\neg F \vee \neg I)$

$\varphi \equiv B \wedge \neg(F \wedge I)$

Proving or Computing?

B	F	I	$\neg B \to F$	$B \wedge F \to \neg I$	$I \vee \neg B \to \neg F$	φ
1	0	0	1	1	1	1
1	0	1	1	1	1	1
1	1	0	1	1	1	1

$\varphi \equiv$
$(B \wedge \neg F \wedge \neg I) \vee$
$(B \wedge \neg F \wedge I) \vee$
$(B \wedge F \wedge \neg I)$
$\equiv (B \wedge \neg F) \vee (B \wedge F \wedge \neg I) \equiv B \wedge (\neg F \vee [F \wedge \neg I]) \equiv B \wedge (\neg F \vee \neg I)$
$\varphi \equiv B \wedge \neg(F \wedge I)$

Proving or Computing?

B	F	I	$\neg B \to F$	$B \wedge F \to \neg I$	$I \vee \neg B \to \neg F$	φ
1	0	0	1	1	1	1
1	0	1	1	1	1	1
1	1	0	1	1	1	1

$\varphi \equiv$
$(B \wedge \neg F \wedge \neg I) \vee$
$(B \wedge \neg F \wedge I) \vee$
$(B \wedge F \wedge \neg I)$
$\equiv (B \wedge \neg F) \vee (B \wedge F \wedge \neg I) \equiv B \wedge (\neg F \vee [F \wedge \neg I]) \equiv B \wedge (\neg F \vee \neg I)$

$\varphi \equiv B \wedge \neg (F \wedge I)$

Proving or Computing?

B	F	I	$\neg B \to F$	$B \wedge F \to \neg I$	$I \vee \neg B \to \neg F$	φ
1	0	0	1	1	1	1
1	0	1	1	1	1	1
1	1	0	1	1	1	1

$\varphi \equiv$
$(B \wedge \neg F \wedge \neg I) \vee$
$(B \wedge \neg F \wedge I) \vee$
$(B \wedge F \wedge \neg I)$
$\equiv (B \wedge \neg F) \vee (B \wedge F \wedge \neg I) \equiv B \wedge (\neg F \vee [F \wedge \neg I]) \equiv B \wedge (\neg F \vee \neg I)$

$\varphi \equiv B \wedge \neg (F \wedge I)$

Axiom / Axiomatic / Axiomaitze

Merriam-Webster: www.merriam-webster.com

AXIOM:
a statement accepted as true as the basis for argument or inference
Postulate

AXIOMATIC:
based on or involving an axiom or system of axioms

AXIOMATIZATION:
the act or process of reducing to a system of axioms

Axiom / Axiomatic / Axiomaitze

Merriam-Webster: www.merriam-webster.com

AXIOM:

a statement accepted as true as the basis for argument or inference

Postulate

AXIOMATIC:

based on or involving an axiom or system of axioms

AXIOMATIZATION:

the act or process of reducing to a system of axioms

Axiom / Axiomatic / Axiomaitze

Merriam-Webster: www.merriam-webster.com

AXIOM:

a statement accepted as true as the basis for argument or inference

Postulate

AXIOMATIC:

based on or involving an axiom or system of axioms

AXIOMATIZATION:

the act or process of reducing to a system of axioms

Axiom / Axiomatic / Axiomaitze

Merriam-Webster: www.merriam-webster.com

AXIOM:

a statement accepted as true as the basis for argument or inference

Postulate

AXIOMATIC:

based on or involving an axiom or system of axioms

AXIOMATIZATION:

the act or process of reducing to a system of axioms

SAEED SALEHI, **Logic and Computation, & their interactions**, Arak 2019.

Axiom / Axiomatic / Axiomaitze

Oxford: www.oxforddictionaries.com

AXIOM:

a statement or proposition which is regarded as being established, accepted, or self-evidently true *the axiom that sport builds character*

Math: a statement or proposition on which an abstractly defined structure is based

Origin: late 15th century: from French axiome or Latin axioma, from Greek axio-ma 'what is thought fitting', from axios 'worthy'

AXIOMATIC: self-evident or unquestionable

it is axiomatic that good athletes have a strong mental attitude

Math: relating to or containing axioms

AXIOMATIZE: express (a theory) as a set of axioms

the attempts that are made to axiomatize linguistics

Axiom / Axiomatic / Axiomaitze

Oxford: www.oxforddictionaries.com

AXIOM:

a statement or proposition which is regarded as being established, accepted, or self-evidently true *the axiom that sport builds character*

Math: a statement or proposition on which an abstractly defined structure is based

Origin: late 15th century: from French axiome or Latin axioma, from Greek axio-ma 'what is thought fitting', from axios 'worthy'

AXIOMATIC: self-evident or unquestionable

it is axiomatic that good athletes have a strong mental attitude

Math: relating to or containing axioms

AXIOMATIZE: express (a theory) as a set of axioms

the attempts that are made to axiomatize linguistics

Axiom / Axiomatic / Axiomaitze

Oxford: www.oxforddictionaries.com

AXIOM:

a statement or proposition which is regarded as being established, accepted, or self-evidently true *the axiom that sport builds character*

Math: a statement or proposition on which an abstractly defined structure is based

Origin: late 15th century: from French axiome or Latin axioma, from Greek axio-ma 'what is thought fitting', from axios 'worthy'

AXIOMATIC: self-evident or unquestionable

it is axiomatic that good athletes have a strong mental attitude

Math: relating to or containing axioms

AXIOMATIZE: express (a theory) as a set of axioms

the attempts that are made to axiomatize linguistics

Axiom / Axiomatic / Axiomaitze

Oxford: www.oxforddictionaries.com

AXIOM:

a statement or proposition which is regarded as being established, accepted, or self-evidently true *the axiom that sport builds character*

Math: a statement or proposition on which an abstractly defined structure is based

Origin: late 15th century: from French axiome or Latin axioma, from Greek axio-ma 'what is thought fitting', from axios 'worthy'

AXIOMATIC: self-evident or unquestionable

it is axiomatic that good athletes have a strong mental attitude

Math: relating to or containing axioms

AXIOMATIZE: express (a theory) as a set of axioms

the attempts that are made to axiomatize linguistics

Axiom / Axiomatic / Axiomaitze

Oxford: www.oxforddictionaries.com

AXIOM:

a statement or proposition which is regarded as being established, accepted, or self-evidently true *the axiom that sport builds character*

Math: a statement or proposition on which an abstractly defined structure is based

Origin: late 15th century: from French axiome or Latin axioma, from Greek axio-ma 'what is thought fitting', from axios 'worthy'

AXIOMATIC: self-evident or unquestionable

it is axiomatic that good athletes have a strong mental attitude

Math: relating to or containing axioms

AXIOMATIZE: express (a theory) as a set of axioms

the attempts that are made to axiomatize linguistics

Axiom / Axiomatic / Axiomaitze

Oxford: www.oxforddictionaries.com

AXIOM:

a statement or proposition which is regarded as being established, accepted, or self-evidently true *the axiom that sport builds character*

Math: a statement or proposition on which an abstractly defined structure is based

Origin: late 15th century: from French axiome or Latin axioma, from Greek axio-ma 'what is thought fitting', from axios 'worthy'

AXIOMATIC: self-evident or unquestionable

it is axiomatic that good athletes have a strong mental attitude

Math: relating to or containing axioms

AXIOMATIZE: express (a theory) as a set of axioms

the attempts that are made to axiomatize linguistics

Axiom / Axiomatic / Axiomaitze

Oxford: www.oxforddictionaries.com

AXIOM:

a statement or proposition which is regarded as being established, accepted, or self-evidently true *the axiom that sport builds character*

Math: a statement or proposition on which an abstractly defined structure is based

Origin: late 15th century: from French axiome or Latin axioma, from Greek axio-ma 'what is thought fitting', from axios 'worthy'

AXIOMATIC: self-evident or unquestionable

it is axiomatic that good athletes have a strong mental attitude

Math: relating to or containing axioms

AXIOMATIZE: express (a theory) as a set of axioms

the attempts that are made to axiomatize linguistics

Algebraic Axiomatizing "The Laws of Thought"

Language: $\bot, \top$ $\neg$ $\wedge, \vee$ $\equiv$

Idempotence:	$p \wedge p \equiv p$	$p \vee p \equiv p$
Commutativity:	$p \wedge q \equiv q \wedge p$	$p \vee q \equiv q \vee p$
Associativity:	$p \wedge (q \wedge r) \equiv (p \wedge q) \wedge r$	$p \vee (q \vee r) \equiv (p \vee q) \vee r$
Distributivity:	$p \wedge (q \vee r) \equiv (p \wedge q) \vee (p \wedge r)$	
Distributivity:	$p \vee (q \wedge r) \equiv (p \vee q) \wedge (p \vee r)$	
Tautology:	$p \wedge \top \equiv p$	$p \vee \top \equiv \top$
Contradiction:	$p \wedge \bot \equiv \bot$	$p \vee \bot \equiv p$
Negation:	$p \wedge (\neg p) \equiv \bot$	$p \vee (\neg p) \equiv \top$
Negation:		$\neg(\neg p) \equiv p$
DeMorgan:	$\neg(p \wedge q) \equiv (\neg p) \vee (\neg q)$	$\neg(p \vee q) \equiv (\neg p) \wedge (\neg q)$

Algebraic Axiomatizing "The Laws of Thought"

Language: $\bot, \top$ $\neg$ $\wedge, \vee$ $\equiv$

Idempotence:	$p \wedge p \equiv p$	$p \vee p \equiv p$
Commutativity:	$p \wedge q \equiv q \wedge p$	$p \vee q \equiv q \vee p$
Associativity:	$p \wedge (q \wedge r) \equiv (p \wedge q) \wedge r$	$p \vee (q \vee r) \equiv (p \vee q) \vee r$
Distributivity:	$p \wedge (q \vee r) \equiv (p \wedge q) \vee (p \wedge r)$	
Distributivity:	$p \vee (q \wedge r) \equiv (p \vee q) \wedge (p \vee r)$	
Tautology:	$p \wedge \top \equiv p$	$p \vee \top \equiv \top$
Contradiction:	$p \wedge \bot \equiv \bot$	$p \vee \bot \equiv p$
Negation:	$p \wedge (\neg p) \equiv \bot$	$p \vee (\neg p) \equiv \top$
Negation:		$\neg(\neg p) \equiv p$
DeMorgan:	$\neg(p \wedge q) \equiv (\neg p) \vee (\neg q)$	$\neg(p \vee q) \equiv (\neg p) \wedge (\neg q)$

Algebraic Axiomatizing "The Laws of Thought"

Language: $\bot, \top$ $\neg$ $\wedge, \vee$ $\equiv$

Idempotence:	$p \wedge p \equiv p$	$p \vee p \equiv p$
Commutativity:	$p \wedge q \equiv q \wedge p$	$p \vee q \equiv q \vee p$
Associativity:	$p \wedge (q \wedge r) \equiv (p \wedge q) \wedge r$	$p \vee (q \vee r) \equiv (p \vee q) \vee r$
Distributivity:	$p \wedge (q \vee r) \equiv (p \wedge q) \vee (p \wedge r)$	
Distributivity:	$p \vee (q \wedge r) \equiv (p \vee q) \wedge (p \vee r)$	
Tautology:	$p \wedge \top \equiv p$	$p \vee \top \equiv \top$
Contradiction:	$p \wedge \bot \equiv \bot$	$p \vee \bot \equiv p$
Negation:	$p \wedge (\neg p) \equiv \bot$	$p \vee (\neg p) \equiv \top$
Negation:		$\neg(\neg p) \equiv p$
DeMorgan:	$\neg(p \wedge q) \equiv (\neg p) \vee (\neg q)$	$\neg(p \vee q) \equiv (\neg p) \wedge (\neg q)$

Algebraic Axiomatizing "The Laws of Thought"

