

# GNU Aris

a web application for students

Saksham Attri<sup>1</sup>   Zoltán Kovács<sup>2</sup>   Aaron Windischbauer<sup>2</sup>

<sup>1</sup>Birla Institute of Technology and Science Pilani, Hyderabad Campus, India

<sup>2</sup>The Private University College of Education of the Diocese of Linz, Austria



ThEdu'24, July 2, 2024, Nancy, France



# Abstract

We report on recent improvements to the free logic education software tool **GNU Aris**, including the latest features added during the **Google Summer of Code 2023** project. We focused on making GNU Aris a **web application** to enable almost all users to use it as a standalone offline web application written in a combination of HTML, JavaScript, and WebAssembly. We used the Qt Quick framework with Emscripten to compile the application to WebAssembly. In the report we summarize the user feedback of university students given during **a course on logic**.



# Why yet another software in teaching logic?

- ▶ In fact, it's **not new**, it exists since 2011 (programmed by Ian Dunn, until 2014).
- ▶ The well-known software (Coq, Isabelle, LEAN, etc.) are too complex, too big, too difficult, etc., for **prospective mathematics teachers**.
- ▶ It is usually enough to show just a **taste** of formal logic for most audience, during introductory courses.
- ▶ Aris seems a **good balance** between acceptable syntax for beginners and rich enough to demonstrate both propositional logic and first order logic.

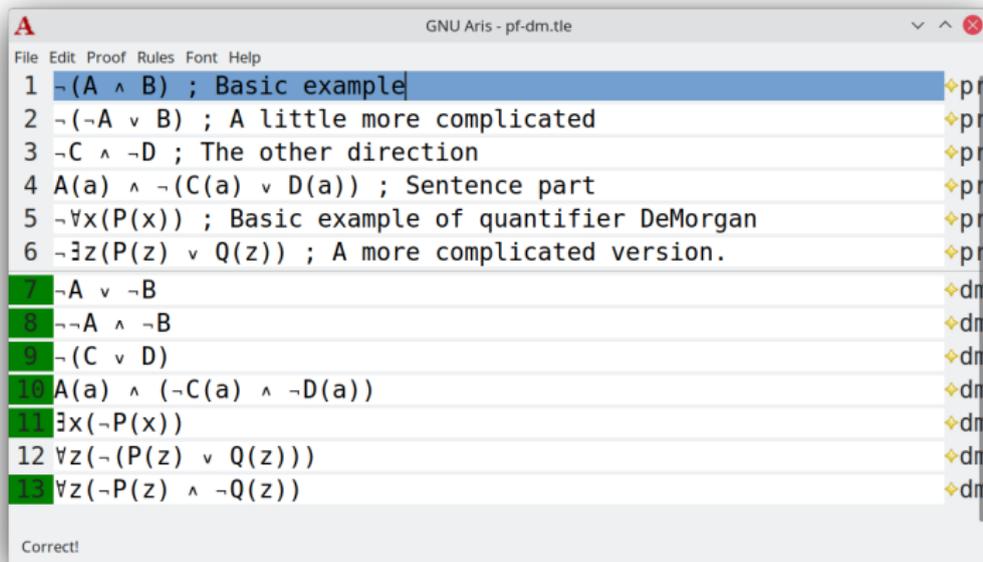


# What is missing from the original version?

- ▶ Difficult **installation** (there are bundles, but they may be outdated, and only for certain platforms).
- ▶ Exotic **shortcuts** for logic operations and symbols (maybe unsupported on some national keyboards).
- ▶ Only **English** is supported.
- ▶ Import/export may be difficult because of its own XML **format**.
- ▶ Random **crashes** on certain inputs.



# GNU Aris 2.2, GTK interface (original)



The screenshot shows the GNU Aris 2.2 GTK interface. The window title is "GNU Aris - pf-dm.tle". The menu bar includes "File", "Edit", "Proof", "Rules", "Font", and "Help". The main area displays a list of logical statements and their corresponding proof methods:

Statement	Proof Method
1 $\neg(A \wedge B)$ ; Basic example	pr
2 $\neg(\neg A \vee B)$ ; A little more complicated	pr
3 $\neg C \wedge \neg D$ ; The other direction	pr
4 $A(a) \wedge \neg(C(a) \vee D(a))$ ; Sentence part	pr
5 $\neg\forall x(P(x))$ ; Basic example of quantifier DeMorgan	pr
6 $\neg\exists z(P(z) \vee Q(z))$ ; A more complicated version.	pr
7 $\neg A \vee \neg B$	dm
8 $\neg\neg A \wedge \neg B$	dm
9 $\neg(C \vee D)$	dm
10 $A(a) \wedge (\neg C(a) \wedge \neg D(a))$	dm
11 $\exists x(\neg P(x))$	dm
12 $\forall z(\neg(P(z) \vee Q(z)))$	dm
13 $\forall z(\neg P(z) \wedge \neg Q(z))$	dm

At the bottom of the window, the text "Correct!" is displayed.



# The Google Summer of Code 2023 project

- ▶ **3 months** of active work (S. A.).
- ▶ Supported by **3 mentors**  
(Z. K., Andreas Ebetshuber and Alexander Thaller).
- ▶ Work plan:
  1. Create a **Qt interface** (transparently).
  2. Compile GNU Aris as a **HTML/WebAssembly** application.
  3. Polish the interface.
  4. Publish the new version.
  5. Get user **feedback**, fix issues.

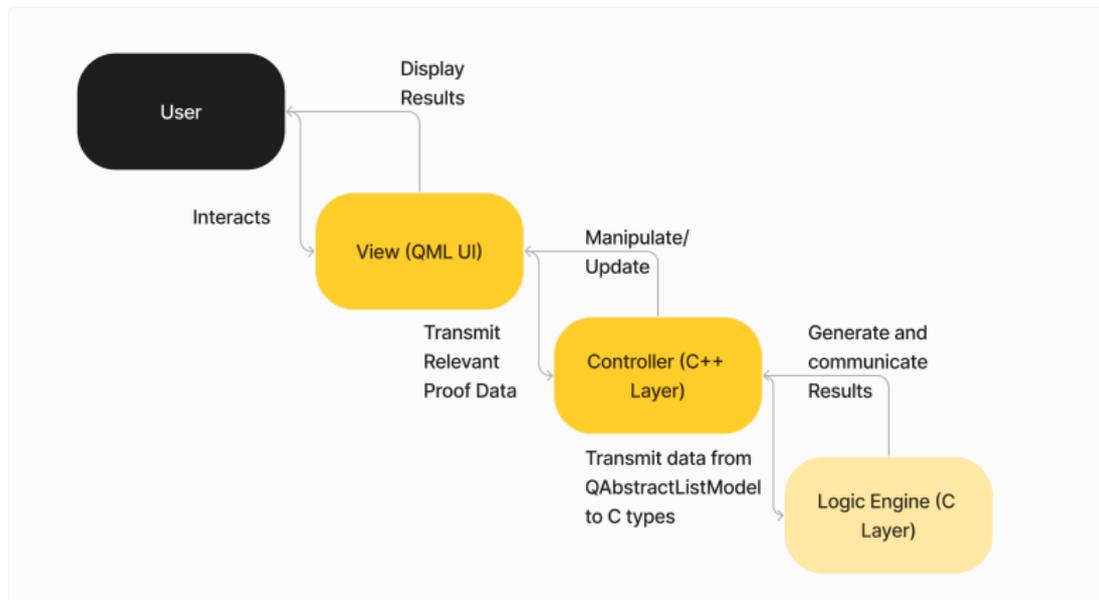


# The Google Summer of Code 2023 project

S. A.



# Preparations for the Qt interface



# Qt native interface

The screenshot shows the GNU Aris software interface. The title bar reads "pf-dm.tle | GNU Aris". On the left, there is a vertical toolbar with logical symbols:  $\wedge$ ,  $\vee$ ,  $\neg$ ,  $\rightarrow$ ,  $\leftrightarrow$ ,  $\forall$ ,  $\exists$ ,  $\top$ ,  $\perp$ ,  $\in$ ,  $\emptyset$ , and a green square. The main area contains a list of 13 items, each with a number, a text input field, and a "premise" label. Items 7 through 13 have additional dropdown menus for "Equivalence" and "DeMorgan", and a number field. Item 12 is highlighted in yellow.