Language: $\bot, \top$ $\neg$ $\wedge, \vee$ $\equiv$

Idempotence:	$p \wedge p \equiv p$	$p \vee p \equiv p$
Commutativity:	$p \wedge q \equiv q \wedge p$	$p \vee q \equiv q \vee p$
Associativity:	$p \wedge (q \wedge r) \equiv (p \wedge q) \wedge r$	$p \vee (q \vee r) \equiv (p \vee q) \vee r$
Distributivity:	$p \wedge (q \vee r) \equiv (p \wedge q) \vee (p \wedge r)$	
Distributivity:	$p \vee (q \wedge r) \equiv (p \vee q) \wedge (p \vee r)$	
Tautology:	$p \wedge \top \equiv p$	$p \vee \top \equiv \top$
Contradiction:	$p \wedge \bot \equiv \bot$	$p \vee \bot \equiv p$
Negation:	$p \wedge (\neg p) \equiv \bot$	$p \vee (\neg p) \equiv \top$
Negation:		$\neg(\neg p) \equiv p$
DeMorgan:	$\neg(p \wedge q) \equiv (\neg p) \vee (\neg q)$	$\neg(p \vee q) \equiv (\neg p) \wedge (\neg q)$

Algebraic Axiomatizing "The Laws of Thought"

Language: $\bot, \top$ $\quad \neg$ $\quad \wedge, \vee$ $\quad \equiv$

Idempotence:	$p \wedge p \equiv p$	$p \vee p \equiv p$
Commutativity:	$p \wedge q \equiv q \wedge p$	$p \vee q \equiv q \vee p$
Associativity:	$p \wedge (q \wedge r) \equiv (p \wedge q) \wedge r$	$p \vee (q \vee r) \equiv (p \vee q) \vee r$
Distributivity:	$p \wedge (q \vee r) \equiv (p \wedge q) \vee (p \wedge r)$	
Distributivity:	$p \vee (q \wedge r) \equiv (p \vee q) \wedge (p \vee r)$	
Tautology:	$p \wedge \top \equiv p$	$p \vee \top \equiv \top$
Contradiction:	$p \wedge \bot \equiv \bot$	$p \vee \bot \equiv p$
Negation:	$p \wedge (\neg p) \equiv \bot$	$p \vee (\neg p) \equiv \top$
Negation:		$\neg(\neg p) \equiv p$
DeMorgan:	$\neg(p \wedge q) \equiv (\neg p) \vee (\neg q)$	$\neg(p \vee q) \equiv (\neg p) \wedge (\neg q)$

Algebraic Axiomatizing "The Laws of Thought"

Language: $\bot, \top$ $\neg$ $\wedge, \vee$ $\equiv$

Idempotence:	$p \wedge p \equiv p$	$p \vee p \equiv p$
Commutativity:	$p \wedge q \equiv q \wedge p$	$p \vee q \equiv q \vee p$
Associativity:	$p \wedge (q \wedge r) \equiv (p \wedge q) \wedge r$	$p \vee (q \vee r) \equiv (p \vee q) \vee r$
Distributivity:	$p \wedge (q \vee r) \equiv (p \wedge q) \vee (p \wedge r)$	
Distributivity:	$p \vee (q \wedge r) \equiv (p \vee q) \wedge (p \vee r)$	
Tautology:	$p \wedge \top \equiv p$	$p \vee \top \equiv \top$
Contradiction:	$p \wedge \bot \equiv \bot$	$p \vee \bot \equiv p$
Negation:	$p \wedge (\neg p) \equiv \bot$	$p \vee (\neg p) \equiv \top$
Negation:		$\neg(\neg p) \equiv p$
DeMorgan:	$\neg(p \wedge q) \equiv (\neg p) \vee (\neg q)$	$\neg(p \vee q) \equiv (\neg p) \wedge (\neg q)$

Algebraic Axiomatizing "The Laws of Thought"

Language: $\bot, \top$ $\neg$ $\wedge, \vee$ $\equiv$

Idempotence:	$p \wedge p \equiv p$	$p \vee p \equiv p$
Commutativity:	$p \wedge q \equiv q \wedge p$	$p \vee q \equiv q \vee p$
Associativity:	$p \wedge (q \wedge r) \equiv (p \wedge q) \wedge r$	$p \vee (q \vee r) \equiv (p \vee q) \vee r$
Distributivity:	$p \wedge (q \vee r) \equiv (p \wedge q) \vee (p \wedge r)$	
Distributivity:	$p \vee (q \wedge r) \equiv (p \vee q) \wedge (p \vee r)$	
Tautology:	$p \wedge \top \equiv p$	$p \vee \top \equiv \top$
Contradiction:	$p \wedge \bot \equiv \bot$	$p \vee \bot \equiv p$
Negation:	$p \wedge (\neg p) \equiv \bot$	$p \vee (\neg p) \equiv \top$
Negation:		$\neg(\neg p) \equiv p$
DeMorgan:	$\neg(p \wedge q) \equiv (\neg p) \vee (\neg q)$	$\neg(p \vee q) \equiv (\neg p) \wedge (\neg q)$

Algebraic Axiomatizing "The Laws of Thought"

Language: $\bot, \top$ $\neg$ $\wedge, \vee$ $\equiv$

Idempotence:	$p \wedge p \equiv p$	$p \vee p \equiv p$
Commutativity:	$p \wedge q \equiv q \wedge p$	$p \vee q \equiv q \vee p$
Associativity:	$p \wedge (q \wedge r) \equiv (p \wedge q) \wedge r$	$p \vee (q \vee r) \equiv (p \vee q) \vee r$
Distributivity:	$p \wedge (q \vee r) \equiv (p \wedge q) \vee (p \wedge r)$	
Distributivity:	$p \vee (q \wedge r) \equiv (p \vee q) \wedge (p \vee r)$	
Tautology:	$p \wedge \top \equiv p$	$p \vee \top \equiv \top$
Contradiction:	$p \wedge \bot \equiv \bot$	$p \vee \bot \equiv p$
Negation:	$p \wedge (\neg p) \equiv \bot$	$p \vee (\neg p) \equiv \top$
Negation:		$\neg(\neg p) \equiv p$
DeMorgan:	$\neg(p \wedge q) \equiv (\neg p) \vee (\neg q)$	$\neg(p \vee q) \equiv (\neg p) \wedge (\neg q)$

Algebraic Axiomatizing "The Laws of Thought"

Language: $\bot, \top \quad \neg \quad \wedge, \vee \quad \equiv$

Idempotence:	$p \wedge p \equiv p$	$p \vee p \equiv p$
Commutativity:	$p \wedge q \equiv q \wedge p$	$p \vee q \equiv q \vee p$
Associativity:	$p \wedge (q \wedge r) \equiv (p \wedge q) \wedge r$	$p \vee (q \vee r) \equiv (p \vee q) \vee r$
Distributivity:	$p \wedge (q \vee r) \equiv (p \wedge q) \vee (p \wedge r)$	
Distributivity:	$p \vee (q \wedge r) \equiv (p \vee q) \wedge (p \vee r)$	
Tautology:	$p \wedge \top \equiv p$	$p \vee \top \equiv \top$
Contradiction:	$p \wedge \bot \equiv \bot$	$p \vee \bot \equiv p$
Negation:	$p \wedge (\neg p) \equiv \bot$	$p \vee (\neg p) \equiv \top$
Negation:		$\neg(\neg p) \equiv p$
DeMorgan:	$\neg(p \wedge q) \equiv (\neg p) \vee (\neg q)$	$\neg(p \vee q) \equiv (\neg p) \wedge (\neg q)$

Algebraic Axiomatizing "The Laws of Thought"

Language: $\bot, \top \qquad \neg \qquad \wedge, \vee \qquad \equiv$

Idempotence:	$p \wedge p \equiv p$	$p \vee p \equiv p$
Commutativity:	$p \wedge q \equiv q \wedge p$	$p \vee q \equiv q \vee p$
Associativity:	$p \wedge (q \wedge r) \equiv (p \wedge q) \wedge r$	$p \vee (q \vee r) \equiv (p \vee q) \vee r$
Distributivity:	$p \wedge (q \vee r) \equiv (p \wedge q) \vee (p \wedge r)$	
Distributivity:	$p \vee (q \wedge r) \equiv (p \vee q) \wedge (p \vee r)$	
Tautology:	$p \wedge \top \equiv p$	$p \vee \top \equiv \top$
Contradiction:	$p \wedge \bot \equiv \bot$	$p \vee \bot \equiv p$
Negation:	$p \wedge (\neg p) \equiv \bot$	$p \vee (\neg p) \equiv \top$
Negation:		$\neg(\neg p) \equiv p$
DeMorgan:	$\neg(p \wedge q) \equiv (\neg p) \vee (\neg q)$	$\neg(p \vee q) \equiv (\neg p) \wedge (\neg q)$

Algebraic Axiomatizing "The Laws of Thought"

Language: $\bot, \top$ $\neg$ $\wedge, \vee$ $\equiv$

Idempotence:	$p \wedge p \equiv p$	$p \vee p \equiv p$
Commutativity:	$p \wedge q \equiv q \wedge p$	$p \vee q \equiv q \vee p$
Associativity:	$p \wedge (q \wedge r) \equiv (p \wedge q) \wedge r$	$p \vee (q \vee r) \equiv (p \vee q) \vee r$
Distributivity:	$p \wedge (q \vee r) \equiv (p \wedge q) \vee (p \wedge r)$	
Distributivity:	$p \vee (q \wedge r) \equiv (p \vee q) \wedge (p \vee r)$	
Tautology:	$p \wedge \top \equiv p$	$p \vee \top \equiv \top$
Contradiction:	$p \wedge \bot \equiv \bot$	$p \vee \bot \equiv p$
Negation:	$p \wedge (\neg p) \equiv \bot$	$p \vee (\neg p) \equiv \top$
Negation:		$\neg(\neg p) \equiv p$
DeMorgan:	$\neg(p \wedge q) \equiv (\neg p) \vee (\neg q)$	$\neg(p \vee q) \equiv (\neg p) \wedge (\neg q)$

Algebraic Axiomatizing "The Laws of Thought"

Language: $\bot, \top \qquad \neg \qquad \wedge, \vee \qquad \equiv$

Idempotence:	$p \wedge p \equiv p$	$p \vee p \equiv p$
Commutativity:	$p \wedge q \equiv q \wedge p$	$p \vee q \equiv q \vee p$
Associativity:	$p \wedge (q \wedge r) \equiv (p \wedge q) \wedge r$	$p \vee (q \vee r) \equiv (p \vee q) \vee r$
Distributivity:	$p \wedge (q \vee r) \equiv (p \wedge q) \vee (p \wedge r)$	
Distributivity:	$p \vee (q \wedge r) \equiv (p \vee q) \wedge (p \vee r)$	
Tautology:	$p \wedge \top \equiv p$	$p \vee \top \equiv \top$
Contradiction:	$p \wedge \bot \equiv \bot$	$p \vee \bot \equiv p$
Negation:	$p \wedge (\neg p) \equiv \bot$	$p \vee (\neg p) \equiv \top$
Negation:		$\neg(\neg p) \equiv p$
DeMorgan:	$\neg(p \wedge q) \equiv (\neg p) \vee (\neg q)$	$\neg(p \vee q) \equiv (\neg p) \wedge (\neg q)$

Computing the PROOF!

$$\varphi = (\neg B \rightarrow F) \wedge (B \wedge F \rightarrow \neg I) \wedge (I \vee \neg B \rightarrow \neg F)$$

$$\equiv (B \vee F) \wedge (\neg B \neg \vee F \vee \neg I) \wedge ([B \wedge \neg I] \vee \neg F)$$

$$\equiv (B \vee F) \wedge (\neg B \vee \neg F \vee \neg I) \wedge (B \vee \neg F) \wedge (\neg I \vee \neg F)$$

$$\equiv (B \vee F) \wedge (B \vee \neg F) \wedge (\neg B \vee \neg I \vee \neg F) \wedge (\neg I \vee \neg F)$$

$$\equiv (B \vee [F \wedge \neg F]) \quad \wedge \quad (\neg I \vee \neg F)$$

$$\equiv B \wedge \neg (I \wedge F)$$

https://www.wolframalpha.com/

Computing the PROOF!

$$\varphi = (\neg B \to F) \wedge (B \wedge F \to \neg I) \wedge (I \vee \neg B \to \neg F)$$

$$\equiv (B \vee F) \wedge (\neg B \neg \vee F \vee \neg I) \wedge ([B \wedge \neg I] \vee \neg F)$$

$$\equiv (B \vee F) \wedge (\neg B \vee \neg F \vee \neg I) \wedge (B \vee \neg F) \wedge (\neg I \vee \neg F)$$

$$\equiv (B \vee F) \wedge (B \vee \neg F) \wedge (\neg B \vee \neg I \vee \neg F) \wedge (\neg I \vee \neg F)$$

$$\equiv (B \vee [F \wedge \neg F]) \qquad \wedge \qquad (\neg I \vee \neg F)$$

$$\equiv B \wedge \neg (I \wedge F)$$

https://www.wolframalpha.com/

Computing the PROOF!

$$\varphi = (\neg B \to F) \wedge (B \wedge F \to \neg I) \wedge (I \vee \neg B \to \neg F)$$

$$\equiv (B \vee F) \wedge (\neg B \neg \vee F \vee \neg I) \wedge ([B \wedge \neg I] \vee \neg F)$$

$$\equiv (B \vee F) \wedge (\neg B \vee \neg F \vee \neg I) \wedge (B \vee \neg F) \wedge (\neg I \vee \neg F)$$

$$\equiv (B \vee F) \wedge (B \vee \neg F) \wedge (\neg B \vee \neg I \vee \neg F) \wedge (\neg I \vee \neg F)$$

$$\equiv (B \vee [F \wedge \neg F]) \quad \wedge \quad (\neg I \vee \neg F)$$

$$\equiv B \wedge \neg (I \wedge F)$$

https://www.wolframalpha.com/

Computing the PROOF!

$$\varphi = (\neg B \to F) \wedge (B \wedge F \to \neg I) \wedge (I \vee \neg B \to \neg F)$$

$$\equiv (B \vee F) \wedge (\neg B \neg \vee F \vee \neg I) \wedge ([B \wedge \neg I] \vee \neg F)$$

$$\equiv (B \vee F) \wedge (\neg B \vee \neg F \vee \neg I) \wedge (B \vee \neg F) \wedge (\neg I \vee \neg F)$$

$$\equiv (B \vee F) \wedge (B \vee \neg F) \wedge (\neg B \vee \neg I \vee \neg F) \wedge (\neg I \vee \neg F)$$

$$\equiv (B \vee [F \wedge \neg F]) \quad \wedge \quad (\neg I \vee \neg F)$$

$$\equiv B \wedge \neg(I \wedge F)$$

https://www.wolframalpha.com/

Computing the PROOF!

$$\varphi = (\neg B \to F) \wedge (B \wedge F \to \neg I) \wedge (I \vee \neg B \to \neg F)$$

$$\equiv (B \vee F) \wedge (\neg B \neg \vee F \vee \neg I) \wedge ([B \wedge \neg I] \vee \neg F)$$

$$\equiv (B \vee F) \wedge (\neg B \vee \neg F \vee \neg I) \wedge (B \vee \neg F) \wedge (\neg I \vee \neg F)$$

$$\equiv (B \vee F) \wedge (B \vee \neg F) \wedge (\neg B \vee \neg I \vee \neg F) \wedge (\neg I \vee \neg F)$$

$$\equiv (B \vee [F \wedge \neg F]) \quad \wedge \quad (\neg I \vee \neg F)$$

$$\equiv B \wedge \neg(I \wedge F)$$

https://www.wolframalpha.com/

Computing the PROOF!

$$\varphi = (\neg B \to F) \wedge (B \wedge F \to \neg I) \wedge (I \vee \neg B \to \neg F)$$

$$\equiv (B \vee F) \wedge (\neg B \neg \vee F \vee \neg I) \wedge ([B \wedge \neg I] \vee \neg F)$$

$$\equiv (B \vee F) \wedge (\neg B \vee \neg F \vee \neg I) \wedge (B \vee \neg F) \wedge (\neg I \vee \neg F)$$

$$\equiv (B \vee F) \wedge (B \vee \neg F) \wedge (\neg B \vee \neg I \vee \neg F) \wedge (\neg I \vee \neg F)$$

$$\equiv (B \vee [F \wedge \neg F]) \quad \wedge \quad (\neg I \vee \neg F)$$

$$\equiv B \wedge \neg(I \wedge F)$$

https://www.wolframalpha.com/

Computing the PROOF!

$$\varphi = (\neg B \to F) \wedge (B \wedge F \to \neg I) \wedge (I \vee \neg B \to \neg F)$$

$$\equiv (B \vee F) \wedge (\neg B \neg \vee F \vee \neg I) \wedge ([B \wedge \neg I] \vee \neg F)$$

$$\equiv (B \vee F) \wedge (\neg B \vee \neg F \vee \neg I) \wedge (B \vee \neg F) \wedge (\neg I \vee \neg F)$$

$$\equiv (B \vee F) \wedge (B \vee \neg F) \wedge (\neg B \vee \neg I \vee \neg F) \wedge (\neg I \vee \neg F)$$

$$\equiv (B \vee [F \wedge \neg F]) \quad \wedge \quad (\neg I \vee \neg F)$$

$$\equiv B \wedge \neg(I \wedge F)$$

https://www.wolframalpha.com/

Computing the PROOF!

$$\varphi = (\neg B \to F) \wedge (B \wedge F \to \neg I) \wedge (I \vee \neg B \to \neg F)$$

$$\equiv (B \vee F) \wedge (\neg B \neg \vee F \vee \neg I) \wedge ([B \wedge \neg I] \vee \neg F)$$

$$\equiv (B \vee F) \wedge (\neg B \vee \neg F \vee \neg I) \wedge (B \vee \neg F) \wedge (\neg I \vee \neg F)$$

$$\equiv (B \vee F) \wedge (B \vee \neg F) \wedge (\neg B \vee \neg I \vee \neg F) \wedge (\neg I \vee \neg F)$$

$$\equiv (B \vee [F \wedge \neg F]) \qquad \wedge \qquad (\neg I \vee \neg F)$$

$$\equiv B \wedge \neg (I \wedge F)$$

https://www.wolframalpha.com/

Axiomatizing Propositional Logic

AX$_1$ $\alpha \to (\beta \to \alpha)$

AX$_2$ $[\alpha \to (\beta \to \gamma)] \to [(\alpha \to \beta) \to (\alpha \to \gamma)]$

AX$_3$ $(\neg\beta \to \neg\alpha) \to (\alpha \to \beta)$

RUL $\dfrac{\alpha, \quad \alpha \to \beta}{\beta}$

Some Theorems (EXERCISES):

$\alpha \to \alpha$

$(\neg\beta) \to (\beta \to \alpha)$

$(\alpha \to \beta) \to (\neg\beta \to \neg\alpha)$

$[(\alpha \to \beta) \to \alpha] \to \alpha$

Axiomatizing Propositional Logic

AX_1 $\alpha \to (\beta \to \alpha)$

AX_2 $[\alpha \to (\beta \to \gamma)] \to [(\alpha \to \beta) \to (\alpha \to \gamma)]$

AX_3 $(\neg\beta \to \neg\alpha) \to (\alpha \to \beta)$

RUL $\dfrac{\alpha, \quad \alpha \to \beta}{\beta}$

Some Theorems (EXERCISES):

$\alpha \to \alpha$

$(\neg\beta) \to (\beta \to \alpha)$

$(\alpha \to \beta) \to (\neg\beta \to \neg\alpha)$

$[(\alpha \to \beta) \to \alpha] \to \alpha$

Axiomatizing Propositional Logic

AX$_1$ $\alpha \to (\beta \to \alpha)$

AX$_2$ $[\alpha \to (\beta \to \gamma)] \to [(\alpha \to \beta) \to (\alpha \to \gamma)]$

AX$_3$ $(\neg\beta \to \neg\alpha) \to (\alpha \to \beta)$

RUL $\dfrac{\alpha, \quad \alpha \to \beta}{\beta}$

Some Theorems (EXERCISES):

$\alpha \to \alpha$

$(\neg\beta) \to (\beta \to \alpha)$

$(\alpha \to \beta) \to (\neg\beta \to \neg\alpha)$

$[(\alpha \to \beta) \to \alpha] \to \alpha$

Axiomatizing Propositional Logic

AX_1 $\alpha \to (\beta \to \alpha)$

AX_2 $[\alpha \to (\beta \to \gamma)] \to [(\alpha \to \beta) \to (\alpha \to \gamma)]$

AX_3 $(\neg\beta \to \neg\alpha) \to (\alpha \to \beta)$

RUL $\dfrac{\alpha, \quad \alpha \to \beta}{\beta}$

Some Theorems (EXERCISES):

$\alpha \to \alpha$

$(\neg\beta) \to (\beta \to \alpha)$

$(\alpha \to \beta) \to (\neg\beta \to \neg\alpha)$

$[(\alpha \to \beta) \to \alpha] \to \alpha$

Predicate Logic

- Quantifiers $\forall$, $\exists$
- A Language of Undefined Relations or Functions
  (or Constants)
- More Complex Propositions and Models
  (Complicated Algebraic Structures)

Predicate Logic

- Quantifiers $\forall$, $\exists$

- A Language of Undefined Relations or Functions
  (or Constants)

- More Complex Propositions and Models
  (Complicated Algebraic Structures)

Predicate Logic

- Quantifiers $\forall, \ \exists$
- A Language of Undefined Relations or Functions
  (or Constants)
- More Complex Propositions and Models
  (Complicated Algebraic Structures)

Axiomatizing Predicate Logic

Gödel's Completeness Theorem (1929)

From An Axiomatization of (Logically) Valid Formulas:

- $\alpha \to (\beta \to \alpha)$
- $(\neg\beta \to \neg\alpha) \to (\alpha \to \beta)$
- $[\alpha \to (\beta \to \gamma)] \to [(\alpha \to \beta) \to (\alpha \to \gamma)]$
- $\forall x\varphi(x) \to \varphi(t)$
- $\varphi \to \forall x\varphi$ [x is not free in φ]
- $\forall x(\varphi \to \psi) \to (\forall x\varphi \to \forall x\psi)$

With the Modus Ponens Rule:

- $\dfrac{\varphi, \quad \varphi \to \psi}{\psi}$

All the Universally Valid Formulas CAN BE GENERATED.

Axiomatizing Predicate Logic

Gödel's Completeness Theorem (1929)

From An Axiomatization of (Logically) Valid Formulas:

- $\alpha \to (\beta \to \alpha)$
- $(\neg\beta \to \neg\alpha) \to (\alpha \to \beta)$
- $[\alpha \to (\beta \to \gamma)] \to [(\alpha \to \beta) \to (\alpha \to \gamma)]$
- $\forall x \varphi(x) \to \varphi(t)$
- $\varphi \to \forall x \varphi$ [x is not free in φ]
- $\forall x(\varphi \to \psi) \to (\forall x \varphi \to \forall x \psi)$

With the Modus Ponens Rule:

- $\dfrac{\varphi, \quad \varphi \to \psi}{\psi}$

All the Universally Valid Formulas CAN BE GENERATED.

Axiomatizing Predicate Logic

Gödel's Completeness Theorem (1929)

From An Axiomatization of (Logically) Valid Formulas:

- $\alpha \to (\beta \to \alpha)$
- $(\neg\beta \to \neg\alpha) \to (\alpha \to \beta)$
- $[\alpha \to (\beta \to \gamma)] \to [(\alpha \to \beta) \to (\alpha \to \gamma)]$
- $\forall x\varphi(x) \to \varphi(t)$
- $\varphi \to \forall x\varphi$ [x is not free in φ]
- $\forall x(\varphi \to \psi) \to (\forall x\varphi \to \forall x\psi)$

With the Modus Ponens Rule:

- $\dfrac{\varphi, \quad \varphi \to \psi}{\psi}$

All the Universally Valid Formulas CAN BE GENERATED.

Axiomatizing Predicate Logic

Gödel's Completeness Theorem (1929)

From An Axiomatization of (Logically) Valid Formulas:

- $\alpha \to (\beta \to \alpha)$
- $(\neg\beta \to \neg\alpha) \to (\alpha \to \beta)$
- $[\alpha \to (\beta \to \gamma)] \to [(\alpha \to \beta) \to (\alpha \to \gamma)]$
- $\forall x \varphi(x) \to \varphi(t)$
- $\varphi \to \forall x \varphi$ [x is not free in φ]
- $\forall x(\varphi \to \psi) \to (\forall x \varphi \to \forall x \psi)$

With the Modus Ponens Rule:

- $$\frac{\varphi, \quad \varphi \to \psi}{\psi}$$

All the Universally Valid Formulas CAN BE GENERATED.