Item	Text	Equivalence	DeMorgan	Number	Label
1	$\neg(A \& B)$ ; Basic example				premise
2	$\neg(\neg A \mid B)$ ; A little more complicated				premise
3	$\neg C \& \neg D$ ; The other direction				premise
4	$A(a) \& \neg(C(a) \mid D(a))$ ; Sentence part				premise
5	$\neg @x(P(x))$ ; Basic example of quantifier DeMorgan				premise
6	$\neg \#z(P(z) \mid Q(z))$ ; A more complicated version.				premise
7	$\neg A \mid \neg B$	Equivalence	DeMorgan	1	
8	$\neg \neg A \& \neg B$	Equivalence	DeMorgan	2	
9	$\neg(C \mid D)$	Equivalence	DeMorgan	3	
10	$A(a) \& (\neg C(a) \& \neg D(a))$	Equivalence	DeMorgan	4	
11	$\#x(\neg P(x))$	Equivalence	DeMorgan	5	
12	$@z(\neg(P(z) \mid Q(z)))$	Equivalence	DeMorgan	6	
13	$@z(\neg(P(z) \& \neg Q(z)))$	Equivalence	DeMorgan	12	



# Web interface

<https://matek.hu/kovzol/aris/>



1	$\exists x(P(x) \wedge Q(x) \wedge A \wedge B)$ :: A basic example of Prenex.				premise	+ / -
2	$A(a) \wedge \exists z(T(z) \wedge F(z) \wedge P \wedge Q)$				premise	+ / -
3	$\exists y(E(y) \mid L(y)) \mid (C \mid W)$				premise	+ / -
4	$\#ul(lu) \mid P(lu) \mid (D \mid K)$				premise	+ / -
5	$\exists x(P(x) \wedge Q(x)) \wedge (A \wedge B)$	Predicate	Prenex	1		+ / -
6	$A(a) \wedge (\exists z(T(z) \wedge F(z)) \wedge (P \wedge Q))$	Predicate	Prenex	2		+ / -
7	$\exists y(E(y) \mid L(y) \mid C \mid W)$	Predicate	Prenex	3		+ / -
8	$\#ul(lu) \mid P(lu) \mid (D \mid K)$	Predicate	Prenex	4		+ / -



# Feedback: A course in logic

For prospective mathematics teachers

- ▶ October 2023 – January 2024
- ▶ 30 students (some with bachelor degree, some without)
- ▶ Discussed topics: propositional logic, first order logic
- ▶ Two examples:
  1. Puzzles from Smullyan's "The Lady or the Tiger?"
  2. Category theory



# A puzzle from Smullyan's "The Lady or the Tiger?"

## 5. Der fünfte Versuch

Es gelten dieselben Regeln, und die Schilder lauteten so:

I

Zumindest in einem  
Raum ist eine Dame

II

Im andern Raum  
ist eine Dame

Wie sollte der Gefangene sich entscheiden?

# A puzzle from Smullyan's "The Lady or the Tiger?"

## THE LADY OR THE TIGER?

### 5 ♦ The Fifth Trial

The same rules apply, and here are the signs:

I  
AT LEAST ONE ROOM  
CONTAINS A LADY

II  
THE OTHER ROOM  
CONTAINS A LADY

## A puzzle from Smullyan's "The Lady or the Tiger?"

### THE SECOND DAY.

"Yesterday was a fiasco," said the king to his minister. "All three prisoners solved their puzzles! Well, we have five trials coming up today, and I think I'll make them a little tougher."

"Excellent idea!" said the minister.

Well, in each of the trials of this day, the king explained that in the lefthand room (Room I), if a lady is in it, then the sign on the door is true, but if a tiger is in it, the sign is false. In the righthand room (Room II), the situation is the opposite: a lady in the room means the sign on the door is false, and a tiger in the room means the sign is true. Again, it is possible that both rooms contain ladies or both rooms contain tigers, or that one room contains a lady and the other a tiger.