Axiomatizing Predicate Logic

Gödel's Completeness Theorem (1929)

From An Axiomatization of (Logically) Valid Formulas:

- $\alpha \rightarrow (\beta \rightarrow \alpha)$
- $(\neg\beta \rightarrow \neg\alpha) \rightarrow (\alpha \rightarrow \beta)$
- $[\alpha \rightarrow (\beta \rightarrow \gamma)] \rightarrow [(\alpha \rightarrow \beta) \rightarrow (\alpha \rightarrow \gamma)]$
- $\forall x\varphi(x) \rightarrow \varphi(t)$
- $\varphi \rightarrow \forall x\varphi$ [x is not free in φ]
- $\forall x(\varphi \rightarrow \psi) \rightarrow (\forall x\varphi \rightarrow \forall x\psi)$

With the Modus Ponens Rule:

- $\dfrac{\varphi, \quad \varphi \rightarrow \psi}{\psi}$

All the Universally Valid Formulas CAN BE GENERATED.

Computably Decidable Set

Computably Decidable set A: an algorithm $\mathcal{P}$ decides on any input x whether $x \in A$ (outputs **YES**) or $x \notin A$ (outputs **NO**).

input: $x \in \mathscr{M}$ ⟶ Algorithm ⟶ **output**: $\begin{cases} \texttt{YES} & \text{if } x \in A \\ \texttt{NO} & \text{if } x \notin A \end{cases}$

Algorithm: single–input, Boolean–output (**1,0**)

Propositional Logic is DECIDABLE.

Algorithms: Truth-Tables, Various Deductive Calculi, etc.
Now the question is the speed of algorithms ...

Computably Decidable Set

Computably Decidable set A: an algorithm $\mathcal{P}$ decides on any input x whether $x \in A$ (outputs **YES**) or $x \notin A$ (outputs **NO**).

$$\xrightarrow{\texttt{input}:\quad x\in\mathscr{M}} \boxed{\text{Algorithm}} \xrightarrow{\texttt{output}:} \begin{cases} \texttt{YES} & \text{if } x \in A \\ \texttt{NO} & \text{if } x \notin A \end{cases}$$

Algorithm: single–input, Boolean–output (**1**, **0**)

Propositional Logic is DECIDABLE.

Algorithms: Truth-Tables, Various Deductive Calculi, etc.
Now the question is the speed of algorithms ...

Computably Decidable Set

Computably Decidable set A: an algorithm $\mathcal{P}$ decides on any input x whether $x \in A$ (outputs **YES**) or $x \notin A$ (outputs **NO**).

input: $x \in \mathscr{M}$ → Algorithm → **output**: $\begin{cases} \texttt{YES} & \text{if } x \in A \\ \texttt{NO} & \text{if } x \notin A \end{cases}$

Algorithm: single–input, Boolean–output (**1**, **0**)

Propositional Logic is DECIDABLE.

Algorithms: Truth-Tables, Various Deductive Calculi, etc.

Now the question is the speed of algorithms ...

Computably Decidable Set

Computably Decidable set A: an algorithm $\mathcal{P}$ decides on any input x whether $x \in A$ (outputs **YES**) or $x \notin A$ (outputs **NO**).

$$\xrightarrow{\texttt{input}:\quad x\in\mathscr{M}} \boxed{\text{Algorithm}} \xrightarrow{\texttt{output}:} \begin{cases} \texttt{YES} & \text{if } x \in A \\ \texttt{NO} & \text{if } x \notin A \end{cases}$$

Algorithm: single–input, Boolean–output (**1,0**)

Propositional Logic is DECIDABLE.

Algorithms: Truth-Tables, Various Deductive Calculi, etc.
Now the question is the speed of algorithms ...

Computably Decidable Set

Computably Decidable set A: an algorithm $\mathcal{P}$ decides on any input x whether $x \in A$ (outputs **YES**) or $x \notin A$ (outputs **NO**).

input: $x \in \mathscr{M}$ ⟶ Algorithm ⟶ **output**: $\begin{cases} \texttt{YES} & \text{if } x \in A \\ \texttt{NO} & \text{if } x \notin A \end{cases}$

Algorithm: single–input, Boolean–output (**1**, **0**)

Propositional Logic is DECIDABLE.

Algorithms: Truth-Tables, Various Deductive Calculi, etc.
Now the question is the speed of algorithms ...

SAEED SALEHI, **Logic and Computation, & their interactions**, Arak 2019.

Computably Decidable Set

Computably Decidable set A: an algorithm $\mathcal{P}$ decides on any input x whether $x \in A$ (outputs **YES**) or $x \notin A$ (outputs **NO**).

input: $x \in \mathscr{M}$ → Algorithm → **output**: $\begin{cases} \texttt{YES} & \text{if } x \in A \\ \texttt{NO} & \text{if } x \notin A \end{cases}$

Algorithm: single–input, Boolean–output (**1,0**)

Propositional Logic is DECIDABLE.

Algorithms: Truth-Tables, Various Deductive Calculi, etc.
Now the question is the speed of algorithms ...

Computably Enumerable Set

Computably Enumerable set A: an (input-free) algorithm $\mathcal{P}$ lists all members of A; i.e., $A = \texttt{output}(\mathcal{P})$.

$$\boxed{\text{Algorithm}} \xrightarrow{\texttt{output}:} \{a_0, a_1, a_2, \cdots\} = A$$

Algorithm: input–free, outputs a set.

Predicate Logic is COMPUTABLY ENUMERABLE (GÖDEL 1929).

Predicate Logic is NOT DECIDABLE (CHURCH & TURING 1936).

► A Good Outcome: Introducing Turing Machines
– the grand grandfather of today's modern computers.

Computably Enumerable Set

Computably Enumerable set A: an (input-free) algorithm $\mathcal{P}$ lists all members of A; i.e., $A = \texttt{output}(\mathcal{P})$.

$$\boxed{\text{Algorithm}} \xrightarrow{\texttt{output:}} \{a_0, a_1, a_2, \cdots\} = A$$

Algorithm: input–free, outputs a set.

Predicate Logic is COMPUTABLY ENUMERABLE (GÖDEL 1929).

Predicate Logic is NOT DECIDABLE (CHURCH & TURING 1936).

▶ A Good Outcome: Introducing Turing Machines
– the grand grandfather of today's modern computers.

Computably Enumerable Set

Computably Enumerable set A: an (input-free) algorithm $\mathcal{P}$ lists all members of A; i.e., $A = \texttt{output}(\mathcal{P})$.

$$\boxed{\text{Algorithm}} \xrightarrow{\texttt{output}:} \{a_0, a_1, a_2, \cdots\} = A$$

Algorithm: input–free, outputs a set.

Predicate Logic is COMPUTABLY ENUMERABLE (GÖDEL 1929).

Predicate Logic is NOT DECIDABLE (CHURCH & TURING 1936).

► A Good Outcome: Introducing Turing Machines
– the grand grandfather of today's modern computers.

Computably Enumerable Set

Computably Enumerable set A: an (input-free) algorithm $\mathcal{P}$ lists all members of A; i.e., $A = \mathtt{output}(\mathcal{P})$.

$$\boxed{\text{Algorithm}} \xrightarrow{\mathtt{output}:} \{a_0, a_1, a_2, \cdots\} = A$$

Algorithm: input–free, outputs a set.

Predicate Logic is COMPUTABLY ENUMERABLE (GÖDEL 1929).

Predicate Logic is NOT DECIDABLE (CHURCH & TURING 1936).

► A Good Outcome: Introducing Turing Machines
– the grand grandfather of today's modern computers.

Computably Enumerable Set

Computably Enumerable set A: an (input-free) algorithm $\mathcal{P}$ lists all members of A; i.e., $A = \texttt{output}(\mathcal{P})$.

$$\boxed{\text{Algorithm}} \xrightarrow{\texttt{output}:} \{a_0, a_1, a_2, \cdots\} = A$$

Algorithm: input–free, outputs a set.

Predicate Logic is COMPUTABLY ENUMERABLE (GÖDEL 1929).

Predicate Logic is NOT DECIDABLE (CHURCH & TURING 1936).

► A Good Outcome: Introducing Turing Machines
– the grand grandfather of today's modern computers.

Computably Enumerable Set

Computably Enumerable set A: an (input-free) algorithm $\mathcal{P}$ lists all members of A; i.e., $A = \texttt{output}(\mathcal{P})$.

$$\boxed{\text{Algorithm}} \xrightarrow{\texttt{output}:} \{a_0, a_1, a_2, \cdots\} = A$$

Algorithm: input–free, outputs a set.

Predicate Logic is COMPUTABLY ENUMERABLE (GÖDEL 1929).

Predicate Logic is NOT DECIDABLE (CHURCH & TURING 1936).

► A Good Outcome: Introducing Turing Machines
– the grand grandfather of today's modern computers.

Computably Enumerable Set

Computably Enumerable set A: an (input-free) algorithm $\mathcal{P}$ lists all members of A; i.e., $A = \texttt{output}(\mathcal{P})$.

$$\boxed{\text{Algorithm}} \xrightarrow{\texttt{output}:} \{a_0, a_1, a_2, \cdots\} = A$$

Algorithm: input–free, outputs a set.

Predicate Logic is COMPUTABLY ENUMERABLE (GÖDEL 1929).

Predicate Logic is NOT DECIDABLE (CHURCH & TURING 1936).

► A Good Outcome: Introducing Turing Machines
– the grand grandfather of today's modern computers.

Decision Problem, again

Decision Problem for the Structure $(\mathfrak{M}, \mathcal{L})$:

Input: A First–Order Sentence φ in the Language $\mathcal{L}$.
Output: *YES* (if $\mathfrak{M} \models \varphi$) *NO* (if $\mathfrak{M} \not\models \varphi$).

Examples:

- $\mathbb{N} \not\models \forall x \exists y (x + y = 0)$ but $\mathbb{Z} \models \forall x \exists y (x + y = 0)$.
- $\mathbb{Z} \not\models \forall x \exists y (x \neq 0 \rightarrow [x \cdot y = 1])$ but $\mathbb{Q} \models \forall x \exists y (x \neq 0 \rightarrow [x \cdot y = 1])$.
- $\mathbb{Q} \not\models \forall x \exists y (0 \leqslant x \rightarrow [y \cdot y = x])$ but $\mathbb{R} \models \forall x \exists y (0 \leqslant x \rightarrow [y \cdot y = x])$.
- $\mathbb{R} \not\models \forall x \exists y (y \cdot y + x = 0)$ but $\mathbb{C} \models \forall x \exists y (y \cdot y + x = 0)$.

Decision Problem, again

Decision Problem for the Structure $(\mathfrak{M}, \mathcal{L})$:

Input: A First–Order Sentence φ in the Language $\mathcal{L}$.

Output: *YES* (if $\mathfrak{M} \models \varphi$) *NO* (if $\mathfrak{M} \not\models \varphi$).

Examples:

- $\mathbb{N} \not\models \forall x \exists y (x + y = 0)$ but $\mathbb{Z} \models \forall x \exists y (x + y = 0)$.
- $\mathbb{Z} \not\models \forall x \exists y (x \neq 0 \rightarrow [x \cdot y = 1])$ but $\mathbb{Q} \models \forall x \exists y (x \neq 0 \rightarrow [x \cdot y = 1])$.
- $\mathbb{Q} \not\models \forall x \exists y (0 \leqslant x \rightarrow [y \cdot y = x])$ but $\mathbb{R} \models \forall x \exists y (0 \leqslant x \rightarrow [y \cdot y = x])$.
- $\mathbb{R} \not\models \forall x \exists y (y \cdot y + x = 0)$ but $\mathbb{C} \models \forall x \exists y (y \cdot y + x = 0)$.

Decision Problem, again

Decision Problem for the Structure $(\mathfrak{M}, \mathcal{L})$:

Input: A First–Order Sentence φ in the Language $\mathcal{L}$.

Output: *YES* (if $\mathfrak{M} \models \varphi$) *NO* (if $\mathfrak{M} \not\models \varphi$).

Examples:

- $\mathbb{N} \not\models \forall x \exists y (x + y = 0)$ but $\mathbb{Z} \models \forall x \exists y (x + y = 0)$.
- $\mathbb{Z} \not\models \forall x \exists y (x \neq 0 \rightarrow [x \cdot y = 1])$ but $\mathbb{Q} \models \forall x \exists y (x \neq 0 \rightarrow [x \cdot y = 1])$.
- $\mathbb{Q} \not\models \forall x \exists y (0 \leqslant x \rightarrow [y \cdot y = x])$ but $\mathbb{R} \models \forall x \exists y (0 \leqslant x \rightarrow [y \cdot y = x])$.
- $\mathbb{R} \not\models \forall x \exists y (y \cdot y + x = 0)$ but $\mathbb{C} \models \forall x \exists y (y \cdot y + x = 0)$.

Decision Problem, again

Decision Problem for the Structure $(\mathfrak{M}, \mathcal{L})$:

Input: A First–Order Sentence φ in the Language $\mathcal{L}$.

Output: *YES* (if $\mathfrak{M} \models \varphi$) *NO* (if $\mathfrak{M} \not\models \varphi$).

Examples:

- $\mathbb{N} \not\models \forall x \exists y (x + y = 0)$ but $\mathbb{Z} \models \forall x \exists y (x + y = 0)$.
- $\mathbb{Z} \not\models \forall x \exists y (x \neq 0 \rightarrow [x \cdot y = 1])$ but $\mathbb{Q} \models \forall x \exists y (x \neq 0 \rightarrow [x \cdot y = 1])$.
- $\mathbb{Q} \not\models \forall x \exists y (0 \leqslant x \rightarrow [y \cdot y = x])$ but $\mathbb{R} \models \forall x \exists y (0 \leqslant x \rightarrow [y \cdot y = x])$.
- $\mathbb{R} \not\models \forall x \exists y (y \cdot y + x = 0)$ but $\mathbb{C} \models \forall x \exists y (y \cdot y + x = 0)$.

Decision Problem, again

Decision Problem for the Structure $(\mathfrak{M}, \mathcal{L})$:

Input: A First–Order Sentence φ in the Language $\mathcal{L}$.
Output: *YES* (if $\mathfrak{M} \models \varphi$) *NO* (if $\mathfrak{M} \not\models \varphi$).

Examples:

- $\mathbb{N} \not\models \forall x \exists y (x + y = 0)$ but $\mathbb{Z} \models \forall x \exists y (x + y = 0)$.
- $\mathbb{Z} \not\models \forall x \exists y (x \neq 0 \rightarrow [x \cdot y = 1])$ but $\mathbb{Q} \models \forall x \exists y (x \neq 0 \rightarrow [x \cdot y = 1])$.
- $\mathbb{Q} \not\models \forall x \exists y (0 \leqslant x \rightarrow [y \cdot y = x])$ but $\mathbb{R} \models \forall x \exists y (0 \leqslant x \rightarrow [y \cdot y = x])$.
- $\mathbb{R} \not\models \forall x \exists y (y \cdot y + x = 0)$ but $\mathbb{C} \models \forall x \exists y (y \cdot y + x = 0)$.

Decision Problem, again

Decision Problem for the Structure $(\mathfrak{M}, \mathcal{L})$:

Input: A First–Order Sentence φ in the Language $\mathcal{L}$.

Output: *YES* (if $\mathfrak{M} \models \varphi$) *NO* (if $\mathfrak{M} \not\models \varphi$).

Examples:

- $\mathbb{N} \not\models \forall x \exists y (x + y = 0)$ but $\mathbb{Z} \models \forall x \exists y (x + y = 0)$.
- $\mathbb{Z} \not\models \forall x \exists y (x \neq 0 \rightarrow [x \cdot y = 1])$ but $\mathbb{Q} \models \forall x \exists y (x \neq 0 \rightarrow [x \cdot y = 1])$.
- $\mathbb{Q} \not\models \forall x \exists y (0 \leqslant x \rightarrow [y \cdot y = x])$ but $\mathbb{R} \models \forall x \exists y (0 \leqslant x \rightarrow [y \cdot y = x])$.
- $\mathbb{R} \not\models \forall x \exists y (y \cdot y + x = 0)$ but $\mathbb{C} \models \forall x \exists y (y \cdot y + x = 0)$.

Decision Problem, again

Decision Problem for the Structure $(\mathfrak{M}, \mathcal{L})$:

Input: A First–Order Sentence φ in the Language $\mathcal{L}$.
Output: *YES* (if $\mathfrak{M} \models \varphi$) *NO* (if $\mathfrak{M} \not\models \varphi$).

Examples:

- $\mathbb{N} \not\models \forall x \exists y (x + y = 0)$ but $\mathbb{Z} \models \forall x \exists y (x + y = 0)$.
- $\mathbb{Z} \not\models \forall x \exists y (x \neq 0 \rightarrow [x \cdot y = 1])$ but $\mathbb{Q} \models \forall x \exists y (x \neq 0 \rightarrow [x \cdot y = 1])$.
- $\mathbb{Q} \not\models \forall x \exists y (0 \leqslant x \rightarrow [y \cdot y = x])$ but $\mathbb{R} \models \forall x \exists y (0 \leqslant x \rightarrow [y \cdot y = x])$.
- $\mathbb{R} \not\models \forall x \exists y (y \cdot y + x = 0)$ but $\mathbb{C} \models \forall x \exists y (y \cdot y + x = 0)$.

Decision Problem, again

Decision Problem for the Structure $(\mathfrak{M}, \mathcal{L})$:

Input: A First–Order Sentence φ in the Language $\mathcal{L}$.
Output: *YES* (if $\mathfrak{M} \models \varphi$) *NO* (if $\mathfrak{M} \not\models \varphi$).

Examples:

- $\mathbb{N} \not\models \forall x \exists y (x + y = 0)$ but $\mathbb{Z} \models \forall x \exists y (x + y = 0)$.
- $\mathbb{Z} \not\models \forall x \exists y (x \neq 0 \rightarrow [x \cdot y = 1])$ but $\mathbb{Q} \models \forall x \exists y (x \neq 0 \rightarrow [x \cdot y = 1])$.
- $\mathbb{Q} \not\models \forall x \exists y (0 \leqslant x \rightarrow [y \cdot y = x])$ but $\mathbb{R} \models \forall x \exists y (0 \leqslant x \rightarrow [y \cdot y = x])$.
- $\mathbb{R} \not\models \forall x \exists y (y \cdot y + x = 0)$ but $\mathbb{C} \models \forall x \exists y (y \cdot y + x = 0)$.

Decidability of Mathematical Structures

The Decidability Problem for the Structures:

	$\mathbb{N}$	$\mathbb{Z}$	$\mathbb{Q}$	$\mathbb{R}$	$\mathbb{C}$
$\{<\}$	$\langle\mathbb{N};<\rangle$	$\langle\mathbb{Z};<\rangle$	$\langle\mathbb{Q};<\rangle$	$\langle\mathbb{R};<\rangle$	—
$\{+\}$	$\langle\mathbb{N};+\rangle$	$\langle\mathbb{Z};+\rangle$	$\langle\mathbb{Q};+\rangle$	$\langle\mathbb{R};+\rangle$	$\langle\mathbb{C};+\rangle$
$\{\cdot\}$	$\langle\mathbb{N};\cdot\rangle$	$\langle\mathbb{Z};\cdot\rangle$	$\langle\mathbb{Q};\cdot\rangle$	$\langle\mathbb{R};\cdot\rangle$	$\langle\mathbb{C};\cdot\rangle$
$\{+,<\}$	$\langle\mathbb{N};+,<\rangle$	$\langle\mathbb{Z};+,<\rangle$	$\langle\mathbb{Q};+,<\rangle$	$\langle\mathbb{R};+,<\rangle$	—
$\{+,\cdot\}$	$\langle\mathbb{N};+,\cdot\rangle$	$\langle\mathbb{Z};+,\cdot\rangle$	$\langle\mathbb{Q};+,\cdot\rangle$	$\langle\mathbb{R};+,\cdot\rangle$	$\langle\mathbb{C};+,\cdot\rangle$
$\{\cdot,<\}$	$\langle\mathbb{N};\cdot,<\rangle$	$\langle\mathbb{Z};\cdot,<\rangle$	$\langle\mathbb{Q};\cdot,<\rangle$	$\langle\mathbb{R};\cdot,<\rangle$	—
$\{+,\cdot,<\}$	$\langle\mathbb{N};+,\cdot,<\rangle$	$\langle\mathbb{Z};+,\cdot,<\rangle$	$\langle\mathbb{Q};+,\cdot,<\rangle$	$\langle\mathbb{R};+,\cdot,<\rangle$	—
E	$\langle\mathbb{N};\exp\rangle$	—	—	$\langle\mathbb{R};+,\cdot,e^x\rangle$	$\langle\mathbb{C};+,\cdot,e^x\rangle$

Decidability of Mathematical Structures

The Decidability Problem for the Structures:

	$\mathbb{N}$	$\mathbb{Z}$	$\mathbb{Q}$	$\mathbb{R}$	$\mathbb{C}$
$\{<\}$	$\langle\mathbb{N};<\rangle$	$\langle\mathbb{Z};<\rangle$	$\langle\mathbb{Q};<\rangle$	$\langle\mathbb{R};<\rangle$	—
$\{+\}$	$\langle\mathbb{N};+\rangle$	$\langle\mathbb{Z};+\rangle$	$\langle\mathbb{Q};+\rangle$	$\langle\mathbb{R};+\rangle$	$\langle\mathbb{C};+\rangle$
$\{\cdot\}$	$\langle\mathbb{N};\cdot\rangle$	$\langle\mathbb{Z};\cdot\rangle$	$\langle\mathbb{Q};\cdot\rangle$	$\langle\mathbb{R};\cdot\rangle$	$\langle\mathbb{C};\cdot\rangle$
$\{+,<\}$	$\langle\mathbb{N};+,<\rangle$	$\langle\mathbb{Z};+,<\rangle$	$\langle\mathbb{Q};+,<\rangle$	$\langle\mathbb{R};+,<\rangle$	—
$\{+,\cdot\}$	$\langle\mathbb{N};+,\cdot\rangle$	$\langle\mathbb{Z};+,\cdot\rangle$	$\langle\mathbb{Q};+,\cdot\rangle$	$\langle\mathbb{R};+,\cdot\rangle$	$\langle\mathbb{C};+,\cdot\rangle$
$\{\cdot,<\}$	$\langle\mathbb{N};\cdot,<\rangle$	$\langle\mathbb{Z};\cdot,<\rangle$	$\langle\mathbb{Q};\cdot,<\rangle$	$\langle\mathbb{R};\cdot,<\rangle$	—
$\{+,\cdot,<\}$	$\langle\mathbb{N};+,\cdot,<\rangle$	$\langle\mathbb{Z};+,\cdot,<\rangle$	$\langle\mathbb{Q};+,\cdot,<\rangle$	$\langle\mathbb{R};+,\cdot,<\rangle$	—
E	$\langle\mathbb{N};\exp\rangle$	—	—	$\langle\mathbb{R};+,\cdot,e^x\rangle$	$\langle\mathbb{C};+,\cdot,e^x\rangle$

Decidability of Mathematical Structures

The Decidability Problem for the Structures:

	$\mathbb{N}$	$\mathbb{Z}$	$\mathbb{Q}$	$\mathbb{R}$	$\mathbb{C}$
$\{<\}$	$\langle\mathbb{N};<\rangle$	$\langle\mathbb{Z};<\rangle$	$\langle\mathbb{Q};<\rangle$	$\langle\mathbb{R};<\rangle$	—
$\{+\}$	$\langle\mathbb{N};+\rangle$	$\langle\mathbb{Z};+\rangle$	$\langle\mathbb{Q};+\rangle$	$\langle\mathbb{R};+\rangle$	$\langle\mathbb{C};+\rangle$
$\{\cdot\}$	$\langle\mathbb{N};\cdot\rangle$	$\langle\mathbb{Z};\cdot\rangle$	$\langle\mathbb{Q};\cdot\rangle$	$\langle\mathbb{R};\cdot\rangle$	$\langle\mathbb{C};\cdot\rangle$
$\{+,<\}$	$\langle\mathbb{N};+,<\rangle$	$\langle\mathbb{Z};+,<\rangle$	$\langle\mathbb{Q};+,<\rangle$	$\langle\mathbb{R};+,<\rangle$	—
$\{+,\cdot\}$	$\langle\mathbb{N};+,\cdot\rangle$	$\langle\mathbb{Z};+,\cdot\rangle$	$\langle\mathbb{Q};+,\cdot\rangle$	$\langle\mathbb{R};+,\cdot\rangle$	$\langle\mathbb{C};+,\cdot\rangle$
$\{\cdot,<\}$	$\langle\mathbb{N};\cdot,<\rangle$	$\langle\mathbb{Z};\cdot,<\rangle$	$\langle\mathbb{Q};\cdot,<\rangle$	$\langle\mathbb{R};\cdot,<\rangle$	—
$\{+,\cdot,<\}$	$\langle\mathbb{N};+,\cdot,<\rangle$	$\langle\mathbb{Z};+,\cdot,<\rangle$	$\langle\mathbb{Q};+,\cdot,<\rangle$	$\langle\mathbb{R};+,\cdot,<\rangle$	—
E	$\langle\mathbb{N};\exp\rangle$	—	—	$\langle\mathbb{R};+,\cdot,e^x\rangle$	$\langle\mathbb{C};+,\cdot,e^x\rangle$