# A puzzle from Smullyan's "The Lady or the Tiger?"

## THE LADY OR THE TIGER?

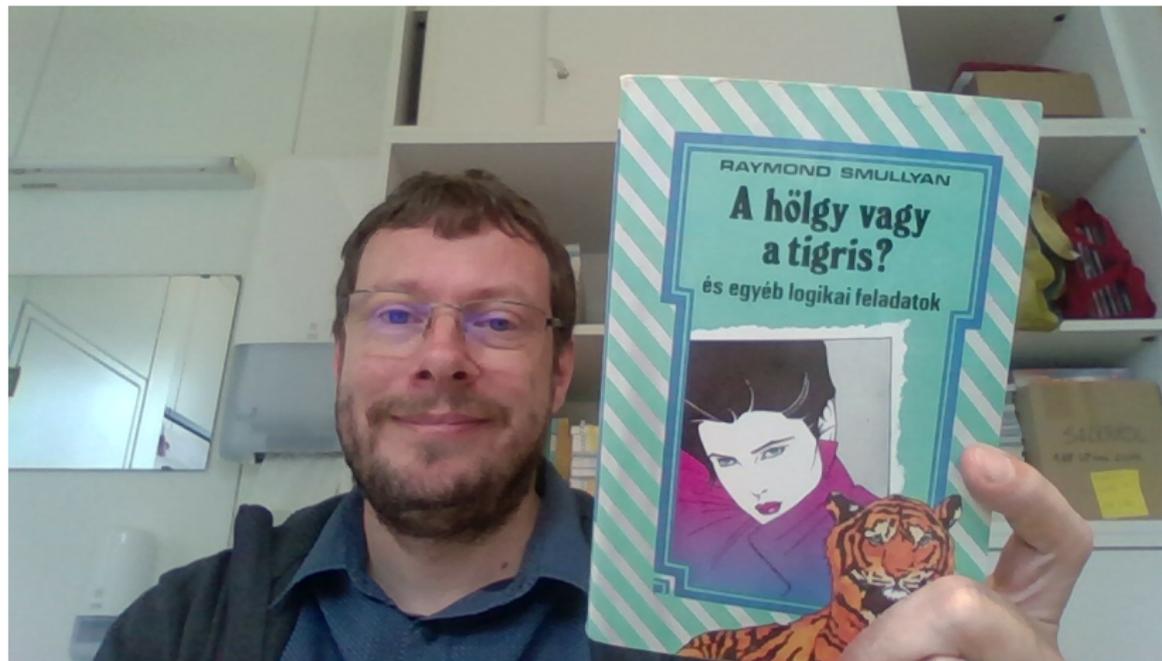
### 5 ♦ The Fifth Trial

The same rules apply, and here are the signs:

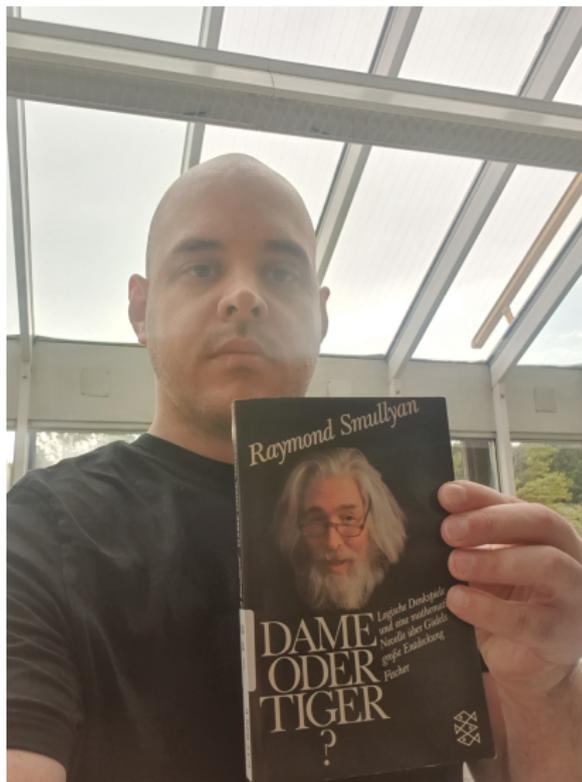
I  
AT LEAST ONE ROOM  
CONTAINS A LADY

II  
THE OTHER ROOM  
CONTAINS A LADY

# A puzzle from Smullyan's "The Lady or the Tiger?"



# A puzzle from Smullyan's "The Lady or the Tiger?"



# A puzzle from Smullyan's "The Lady or the Tiger?"

Solution in the web version of GNU Aris

The screenshot shows the GNU Aris web interface with a logical proof for a puzzle. The proof consists of 26 lines, each with a logical formula, a justification rule, and a line number. The interface includes a vertical navigation bar on the left and a header bar at the top.