Decidability of Mathematical Structures

The Decidability Problem for the Structures:

	$\mathbb{N}$	$\mathbb{Z}$	$\mathbb{Q}$	$\mathbb{R}$	$\mathbb{C}$
$\{<\}$	$\langle\mathbb{N};<\rangle$	$\langle\mathbb{Z};<\rangle$	$\langle\mathbb{Q};<\rangle$	$\langle\mathbb{R};<\rangle$	–
$\{+\}$	$\langle\mathbb{N};+\rangle$	$\langle\mathbb{Z};+\rangle$	$\langle\mathbb{Q};+\rangle$	$\langle\mathbb{R};+\rangle$	$\langle\mathbb{C};+\rangle$
$\{\cdot\}$	$\langle\mathbb{N};\cdot\rangle$	$\langle\mathbb{Z};\cdot\rangle$	$\langle\mathbb{Q};\cdot\rangle$	$\langle\mathbb{R};\cdot\rangle$	$\langle\mathbb{C};\cdot\rangle$
$\{+,<\}$	$\langle\mathbb{N};+,<\rangle$	$\langle\mathbb{Z};+,<\rangle$	$\langle\mathbb{Q};+,<\rangle$	$\langle\mathbb{R};+,<\rangle$	–
$\{+,\cdot\}$	$\langle\mathbb{N};+,\cdot\rangle$	$\langle\mathbb{Z};+,\cdot\rangle$	$\langle\mathbb{Q};+,\cdot\rangle$	$\langle\mathbb{R};+,\cdot\rangle$	$\langle\mathbb{C};+,\cdot\rangle$
$\{\cdot,<\}$	$\langle\mathbb{N};\cdot,<\rangle$	$\langle\mathbb{Z};\cdot,<\rangle$	$\langle\mathbb{Q};\cdot,<\rangle$	$\langle\mathbb{R};\cdot,<\rangle$	–
$\{+,\cdot,<\}$	$\langle\mathbb{N};+,\cdot,<\rangle$	$\langle\mathbb{Z};+,\cdot,<\rangle$	$\langle\mathbb{Q};+,\cdot,<\rangle$	$\langle\mathbb{R};+,\cdot,<\rangle$	–
E	$\langle\mathbb{N};\exp\rangle$	–	–	$\langle\mathbb{R};+,\cdot,e^x\rangle$	$\langle\mathbb{C};+,\cdot,e^x\rangle$

Decidability of Mathematical Structures

The Decidability Problem for the Structures:

	$\mathbb{N}$	$\mathbb{Z}$	$\mathbb{Q}$	$\mathbb{R}$	$\mathbb{C}$
$\{<\}$	$\langle\mathbb{N};<\rangle$	$\langle\mathbb{Z};<\rangle$	$\langle\mathbb{Q};<\rangle$	$\langle\mathbb{R};<\rangle$	—
$\{+\}$	$\langle\mathbb{N};+\rangle$	$\langle\mathbb{Z};+\rangle$	$\langle\mathbb{Q};+\rangle$	$\langle\mathbb{R};+\rangle$	$\langle\mathbb{C};+\rangle$
$\{\cdot\}$	$\langle\mathbb{N};\cdot\rangle$	$\langle\mathbb{Z};\cdot\rangle$	$\langle\mathbb{Q};\cdot\rangle$	$\langle\mathbb{R};\cdot\rangle$	$\langle\mathbb{C};\cdot\rangle$
$\{+,<\}$	$\langle\mathbb{N};+,<\rangle$	$\langle\mathbb{Z};+,<\rangle$	$\langle\mathbb{Q};+,<\rangle$	$\langle\mathbb{R};+,<\rangle$	—
$\{+,\cdot\}$	$\langle\mathbb{N};+,\cdot\rangle$	$\langle\mathbb{Z};+,\cdot\rangle$	$\langle\mathbb{Q};+,\cdot\rangle$	$\langle\mathbb{R};+,\cdot\rangle$	$\langle\mathbb{C};+,\cdot\rangle$
$\{\cdot,<\}$	$\langle\mathbb{N};\cdot,<\rangle$	$\langle\mathbb{Z};\cdot,<\rangle$	$\langle\mathbb{Q};\cdot,<\rangle$	$\langle\mathbb{R};\cdot,<\rangle$	—
$\{+,\cdot,<\}$	$\langle\mathbb{N};+,\cdot,<\rangle$	$\langle\mathbb{Z};+,\cdot,<\rangle$	$\langle\mathbb{Q};+,\cdot,<\rangle$	$\langle\mathbb{R};+,\cdot,<\rangle$	—
E	$\langle\mathbb{N};\exp\rangle$	—	—	$\langle\mathbb{R};+,\cdot,e^x\rangle$	$\langle\mathbb{C};+,\cdot,e^x\rangle$

Decidability of Mathematical Structures

The Decidability Problem for the Structures:

	$\mathbb{N}$	$\mathbb{Z}$	$\mathbb{Q}$	$\mathbb{R}$	$\mathbb{C}$
$\{<\}$	$\langle\mathbb{N};<\rangle$	$\langle\mathbb{Z};<\rangle$	$\langle\mathbb{Q};<\rangle$	$\langle\mathbb{R};<\rangle$	—
$\{+\}$	$\langle\mathbb{N};+\rangle$	$\langle\mathbb{Z};+\rangle$	$\langle\mathbb{Q};+\rangle$	$\langle\mathbb{R};+\rangle$	$\langle\mathbb{C};+\rangle$
$\{\cdot\}$	$\langle\mathbb{N};\cdot\rangle$	$\langle\mathbb{Z};\cdot\rangle$	$\langle\mathbb{Q};\cdot\rangle$	$\langle\mathbb{R};\cdot\rangle$	$\langle\mathbb{C};\cdot\rangle$
$\{+,<\}$	$\langle\mathbb{N};+,<\rangle$	$\langle\mathbb{Z};+,<\rangle$	$\langle\mathbb{Q};+,<\rangle$	$\langle\mathbb{R};+,<\rangle$	—
$\{+,\cdot\}$	$\langle\mathbb{N};+,\cdot\rangle$	$\langle\mathbb{Z};+,\cdot\rangle$	$\langle\mathbb{Q};+,\cdot\rangle$	$\langle\mathbb{R};+,\cdot\rangle$	$\langle\mathbb{C};+,\cdot\rangle$
$\{\cdot,<\}$	$\langle\mathbb{N};\cdot,<\rangle$	$\langle\mathbb{Z};\cdot,<\rangle$	$\langle\mathbb{Q};\cdot,<\rangle$	$\langle\mathbb{R};\cdot,<\rangle$	—
$\{+,\cdot,<\}$	$\langle\mathbb{N};+,\cdot,<\rangle$	$\langle\mathbb{Z};+,\cdot,<\rangle$	$\langle\mathbb{Q};+,\cdot,<\rangle$	$\langle\mathbb{R};+,\cdot,<\rangle$	—
E	$\langle\mathbb{N};\exp\rangle$	—	—	$\langle\mathbb{R};+,\cdot,e^x\rangle$	$\langle\mathbb{C};+,\cdot,e^x\rangle$

Decidability of Mathematical Structures

The Decidability Problem for the Structures:

	$\mathbb{N}$	$\mathbb{Z}$	$\mathbb{Q}$	$\mathbb{R}$	$\mathbb{C}$
$\{<\}$	$\langle\mathbb{N};<\rangle$	$\langle\mathbb{Z};<\rangle$	$\langle\mathbb{Q};<\rangle$	$\langle\mathbb{R};<\rangle$	—
$\{+\}$	$\langle\mathbb{N};+\rangle$	$\langle\mathbb{Z};+\rangle$	$\langle\mathbb{Q};+\rangle$	$\langle\mathbb{R};+\rangle$	$\langle\mathbb{C};+\rangle$
$\{\cdot\}$	$\langle\mathbb{N};\cdot\rangle$	$\langle\mathbb{Z};\cdot\rangle$	$\langle\mathbb{Q};\cdot\rangle$	$\langle\mathbb{R};\cdot\rangle$	$\langle\mathbb{C};\cdot\rangle$
$\{+,<\}$	$\langle\mathbb{N};+,<\rangle$	$\langle\mathbb{Z};+,<\rangle$	$\langle\mathbb{Q};+,<\rangle$	$\langle\mathbb{R};+,<\rangle$	—
$\{+,\cdot\}$	$\langle\mathbb{N};+,\cdot\rangle$	$\langle\mathbb{Z};+,\cdot\rangle$	$\langle\mathbb{Q};+,\cdot\rangle$	$\langle\mathbb{R};+,\cdot\rangle$	$\langle\mathbb{C};+,\cdot\rangle$
$\{\cdot,<\}$	$\langle\mathbb{N};\cdot,<\rangle$	$\langle\mathbb{Z};\cdot,<\rangle$	$\langle\mathbb{Q};\cdot,<\rangle$	$\langle\mathbb{R};\cdot,<\rangle$	—
$\{+,\cdot,<\}$	$\langle\mathbb{N};+,\cdot,<\rangle$	$\langle\mathbb{Z};+,\cdot,<\rangle$	$\langle\mathbb{Q};+,\cdot,<\rangle$	$\langle\mathbb{R};+,\cdot,<\rangle$	—
E	$\langle\mathbb{N};\exp\rangle$	—	—	$\langle\mathbb{R};+,\cdot,e^x\rangle$	$\langle\mathbb{C};+,\cdot,e^x\rangle$

Decidability of Mathematical Structures

The Decidability Problem for the Structures:

	$\mathbb{N}$	$\mathbb{Z}$	$\mathbb{Q}$	$\mathbb{R}$	$\mathbb{C}$
$\{<\}$	$\langle\mathbb{N};<\rangle$	$\langle\mathbb{Z};<\rangle$	$\langle\mathbb{Q};<\rangle$	$\langle\mathbb{R};<\rangle$	–
$\{+\}$	$\langle\mathbb{N};+\rangle$	$\langle\mathbb{Z};+\rangle$	$\langle\mathbb{Q};+\rangle$	$\langle\mathbb{R};+\rangle$	$\langle\mathbb{C};+\rangle$
$\{\cdot\}$	$\langle\mathbb{N};\cdot\rangle$	$\langle\mathbb{Z};\cdot\rangle$	$\langle\mathbb{Q};\cdot\rangle$	$\langle\mathbb{R};\cdot\rangle$	$\langle\mathbb{C};\cdot\rangle$
$\{+,<\}$	$\langle\mathbb{N};+,<\rangle$	$\langle\mathbb{Z};+,<\rangle$	$\langle\mathbb{Q};+,<\rangle$	$\langle\mathbb{R};+,<\rangle$	–
$\{+,\cdot\}$	$\langle\mathbb{N};+,\cdot\rangle$	$\langle\mathbb{Z};+,\cdot\rangle$	$\langle\mathbb{Q};+,\cdot\rangle$	$\langle\mathbb{R};+,\cdot\rangle$	$\langle\mathbb{C};+,\cdot\rangle$
$\{\cdot,<\}$	$\langle\mathbb{N};\cdot,<\rangle$	$\langle\mathbb{Z};\cdot,<\rangle$	$\langle\mathbb{Q};\cdot,<\rangle$	$\langle\mathbb{R};\cdot,<\rangle$	–
$\{+,\cdot,<\}$	$\langle\mathbb{N};+,\cdot,<\rangle$	$\langle\mathbb{Z};+,\cdot,<\rangle$	$\langle\mathbb{Q};+,\cdot,<\rangle$	$\langle\mathbb{R};+,\cdot,<\rangle$	–
E	$\langle\mathbb{N};\exp\rangle$	–	–	$\langle\mathbb{R};+,\cdot,e^x\rangle$	$\langle\mathbb{C};+,\cdot,e^x\rangle$

Decidability of Mathematical Structures

The Decidability Problem for the Structures:

	$\mathbb{N}$	$\mathbb{Z}$	$\mathbb{Q}$	$\mathbb{R}$	$\mathbb{C}$
$\{<\}$	$\langle\mathbb{N};<\rangle$	$\langle\mathbb{Z};<\rangle$	$\langle\mathbb{Q};<\rangle$	$\langle\mathbb{R};<\rangle$	—
$\{+\}$	$\langle\mathbb{N};+\rangle$	$\langle\mathbb{Z};+\rangle$	$\langle\mathbb{Q};+\rangle$	$\langle\mathbb{R};+\rangle$	$\langle\mathbb{C};+\rangle$
$\{\cdot\}$	$\langle\mathbb{N};\cdot\rangle$	$\langle\mathbb{Z};\cdot\rangle$	$\langle\mathbb{Q};\cdot\rangle$	$\langle\mathbb{R};\cdot\rangle$	$\langle\mathbb{C};\cdot\rangle$
$\{+,<\}$	$\langle\mathbb{N};+,<\rangle$	$\langle\mathbb{Z};+,<\rangle$	$\langle\mathbb{Q};+,<\rangle$	$\langle\mathbb{R};+,<\rangle$	—
$\{+,\cdot\}$	$\langle\mathbb{N};+,\cdot\rangle$	$\langle\mathbb{Z};+,\cdot\rangle$	$\langle\mathbb{Q};+,\cdot\rangle$	$\langle\mathbb{R};+,\cdot\rangle$	$\langle\mathbb{C};+,\cdot\rangle$
$\{\cdot,<\}$	$\langle\mathbb{N};\cdot,<\rangle$	$\langle\mathbb{Z};\cdot,<\rangle$	$\langle\mathbb{Q};\cdot,<\rangle$	$\langle\mathbb{R};\cdot,<\rangle$	—
$\{+,\cdot,<\}$	$\langle\mathbb{N};+,\cdot,<\rangle$	$\langle\mathbb{Z};+,\cdot,<\rangle$	$\langle\mathbb{Q};+,\cdot,<\rangle$	$\langle\mathbb{R};+,\cdot,<\rangle$	—
E	$\langle\mathbb{N};\exp\rangle$	—	—	$\langle\mathbb{R};+,\cdot,e^x\rangle$	$\langle\mathbb{C};+,\cdot,e^x\rangle$

SAEED SALEHI, **Logic and Computation, & their interactions**, Arak 2019.

Decidability of Mathematical Structures

The Decidability Problem for the Structures:

	$\mathbb{N}$	$\mathbb{Z}$	$\mathbb{Q}$	$\mathbb{R}$	$\mathbb{C}$
$\{<\}$	$\langle\mathbb{N};<\rangle$	$\langle\mathbb{Z};<\rangle$	$\langle\mathbb{Q};<\rangle$	$\langle\mathbb{R};<\rangle$	—
$\{+\}$	$\langle\mathbb{N};+\rangle$	$\langle\mathbb{Z};+\rangle$	$\langle\mathbb{Q};+\rangle$	$\langle\mathbb{R};+\rangle$	$\langle\mathbb{C};+\rangle$
$\{\cdot\}$	$\langle\mathbb{N};\cdot\rangle$	$\langle\mathbb{Z};\cdot\rangle$	$\langle\mathbb{Q};\cdot\rangle$	$\langle\mathbb{R};\cdot\rangle$	$\langle\mathbb{C};\cdot\rangle$
$\{+,<\}$	$\langle\mathbb{N};+,<\rangle$	$\langle\mathbb{Z};+,<\rangle$	$\langle\mathbb{Q};+,<\rangle$	$\langle\mathbb{R};+,<\rangle$	—
$\{+,\cdot\}$	$\langle\mathbb{N};+,\cdot\rangle$	$\langle\mathbb{Z};+,\cdot\rangle$	$\langle\mathbb{Q};+,\cdot\rangle$	$\langle\mathbb{R};+,\cdot\rangle$	$\langle\mathbb{C};+,\cdot\rangle$
$\{\cdot,<\}$	$\langle\mathbb{N};\cdot,<\rangle$	$\langle\mathbb{Z};\cdot,<\rangle$	$\langle\mathbb{Q};\cdot,<\rangle$	$\langle\mathbb{R};\cdot,<\rangle$	—
$\{+,\cdot,<\}$	$\langle\mathbb{N};+,\cdot,<\rangle$	$\langle\mathbb{Z};+,\cdot,<\rangle$	$\langle\mathbb{Q};+,\cdot,<\rangle$	$\langle\mathbb{R};+,\cdot,<\rangle$	—
E	$\langle\mathbb{N};\exp\rangle$	—	—	$\langle\mathbb{R};+,\cdot,e^x\rangle$	$\langle\mathbb{C};+,\cdot,e^x\rangle$

New Results

- SALEHI, SAEED; *On Axiomatizability of the Multiplicative Theory of Numbers*, **Fundamenta Informaticæ** 159:3 (2018) 279–296.

  $\langle \mathbb{Q}; \times \rangle, \langle \mathbb{R}; \times \rangle, \langle \mathbb{C}; \times \rangle.$

- ASSADI, ZIBA & SALEHI, SAEED; *On Decidability and Axiomatizability of Some Ordered Structures*, **Soft Computing** 23:11 (2019) 3615–3626.

  $\langle \mathbb{Q}; \times, < \rangle, \langle \mathbb{R}; \times, < \rangle.$

- SALEHI, SAEED & ZARZA, MOHAMMADSALEH; *First-Order Continuous Induction and a Logical Study of Real Closed Fields*, **Bulletin of the Iranian Mathematical Society** online (2019) DOI: 10.1007/s41980-019-00252-0.

  $\langle \mathbb{R}; +, \times, < \rangle.$

New Results

- SALEHI, SAEED; *On Axiomatizability of the Multiplicative Theory of Numbers*, **Fundamenta Informaticæ** 159:3 (**2018**) 279–296.

  $\langle \mathbb{Q}; \times \rangle$, $\langle \mathbb{R}; \times \rangle$, $\langle \mathbb{C}; \times \rangle$.

- ASSADI, ZIBA & SALEHI, SAEED; *On Decidability and Axiomatizability of Some Ordered Structures*, **Soft Computing** 23:11 (**2019**) 3615–3626.

  $\langle \mathbb{Q}; \times, < \rangle$, $\langle \mathbb{R}; \times, < \rangle$.

- SALEHI, SAEED & ZARZA, MOHAMMADSALEH; *First-Order Continuous Induction and a Logical Study of Real Closed Fields*, **Bulletin of the Iranian Mathematical Society** online (**2019**) DOI: 10.1007/s41980-019-00252-0.

  $\langle \mathbb{R}; +, \times, < \rangle$.

New Results

- SALEHI, SAEED; *On Axiomatizability of the Multiplicative Theory of Numbers*, **Fundamenta Informaticæ** 159:3 (**2018**) 279–296.

  $\langle\mathbb{Q};\times\rangle$, $\langle\mathbb{R};\times\rangle$, $\langle\mathbb{C};\times\rangle$.

- ASSADI, ZIBA & SALEHI, SAEED; *On Decidability and Axiomatizability of Some Ordered Structures*, **Soft Computing** 23:11 (**2019**) 3615–3626.

  $\langle\mathbb{Q};\times,<\rangle$, $\langle\mathbb{R};\times,<\rangle$.

- SALEHI, SAEED & ZARZA, MOHAMMADSALEH; *First-Order Continuous Induction and a Logical Study of Real Closed Fields*, **Bulletin of the Iranian Mathematical Society** online (**2019**) DOI: 10.1007/s41980-019-00252-0.

  $\langle\mathbb{R};+,\times,<\rangle$.

New Results

- SALEHI, SAEED; *On Axiomatizability of the Multiplicative Theory of Numbers*, **Fundamenta Informaticæ** 159:3 (**2018**) 279–296.

  $\langle \mathbb{Q}; \times \rangle$, $\langle \mathbb{R}; \times \rangle$, $\langle \mathbb{C}; \times \rangle$.

- ASSADI, ZIBA & SALEHI, SAEED; *On Decidability and Axiomatizability of Some Ordered Structures*, **Soft Computing** 23:11 (**2019**) 3615–3626.

  $\langle \mathbb{Q}; \times, < \rangle$, $\langle \mathbb{R}; \times, < \rangle$.

- SALEHI, SAEED & ZARZA, MOHAMMADSALEH; *First-Order Continuous Induction and a Logical Study of Real Closed Fields*, **Bulletin of the Iranian Mathematical Society** online (**2019**) DOI: 10.1007/s41980-019-00252-0.

  $\langle \mathbb{R}; +, \times, < \rangle$.

FUNDAMENTA INFORMATICAE

Volume 138
Number 3
2015

ISSN 0169-2968
FUINE 138(3) 285–386 (2015)

CONTENTS

IOS Press
VISIT OUR WEBSITE ON HTTP://WWW.IOSPRESS.NL

SAEED SALEHI, **Logic and Computation, & their interactions**, Arak 2019.

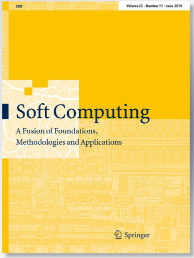
On decidability and axiomatizability of some ordered structures

Ziba Assadi & Saeed Salehi

Soft Computing
A Fusion of Foundations,
Methodologies and Applications

ISSN 1432-7643
Volume 23
Number 11

Soft Comput (2019) 23:3615-3626
DOI 10.1007/s00500-018-3247-1



Springer

SAEED SALEHI, **Logic and Computation, & their interactions**, Arak 2019.

Axiomatizability of Mathematical Structures

A Rather Complete Picture

	$\mathbb{N}$	$\mathbb{Z}$	$\mathbb{Q}$	$\mathbb{R}$	$\mathbb{C}$
$\{<\}$	Δ_1	Δ_1	Δ_1	Δ_1	−
$\{+\}$	Δ_1	Δ_1	Δ_1	Δ_1	Δ_1
$\{\cdot\}$	Δ_1	Δ_1	Δ_1	Δ_1	Δ_1
$\{+,<\}$	Δ_1	Δ_1	Δ_1	Δ_1	−
$\{+,\cdot\}$	$\not\Delta_1$	$\not\Delta_1$	$\not\Delta_1$	Δ_1	Δ_1
$\{\cdot,<\}$	$\not\Delta_1$	$\not\Delta_1$	Δ_1	Δ_1	−
$\{+,\cdot,<\}$	$\not\Delta_1$	$\not\Delta_1$	$\not\Delta_1$	Δ_1	−
E	$\not\Delta_1$	−	−	¿?	$\not\Delta_1$

Tarski's Exponential Function Problem **is open ...**

Is the structure $\langle \mathbb{R};+,\cdot,e^x\rangle$ decidable?

Axiomatizability of Mathematical Structures

A Rather Complete Picture

	$\mathbb{N}$	$\mathbb{Z}$	$\mathbb{Q}$	$\mathbb{R}$	$\mathbb{C}$
$\{<\}$	Δ_1	Δ_1	Δ_1	Δ_1	—
$\{+\}$	Δ_1	Δ_1	Δ_1	Δ_1	Δ_1
$\{\cdot\}$	Δ_1	Δ_1	Δ_1	Δ_1	Δ_1
$\{+,<\}$	Δ_1	Δ_1	Δ_1	Δ_1	—
$\{+,\cdot\}$	$\not\Delta_1$	$\not\Delta_1$	$\not\Delta_1$	Δ_1	Δ_1
$\{\cdot,<\}$	$\not\Delta_1$	$\not\Delta_1$	Δ_1	Δ_1	—
$\{+,\cdot,<\}$	$\not\Delta_1$	$\not\Delta_1$	$\not\Delta_1$	Δ_1	—
E	$\not\Delta_1$	—	—	¿?	$\not\Delta_1$

Tarski's Exponential Function Problem **is open ...**

Is the structure $\langle \mathbb{R}; +, \cdot, e^x \rangle$ decidable?