Line	Formula	Justification	Line
1	$L1 \vee S1$ :: A lady in room one means sign one is true.	premise	1
2	$\neg L2 \vee S2$ :: A tiger in room two means sign two is true.	premise	2
3	$S1 \rightarrow (L1 \vee L2)$ :: The sign of the first room states that at least one of the rooms contains a lady.	premise	3
4	$S2 \rightarrow L1$ :: The sign of the second room states that the lady is in the other (first) room.	premise	4
5	$L2$ :: We assume that the lady is in room two.	sf	5
6	$(\neg L2 \vee S2) \wedge (S2 \rightarrow \neg L2)$	Equivalence	2
7	$S2 \rightarrow \neg L2$	Inference	6
8	$\neg S2 \vee \neg L2$	Equivalence	7
9	$\neg S2$	Inference	8
10	$(S2 \rightarrow L1) \wedge (L1 \rightarrow S2)$	Equivalence	4
11	$L1 \rightarrow S2$	Inference	10
12	$\neg L1 \vee S2$	Equivalence	11
13	$\neg L1$	Inference	12
14	$(L1 \rightarrow S1) \wedge (S1 \rightarrow L1)$	Equivalence	1
15	$S1 \rightarrow L1$	Inference	14
16	$\neg S1 \vee L1$	Equivalence	15
17	$\neg S1$	Inference	16
18	$(S1 \rightarrow (L1 \vee L2)) \wedge ((L1 \vee L2) \rightarrow S1)$	Equivalence	3
19	$(L1 \vee L2) \rightarrow S1$	Inference	18
20	$\neg (L1 \vee L2) \vee S1$	Equivalence	19
21	$\neg (L1 \vee L2)$	Inference	20
22	$\neg L1 \wedge \neg L2$	Equivalence	21
23	$\neg L2$	Inference	22
24	$L2 \wedge \neg L2$	Inference	5
25	$\perp$	Boolean	24
26	$L2 \rightarrow \perp$ :: The assumption of a lady being in room two leads to a contradiction	subproof	5

# A puzzle from Smullyan's "The Lady or the Tiger?"

Solution in the web version of GNU Aris

26	$L2 \rightarrow I$ :: The assumption of a lady being in room 2 leads to a contradiction.			subproof	5	+ / -	
27	$\neg L2 \vee I$	Equivalence	Implication		26	+ / -	
28	$\neg L2$ :: There is no lady in room 2. Therefore, a tiger is in room 2.	Boolean	Identity		27	+ / -	
29	$(\neg L2 \rightarrow S2) \wedge (S2 \rightarrow \neg L2)$	Equivalence	Equivalence		2	+ / -	
30	$\neg L2 \rightarrow S2$	Inference	Simplificati		29	+ / -	
31	$S2$	Inference	Modus Pone		30	28	+ / -
32	$(S2 \rightarrow L1) \wedge (L1 \rightarrow S2)$	Equivalence	Equivalence		4	+ / -	
33	$S2 \rightarrow L1$	Inference	Simplificati		32	+ / -	
34	$L1$ :: The lady is in room 1.	Inference	Modus Pone		33	31	+ / -
35	$(L1 \rightarrow S1) \wedge (S1 \rightarrow L1)$	Equivalence	Equivalence		1	+ / -	
36	$L1 \rightarrow S1$	Inference	Simplificati		35	+ / -	
37	$S1$	Inference	Modus Pone		36	34	+ / -
38	$(S1 \rightarrow (L1 \vee L2)) \wedge ((L1 \vee L2) \rightarrow S1)$	Equivalence	Equivalence		3	+ / -	
39	$S1 \rightarrow (L1 \vee L2)$	Inference	Simplificati		38	+ / -	
40	$L1 \vee L2$	Inference	Modus Pone		39	37	+ / -
41	$L1 \wedge \neg L2$ :: The prisoner should pick room 1, where the lady is. Room 2 contains a tiger.	Inference	Conjunction		34	28	+ / -

# An exercise from category theory, $\text{\LaTeX}$ export

On left and right identity elements of a commutative binary operation

1.	$\forall a \forall b (o(a, b) = o(b, a))$	
2.	$\exists e \forall a (o(a, e) = a)$	
<hr/>		
3.	$\forall a (o(a, e_-) = a)$	ei (2)
4.	$o(x, e_-) = o(e_-, x)$	ui (1)
5.	$o(x, e_-) = x$	ui (3)
6.	$o(e_-, x) = x$	fv (4,5)
7.	$\forall a (o(e_-, a) = a)$	ug (6)
8.	$\exists f \forall a (o(f, a) = a)$	eg (7)