Axiomatizability of Mathematical Structures

A Rather Complete Picture

	$\mathbb{N}$	$\mathbb{Z}$	$\mathbb{Q}$	$\mathbb{R}$	$\mathbb{C}$
$\{<\}$	Δ_1	Δ_1	Δ_1	Δ_1	—
$\{+\}$	Δ_1	Δ_1	Δ_1	Δ_1	Δ_1
$\{\cdot\}$	Δ_1	Δ_1	Δ_1	Δ_1	Δ_1
$\{+,<\}$	Δ_1	Δ_1	Δ_1	Δ_1	—
$\{+,\cdot\}$	$\not\Delta_1$	$\not\Delta_1$	$\not\Delta_1$	Δ_1	Δ_1
$\{\cdot,<\}$	$\not\Delta_1$	$\not\Delta_1$	Δ_1	Δ_1	—
$\{+,\cdot,<\}$	$\not\Delta_1$	$\not\Delta_1$	$\not\Delta_1$	Δ_1	—
E	$\not\Delta_1$	—	—	¿?	$\not\Delta_1$

Tarski's Exponential Function Problem is open ...

Is the structure $\langle\mathbb{R};+,\cdot,e^x\rangle$ decidable?

Axiomatizability of Mathematical Structures

A Rather Complete Picture

	$\mathbb{N}$	$\mathbb{Z}$	$\mathbb{Q}$	$\mathbb{R}$	$\mathbb{C}$
$\{<\}$	Δ_1	Δ_1	Δ_1	Δ_1	$-$
$\{+\}$	Δ_1	Δ_1	Δ_1	Δ_1	Δ_1
$\{\cdot\}$	Δ_1	Δ_1	Δ_1	Δ_1	Δ_1
$\{+,<\}$	Δ_1	Δ_1	Δ_1	Δ_1	$-$
$\{+,\cdot\}$	$\not\Delta_1$	$\not\Delta_1$	$\not\Delta_1$	Δ_1	Δ_1
$\{\cdot,<\}$	$\not\Delta_1$	$\not\Delta_1$	Δ_1	Δ_1	$-$
$\{+,\cdot,<\}$	$\not\Delta_1$	$\not\Delta_1$	$\not\Delta_1$	Δ_1	$-$
E	$\not\Delta_1$	$-$	$-$	¿?	$\not\Delta_1$

Tarski's Exponential Function Problem **is open ...**

Is the structure $\langle \mathbb{R}; +, \cdot, e^x \rangle$ decidable?

Axiomatizability of Mathematical Structures

A Rather Complete Picture

	$\mathbb{N}$	$\mathbb{Z}$	$\mathbb{Q}$	$\mathbb{R}$	$\mathbb{C}$
$\{<\}$	Δ_1	Δ_1	Δ_1	Δ_1	—
$\{+\}$	Δ_1	Δ_1	Δ_1	Δ_1	Δ_1
$\{\cdot\}$	Δ_1	Δ_1	Δ_1	Δ_1	Δ_1
$\{+,<\}$	Δ_1	Δ_1	Δ_1	Δ_1	—
$\{+,\cdot\}$	$\not\Delta_1$	$\not\Delta_1$	$\not\Delta_1$	Δ_1	Δ_1
$\{\cdot,<\}$	$\not\Delta_1$	$\not\Delta_1$	Δ_1	Δ_1	—
$\{+,\cdot,<\}$	$\not\Delta_1$	$\not\Delta_1$	$\not\Delta_1$	Δ_1	—
E	$\not\Delta_1$	—	—	¿?	$\not\Delta_1$

Tarski's Exponential Function Problem **is open ...**

Is the structure $\langle \mathbb{R}; +, \cdot, e^x \rangle$ decidable?

Axiomatizability of Mathematical Structures

A Rather Complete Picture

	$\mathbb{N}$	$\mathbb{Z}$	$\mathbb{Q}$	$\mathbb{R}$	$\mathbb{C}$
$\{<\}$	Δ_1	Δ_1	Δ_1	Δ_1	−
$\{+\}$	Δ_1	Δ_1	Δ_1	Δ_1	Δ_1
$\{\cdot\}$	Δ_1	Δ_1	Δ_1	Δ_1	Δ_1
$\{+,<\}$	Δ_1	Δ_1	Δ_1	Δ_1	−
$\{+,\cdot\}$	$\not\Delta_1$	$\not\Delta_1$	$\not\Delta_1$	Δ_1	Δ_1
$\{\cdot,<\}$	$\not\Delta_1$	$\not\Delta_1$	Δ_1	Δ_1	−
$\{+,\cdot,<\}$	$\not\Delta_1$	$\not\Delta_1$	$\not\Delta_1$	Δ_1	−
E	$\not\Delta_1$	−	−	¿?	$\not\Delta_1$

Tarski's Exponential Function Problem **is open ...**

Is the structure $\langle\mathbb{R};+,\cdot,e^x\rangle$ decidable?

SAEED SALEHI, **Logic and Computation, & their interactions**, Arak 2019.

Axiomatizability of Mathematical Structures

A Rather Complete Picture

	$\mathbb{N}$	$\mathbb{Z}$	$\mathbb{Q}$	$\mathbb{R}$	$\mathbb{C}$
$\{<\}$	Δ_1	Δ_1	Δ_1	Δ_1	–
$\{+\}$	Δ_1	Δ_1	Δ_1	Δ_1	Δ_1
$\{\cdot\}$	Δ_1	Δ_1	Δ_1	Δ_1	Δ_1
$\{+,<\}$	Δ_1	Δ_1	Δ_1	Δ_1	–
$\{+,\cdot\}$	$\not\Delta_1$	$\not\Delta_1$	$\not\Delta_1$	Δ_1	Δ_1
$\{\cdot,<\}$	$\not\Delta_1$	$\not\Delta_1$	Δ_1	Δ_1	–
$\{+,\cdot,<\}$	$\not\Delta_1$	$\not\Delta_1$	$\not\Delta_1$	Δ_1	–
E	$\not\Delta_1$	–	–	¿?	$\not\Delta_1$

Tarski's Exponential Function Problem `is open ...`

Is the structure $\langle\mathbb{R};+,\cdot,e^x\rangle$ decidable?

Axiomatizability of Mathematical Structures

A Rather Complete Picture

	$\mathbb{N}$	$\mathbb{Z}$	$\mathbb{Q}$	$\mathbb{R}$	$\mathbb{C}$
$\{<\}$	Δ_1	Δ_1	Δ_1	Δ_1	—
$\{+\}$	Δ_1	Δ_1	Δ_1	Δ_1	Δ_1
$\{\cdot\}$	Δ_1	Δ_1	Δ_1	Δ_1	Δ_1
$\{+,<\}$	Δ_1	Δ_1	Δ_1	Δ_1	—
$\{+,\cdot\}$	$\not\Delta_1$	$\not\Delta_1$	$\not\Delta_1$	Δ_1	Δ_1
$\{\cdot,<\}$	$\not\Delta_1$	$\not\Delta_1$	Δ_1	Δ_1	—
$\{+,\cdot,<\}$	$\not\Delta_1$	$\not\Delta_1$	$\not\Delta_1$	Δ_1	—
E	$\not\Delta_1$	—	—	¿?	$\not\Delta_1$

Tarski's Exponential Function Problem **is open ...**

Is the structure $\langle\mathbb{R};+,\cdot,e^x\rangle$ decidable?

SAEED SALEHI, **Logic and Computation, & their interactions**, Arak 2019.

Axiomatizability of Mathematical Structures

A Rather Complete Picture

	$\mathbb{N}$	$\mathbb{Z}$	$\mathbb{Q}$	$\mathbb{R}$	$\mathbb{C}$
$\{<\}$	Δ_1	Δ_1	Δ_1	Δ_1	–
$\{+\}$	Δ_1	Δ_1	Δ_1	Δ_1	Δ_1
$\{\cdot\}$	Δ_1	Δ_1	Δ_1	Δ_1	Δ_1
$\{+,<\}$	Δ_1	Δ_1	Δ_1	Δ_1	–
$\{+,\cdot\}$	$\not\Delta_1$	$\not\Delta_1$	$\not\Delta_1$	Δ_1	Δ_1
$\{\cdot,<\}$	$\not\Delta_1$	$\not\Delta_1$	Δ_1	Δ_1	–
$\{+,\cdot,<\}$	$\not\Delta_1$	$\not\Delta_1$	$\not\Delta_1$	Δ_1	–
E	$\not\Delta_1$	–	–	¿?	$\not\Delta_1$

Tarski's Exponential Function Problem **is open ...**

Is the structure $\langle \mathbb{R}; +, \cdot, e^x \rangle$ decidable?

Axiomatizability of Mathematical Structures

A Rather Complete Picture

	$\mathbb{N}$	$\mathbb{Z}$	$\mathbb{Q}$	$\mathbb{R}$	$\mathbb{C}$
$\{<\}$	Δ_1	Δ_1	Δ_1	Δ_1	–
$\{+\}$	Δ_1	Δ_1	Δ_1	Δ_1	Δ_1
$\{\cdot\}$	Δ_1	Δ_1	Δ_1	Δ_1	Δ_1
$\{+,<\}$	Δ_1	Δ_1	Δ_1	Δ_1	–
$\{+,\cdot\}$	$\not\Delta_1$	$\not\Delta_1$	$\not\Delta_1$	Δ_1	Δ_1
$\{\cdot,<\}$	$\not\Delta_1$	$\not\Delta_1$	Δ_1	Δ_1	–
$\{+,\cdot,<\}$	$\not\Delta_1$	$\not\Delta_1$	$\not\Delta_1$	Δ_1	–
E	$\not\Delta_1$	–	–	¿?	$\not\Delta_1$

Tarski's Exponential Function Problem **is open ...**

Is the structure $\langle\mathbb{R};+,\cdot,e^x\rangle$ decidable?

SAEED SALEHI, **Logic and Computation, & their interactions**, Arak 2019.

Axiomatizability of Mathematical Structures

A Rather Complete Picture

	$\mathbb{N}$	$\mathbb{Z}$	$\mathbb{Q}$	$\mathbb{R}$	$\mathbb{C}$
$\{<\}$	Δ_1	Δ_1	Δ_1	Δ_1	–
$\{+\}$	Δ_1	Δ_1	Δ_1	Δ_1	Δ_1
$\{\cdot\}$	Δ_1	Δ_1	Δ_1	Δ_1	Δ_1
$\{+,<\}$	Δ_1	Δ_1	Δ_1	Δ_1	–
$\{+,\cdot\}$	$\not\Delta_1$	$\not\Delta_1$	$\not\Delta_1$	Δ_1	Δ_1
$\{\cdot,<\}$	$\not\Delta_1$	$\not\Delta_1$	Δ_1	Δ_1	–
$\{+,\cdot,<\}$	$\not\Delta_1$	$\not\Delta_1$	$\not\Delta_1$	Δ_1	–
E	$\not\Delta_1$	–	–	¿?	$\not\Delta_1$

Tarski's Exponential Function Problem **is open ...**

Is the structure $\langle\mathbb{R};+,\cdot,e^x\rangle$ decidable?

Axiomatizability of Mathematical Structures

A Rather Complete Picture

	$\mathbb{N}$	$\mathbb{Z}$	$\mathbb{Q}$	$\mathbb{R}$	$\mathbb{C}$
$\{<\}$	Δ_1	Δ_1	Δ_1	Δ_1	$-$
$\{+\}$	Δ_1	Δ_1	Δ_1	Δ_1	Δ_1
$\{\cdot\}$	Δ_1	Δ_1	Δ_1	Δ_1	Δ_1
$\{+,<\}$	Δ_1	Δ_1	Δ_1	Δ_1	$-$
$\{+,\cdot\}$	$\not\Delta_1$	$\not\Delta_1$	$\not\Delta_1$	Δ_1	Δ_1
$\{\cdot,<\}$	$\not\Delta_1$	$\not\Delta_1$	Δ_1	Δ_1	$-$
$\{+,\cdot,<\}$	$\not\Delta_1$	$\not\Delta_1$	$\not\Delta_1$	Δ_1	$-$
E	$\not\Delta_1$	$-$	$-$	¿?	$\not\Delta_1$

Tarski's Exponential Function Problem **is open** ...

Is the structure $\langle\mathbb{R};+,\cdot,e^x\rangle$ decidable?

Thank You!

Thanks to

The Participants ....For Listening...

and

The Organizers ....For Taking Care of Everything...

SAEEDSALEHI.ir

Thank You!

Thanks to

The Participants ....For Listening...

and

The Organizers ....For Taking Care of Everything...

Thank You!

Thanks to

The Participants ....For Listening...

and

The Organizers ....For Taking Care of Everything...

SAEEDSALEHI.ir