# Feedback: A course in logic

## Feature requests by the students

- ▶ More comprehensible and detailed **error messages**.
- ▶ Automatic updates on the **feedback** for the correctness of statements after changing faulty inputs.
- ▶ A feature to **zoom** for smaller device displays.
- ▶ An option to change the user **language**.
- ▶ An option to highlight, **copy and paste** inputs.
- ▶ An option to change the **type of input** between premise and conclusion after inserting a new line.
- ▶ An operator for **exclusive disjunction**.
- ▶ An option to use the rule of interference for a proof by contrapositive.



THANK YOU!

# References I



The Private University College of Education of the Diocese Linz (2018): *Curriculum – Bachelorstudium und Masterstudium für das Lehramt Primarstufe*.  
[https://www.phdl.at/fileadmin/user\\_upload/3\\_Service/2\\_Studienbetrieb/Mitteilungsblatt/MB-012-2018-20180621\\_CurrPrimar\\_BAC\\_MA\\_PH\\_Linz-veroeffentlicht.pdf](https://www.phdl.at/fileadmin/user_upload/3_Service/2_Studienbetrieb/Mitteilungsblatt/MB-012-2018-20180621_CurrPrimar_BAC_MA_PH_Linz-veroeffentlicht.pdf).



Ian Dunn (2014): *GNU Aris*.  
<https://www.gnu.org/software/aris/manual/aris.pdf>.



Qt Group (2010): *Qt QML*.  
<https://doc.qt.io/qt-6/qtqml-index.html>.



Zoltán Kovács (2023): *Logik für Lehramt*.  
<https://matek.hu/kovzol/Logik-20230214.pdf>.



Chu-Min Li & Felip Manyà, editors (2021): *Theory and Applications of Satisfiability Testing – SAT 2021*.  
Springer Cham, doi:10.1007/978-3-030-80223-3.



University College of Education of Salzburg (2019): *Curriculum – Bachelor- und Masterstudium – Primarstufe*.  
[https://qmpilot.phsalzburg.at/File/CoreDownload?id=2170&filename=PG\\_098\\_099\\_Curriculum\\_Primarstufe\\_BA\\_MA-60\\_MA-90\\_red.pdf&langId=1](https://qmpilot.phsalzburg.at/File/CoreDownload?id=2170&filename=PG_098_099_Curriculum_Primarstufe_BA_MA-60_MA-90_red.pdf&langId=1).

# References II



LiLeS Linzer Lehramt Sekundarstufe (2022): *Curriculum Bachelorstudium Lehramt Sekundarstufe (Allgemeinbildung) Entwicklungsverbund "Cluster Mitte"*.

[https://www.liles.at/fileadmin/user\\_upload/BEd\\_2021.pdf](https://www.liles.at/fileadmin/user_upload/BEd_2021.pdf).



Raymond M. Smullyan (1982): *The Lady or the Tiger?: And Other Logic Puzzles. Including a Mathematical Novel That Features Gödel's Great Discovery*. Alfred A. Knopf, New York.



Raymond M. Smullyan (1985): *Dame oder Tiger? Logische Denkspiele und eine mathematische Novelle über Gödels große Entdeckung. Aus dem Amerikanischen von Thea Brandt*.

Fischer Taschenbuchverlag, Frankfurt am Main.



Church University College Edith Stein (2022): *Curriculum Bachelor- und Masterstudium – für das Lehramt Primarstufe*.

[https://www.kph-es.at/fileadmin/user\\_upload/Curriculum\\_Bachelor-\\_und\\_Masterstudium\\_Lehramt\\_Primarstufe2022.pdf](https://www.kph-es.at/fileadmin/user_upload/Curriculum_Bachelor-_und_Masterstudium_Lehramt_Primarstufe2022.pdf).



Bjarne Stroustrup (2013): *The C++ Programming Language*.

Addison-Wesley Professional, Boston, MA.



Makarius Wenzel (2023): *The Isabelle/Isar Reference Manual*.

<https://isabelle.in.tum.de/doc/isar-ref.pdf>.